KUNGL
TEKNISKA
HÖGSKOLAN

Compressible Turbulent Flows: LES and Embedded Boundary Methods

Marco Kupiainen

Doctoral Dissertation

Royal Institute of Technology

Department of Numerical Analysis and Computer Science

Akademisk avhandling som med tillstånd av Kungl Tekniska Högskolan framlägges till offentlig granskning för avläggande av teknologie doktorsexamen 2 april 2009 kl 13.00 i sal F3, Lindstedtsvägen 26, Kungliga Tekniska högskolan, Stockholm.

ISBN 978-91-7415-258-6
TRITA-CSC-A 2009:05
ISSN 1653-5723
ISRN KTH/CSC/A–09/05-SE

Universitetsservice US AB,

Abstract

The thesis considers numerical approximations to solutions of the compressible Navier-Stokes and Large-Eddy Simulation (LES) equations.

An embedded boundary method for representing geometries within the computational domain is considered. Test examples indicate that the discretization errors from the embedded boundary manifest as numerical 'roughness' when the flow is turbulent and numerically unresolved. For low-Reynolds number configurations however the errors made on the embedded boundary are of the same order of magnitude as the errors made by the internal discretization scheme.

The computational grid is Cartesian. This imposes a hard limit on the viable resolving power of boundary layers on the method for high-Reynolds number configurations, even in the presence of local grid refinement. To mitigate this severe limitation wall-models can be used. They in effect model the near-wall-behaviour instead of resolving the thin boundary-layer associated with high-Reynolds number flows. We have tested one wall-model for LES in this thesis and we conclude that the models do not yield perfect results.

ISBN 978-91-7415-258-6 • TRITA-CSC-A 2009:05 • ISSN 1653-5723 • ISRN KTH/CSC/A–09/05-SE

Sammanfattning

Avhandlingen betraktar numeriska approximationer till lösningar till kompressibla Navier-Stokes och Large Eddy Simulation (LES) ekvationerna. En inbäddad randmetod för att representera geometrier inuti beräkningsområdet betraktas. Testexempel indikerar att diskretiseringsfelen från den inbäddade randen visar sig som numerisk 'grovhet' när flödet är turbulent och numeriskt underupplöst. För låg-Reynoldstalsflöden är felen från den inbäddade randen av samma storleksordning som för det interna diskretiseringsschemat.

Beräkningsnätet är ekvidistant och kartesiskt, vilket begränsar starkt metodens möjligheter att lösa upp gränsskikt för flöden med höga Reynoldstal, även om man använder lokal nätförfining. För att lindra detta tillkortakommande kan man använda väggmodeller. Dessa modellerar vad som händer i närheten av en fast vägg, istället för att lösa upp de tunna gränsskikt som uppkommer för flöden med höga Reynoldstal. I denna avhandling har vi provat en väggmodel för LES och slutsatsen är att resultaten inte är perfekta.

ISBN 978-91-7415-258-6 • TRITA-CSC-A 2009:05 • ISSN 1653-5723 • ISRN KTH/CSC/A–09/05-SE

Acknowledgments

First of all I wish to thank my supervisor Professor Björn Sjögreen for giving me the opportunity to work in the field of computations. Furthermore I am grateful for all the help and guidance he has provided, and especially I am very grateful to him for inviting me to come to Lawrence Livermore National Laboratories to visit. Without the two visits this work would probably never have become anything. Professor Jesper Oppelstrup must be given a lot of credit, especially during the last phase of the thesis work for tying thing up. Professor Bjorn Engquist is also gratefully acknowledged for all giving and interesting discussions we've had. I am also grateful to PDC, for letting me use their supercomputers and especially Nils Smeds, who helped me initially to port an early research version of FOAM^{TM} to their machines. I would like to thank my former secondary advisor Professor Christer Fureby for his engagement and skill with LES/combustion modelling and general FOAM usage. Without Christers, **Papers I-III** would never have been made. I could not have managed without the support of Professor Gunilla Kreiss and Professor Olof Runborg, whos advice and encouragement have helped me to pull through the final part of this thesis. All present and former colleagues at both UPMC and NADA are also gratefully acknowledged.

I would also like to thank Professor Pierre Sagaut for having me as a post-doc student during one year at Universite Pierre et Marie Curie in Paris. Without the assistance of Pierre this thesis would certainly never have been finished.

Donna Mapes and Roscoe are gratefully acknowledged for being a great hosts and a good friend in Livermore. Oscar, Simon and Morristown, Mom, Dad, Sussi and Annika & Castor you mean everything to me. This theses is dedicated to the loving memory of Wilma.

The Foundations NorFa, Lars Hierta, Knut & Alice Wallenberg are also gratefully acknowledged for financial support during this thesis work.

Contents

List of Figures

Chapter 1

Scope of work/Summary

The prediction of turbulent compressible flows is a major challenge in terms of modeling, numerical algorithms, and computer performance. The motivating application for these investigations is the modeling of the solid fuel rocket exhaust plume behind a laser guided missile, with an aim to predict the refraction of a laser beam used to communicate with the missile.

Large Eddy simulations of a simplified model with non-reacting and reacting base flow using the FOAM software (**Paper I, II, III**) were set up and compared to measurements [32] in an attempt to clarify the effects and applicability of different approaches to constructing the sub-grid scale filter in **L**arge **E**ddy **S**imulations.

The sub-grid scale filter can be analyzed more conclusively if a scheme with known 'characteristics' is employed so discretization errors and filter action can be separated. This proved to be non-trivial in the FOAM code as it was at the time, and it was decided to look to another family of schemes, the recently introduced Embedded Boundary methods [70, 13]. By virtue of primarily using rectangular grids, even for complex geometries, they offer simplicity and efficiency in terms of implementation and computer resources. The boundary conditions are harder to set, but successful applications of high-order Dirichlet and Neumann boundary conditions for second order wave equations [43, 45] indicated that these problems can be overcome. The embedded boundary technology is very well suited for simulations with moving boundaries. The mesh can be fixed, alleviating the need for grid speed contributions, and the calculations of boundary conditions on a moving solid boundary are not expensive.

It is evident that a Cartesian grid cannot, for obvious reasons of cell aspect ratio, offer the grid point economy in thin solid boundary layers that a boundary-fitted grid can. The Cartesian grid technique must therefore be combined with a smart grid-hierarchy such as the building-cube [57], and also with adaptive mesh refinement, like Berger et. al. [4, 5] to become a fully functional tool.

The work presented here represents the first steps towards a large eddy simulator for compressible turbulence in complex geometries using Cartesian grid methods.

The basic scheme is derived from high-resolution schemes [65, 30, 25], offering sharp discontinuity resolution and second order of accuracy for smooth solutions. Such schemes on boundary fitted grids, referred to as **I**mplicit **LES** [29, 27, 28, 21], have been successful even without specific sub-grid scale filter because of the dissipation provided by the limiters and "entropy fixes". It is believed that the embedded boundary technology developed here can be used also together with more-than-second-order schemes such as the Sjögreen/Yee central schemes [72] for compressible turbulent flow simulations in complex geometries.

Boundary conditions with small-cell stiffness mitigation, proposed in [43] are designed to give continuous dependence on details of grid position w.r.t. boundaries. The use of Kreiss/Peterson, Sjögreen/Peterson embedded boundaries for compressible Navier-Stokes/LES is new. The implementation of the method was made for parallel supercomputer utilizing MPI (Message Passing Interface). The implementation was tested on flows around simple geometries for supersonic high-Re flows and compared to boundary-fitted grid results (**Paper IV, V**). Wall quantities such as friction coefficient and pressure coefficient are of primary interest in applications to fluid-solid interaction simulations.

The 3D test case - the Dutton base-flow experiment [32, 53, 9, 8] - brings us back to the type of application which initiated the work (**Paper VI**). The acceleration of the core flow downstream is strongly influenced by the wall boundary layer which is shed as a turbulent free shear layer. A thick boundary layer will not accelerate the core flow as much as a thin boundary layer.

The MUSCL-scheme with an entropy fix adds artificial dissipation in the boundary layer, and thickens it. The MUSCL effects of artificial viscosity dominate, 2:nd order Godunov physical/LES viscosity is not resolved with current resolution.

1.1 List of papers

The thesis is condensed from the following articles, which the author has contributed to. The papers are included in the end of the thesis.

1.1.1 Paper I

Roger Mattsson, Marco Kupiainen, Per Gren, Anders Wåhlin, Torgny E. Carlsson and Christer Fureby, *Pulsed TV Holography and Schlieren Studies, and Large Eddy Simulations of a Turbulent Jet Diffusion Flame*, Combustion and Flame 139 (2004) 1-15. The author of this thesis contributed to some of the ideas presented, performed the numerical simulations and wrote parts of the manuscript.

1.1.2 Paper II

Christer Fureby and Marco Kupiainen, *Large Eddy Simulation of Supersonic Axisymmetric Baseflow.* In Third International Symposium on Turbulence and Shear

Flow Phenomena (TSFP-3), Sendai, Japan 25-27 June 2003. The author of this thesis contributed to the ideas presented, performed the numerical simulations and wrote the manuscript.

1.1.3 Paper III

Marco Kupiainen and Christer Fureby, *Large Eddy Simulation of a Turbulent Non-premixed Flame*. In Third International Symposium on Turbulence and Shear Flow Phenomena (TSFP-3), Sendai, Japan 25-27 June 2003. The author of this thesis contributed to some of the ideas presented, performed the numerical simulations and wrote parts of the manuscript. The author also presented the paper at TSFP-3.

1.1.4 Paper IV

Marco Kupiainen and Björn Sjögreen, *A Cartesian Embedded Boundary Method for the Compressible Navier-Stokes Equations*. Accepted for pulication in Journal of Scientific Computing 2008. The author of this thesis implemented the method solving the Navier-Stokes equations and wrote parts of the paper. The author presented a part of this work at BIT Circus Conference 2006, Stockholm and ECCOMAS 2008, Venice.

1.1.5 Paper V

Marco Kupiainen, *On the Accuracy of an Immersed/Embedded Boundary Method for Compressible Turbulent flow* The author wrote the computer code, wrote the manuscript and did the analysis. The author presented a part of this work at ECCOMAS 2008 conference, Venice.

1.1.6 Paper VI

Marco Kupiainen, *LES Baseflow simulations using a Cartesian Embedded Boundary Method*. The author wrote the computer code, wrote the manuscript and did the analysis. The author presented a part of this work at ECCOMAS 2008 conference, Venice.

Chapter 2

Introduction

There are three major ways to investigate and analyze complex flows, namely making physical experiments in laboratories using e.g. wind-tunnels, using mathematical analysis and doing numerical simulations of the flow. These methods should be seen as complements to one another, since they are beneficial in different ways.

An example of an application is the control of a supersonic missile by laser guidance from the aft. The laser-beam must travel through the bas-flow shocks and exhaust plume which is very hot, turbulent, and particle-laden. A robust system to steer the missile needs to know what the flow field in the plume looks like. Experiments in a wind-tunnel are very challenging and costly. Numerical simulation is the remaining option. The questions are how accurate such computational predictions are, which methods to use and how large the computational effort/time will be.

The thesis surveys commonly used methods for these types of applications and shows calculations performed with finite volume schemes and difference methods. The development of a Cartesian grid embedded boundary method for Navier-Stokes/LES equations modeling for these types of flows is the major contribution of this thesis.

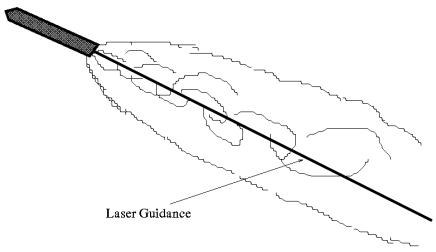


Figure 2.1. Control by laser guidance.

2.1 Computational Fluid Dynamics

The Navier-Stokes equations which describe fluid flow processes are complicated and the existence of long-time solutions is (Nov. 2008) still an open mathematical problem. A proof would earn the prover the Clay Prize of a million dollars. Leaving such fundamental questions aside, the engineering and scientific communities since von Neumann are developing the technology of Computational Fluid Dynamics (CFD) now used routinely both by researchers and engineers to study and understand physical phenomena of fluid flow. Also since von Neumann the need for better flow simulations has been an important driving force for development of high-performance computers. CFD is an interdisciplinary field involving fluid mechanics, the theory of partial differential equations, computational geometry, numerical analysis, and the computer science of programming algorithms and processing data structures. The field has matured substantially as an engineering discipline. There are many commercial flow solvers and the market for CFD software was estimated in 2006 at more than 150 MUSD p.a. (COMSOL, private comm.) Whereas turbulence models for equations of mean flow, when tuned to the problem at hand, provides answers for aeronautics engineers, accurate prediction of turbulent flow from first principles is yet to be demonstrated for all but the simplest flow geometries.

2.2 Turbulent Flows

The notion of turbulence and its meaning is generally accepted. The literature is vast. Some overview work is presented in e.g. [64], [69], [75], [56], [77] and [33]. In [33] the following definition is made:

> *Turbulent fluid motion is an irregular condition of flow in which the various quantities show a random variation with time and space coordinates so that statistically distinct average values can be discerned.*

Turbulence is a flow phenomenon with a wide range of scales in both time and space, characterized by low momentum diffusion, high momentum convection, and pressure and velocity variation with time. Flows that are not turbulent are usually called laminar. The (dimensionless) Reynolds number, $Re = \frac{\rho_\infty U_\infty \mathcal{L}}{\mu_\infty}$ is a measure of the ratio of inertia forces to the molecular viscosity effect, which characterizes whether flow conditions lead to laminar or turbulent flow. It is further generally assumed that turbulence is a continuum phenomenon, an assumption applicable to liquids and gases under atmospheric conditions. Some caution is dictated in the case of hypersonic flows, which can have regions of very low density, e.g. on the lee-side of a wing at high angle of attack, as happens on the Space Shuttle. The continuum assumption is violated when the ratio of the molecular mean free path length to a representative physical length scale, the *Knudsen* number, is small.

A characteristic feature of turbulent flows is the occurrence of eddies of different size. Kolmogorov's famous theory from 1941 [40], [41] (later refined in 1962 [42]), for homogeneous turbulence makes two key assumptions. (i) There is a steady transfer of kinetic energy from the large scales to the small scales and kinetic energy is consumed at the small scales by viscous dissipation. (ii) There is an inertial subrange of scales where the turbulence generating processes are independent of the details of the large scales. The eddies that carry most energy have a characteristic length scale called the *integral length scale* l. The *Kolmogorov scale* η is the smallest length scale in turbulent flows. The range of scales in between is called the *inertial subrange*, see Figure 2.2.

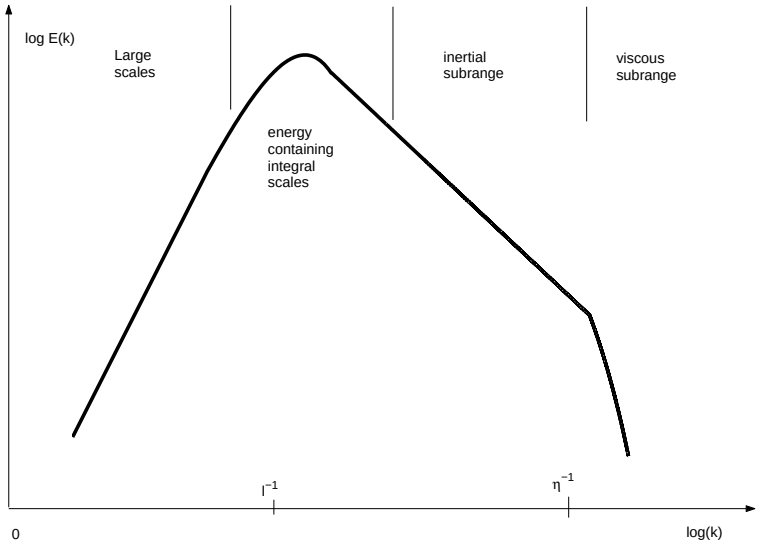


Figure 2.2. Schematic representation of the turbulent kinetic energy spectrum as a function of the wavenumber k.

2.3 Common Levels of Approximation

Numerical prediction methods of turbulence are often classified as Direct Numerical Simulation (DNS), Large Eddy Simulation (LES), or Reynolds Averaged Navier Stokes (RANS) models. In DNS all time and space scales are resolved, e.g. [64]. LES, in which only the larger scales are resolved and sub-grid models are used to represent the effects of the small scales upon the large resolved scales, is described in e.g. [66]. RANS models, in which the mean velocity, density, temperature etc. are solved for, are discussed e.g. [79] and references therein. DNS, LES and RANS can be considered as complementary methods to each other, providing different

levels of information and accuracy, with different areas of application. A schematic picture of the different approaches is presented in Figure (2.3).

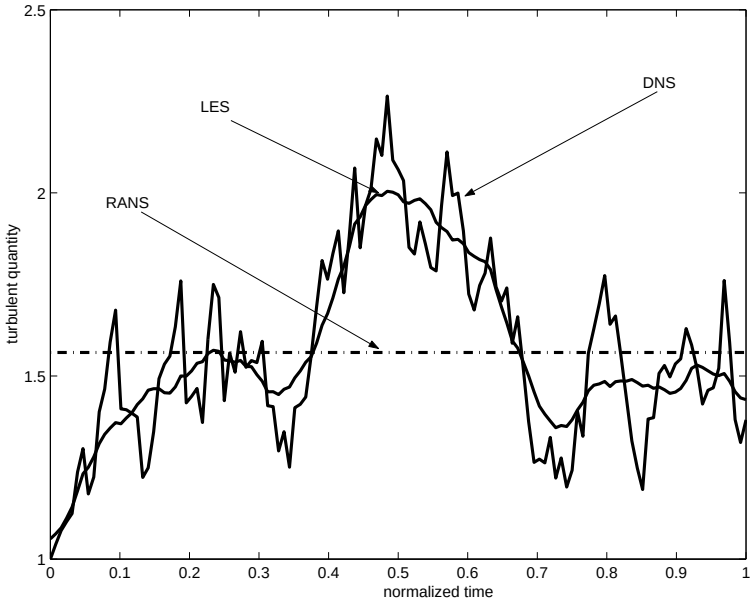


Figure 2.3. A schematic, illustrating the conceptual ideas of RANS, LES and DNS.

2.3.1 Direct Numerical Simulation (DNS)

DNS solves the unsteady Navier-Stokes equations with all scales (time and space) taken into account. This is equivalent to saying that the space-time resolution of the discretization should be at least as fine as the characteristic scales of the continuous problem, i.e. the Kolmogorov scale η, which means exorbitant computational cost. Most interesting flows have much higher Reynolds numbers than can currently be simulated with DNS. The current Petaflops generation of supercomputers are orders of magnitude too small and slow for any but the simplest geometries. Applications in aeronautics and ship design deal with Reynolds numbers as high as 10^8 raising the need of 10^{24} space-time grid points and 10^{27} floating point operations.

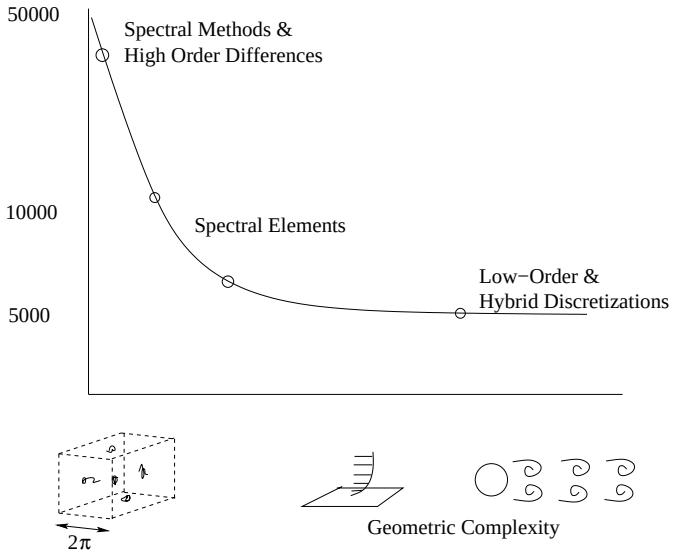


Figure 2.4. Conceptual overview of DNS of turbulent flows: Maximum Reynolds number versus geometric complexity, [38].

2.3.2 Large Eddy Simulation (LES)

In LES the number of resolved scales is reduced by spatial low-pass filtering and hence in the solution process the dynamics of all the scales are not computed explicitly. A cut-off scale is defined, below which the scales are not solved for explicitly, usually denoted sub-grid scales. The non-linearities in the Navier-Stokes equations couple all the scales of the original problem, both above and below the cut-off scale, and in order for the resolved scales to take into account the effects of sub-grid scales, models need to be used. The models are to represent the interaction between the grid scale (resolved scales) and the sub-grid scales. Furthermore, the small scales are assumed to be more universal in character and are therefore believed to be more easily modeled. This assumption is however known to be invalid for near-wall flows, and the development of LES-wall models is an active field of research.

The models increase the computational cost, but by using simple and universal models, a reduction by several orders of magnitude in number of operations is obtained when comparing to resolving all physical scales. The curse of dimensionality is also present in LES. In order to approach a feasible solution algorithm one needs to use some kind of *turbulence modeling* and use a well implemented *efficient method* that is *adaptive* in both space and time.

LES has four conceptual steps:

(i) A spatial *filtering* operation to decompose the equations into filtered (grid-scale) components and residual (sub-grid scale, SGS) components, i.e.

$$\phi = \overline{\phi} + \phi',$$

where $\overline{\phi}(x) = \int_\Omega G(x-\xi,\Delta)\phi(\xi)d\Omega$ is the space filtering convolution, see e.g. [66]. The filtered component $\overline{\phi}$ represents the large scales, that are used explicitly in the calculations and ϕ' represents the small scales whose effect must be modeled. We also introduce Favre filtering defined by

$$\tilde{\phi} = \frac{\overline{\rho\phi}}{\overline{\rho}},$$

where ρ is the density, which is often used in compressible LES formulations.

(ii) The equations for the evolution of the filtered components are derived from the Navier-Stokes equations. These equations are of the same form as the original equations, except that they contain new terms arising from the residual motions.

(iii) Closure is obtained by modeling the SGS terms.

(iv) The filtered equations are solved numerically for $\overline{\phi}$, which is an approximation to the large-scale motions in one realization of the turbulent flow.

The time-averaged mean-flow quantities

$$\langle\phi\rangle_{t_0}^{t_i} = \frac{1}{t_i - t_0}\int_{t_0}^{t_i}\phi(\cdot,\tau)d\tau$$

can be compared to experimental data. In the LES-community there are mainly two views on the separation of modeling issues (i)-(iii) and numerical solution (iv). One school sees filtering and modeling as independent of numerical methods, in particular independent of the computational grids used. The other viewpoint is that modeling and numerical issues cannot practically be separated and hence should deliberately be combined, e.g. [7], [28].

2.3.3 Reynolds Averaged Navier Stokes (RANS)

The RANS approach replaces the Navier-Stokes equations with a set of time-averaged flow equations for the mean flow quantities.

$$\overline{\phi}(x)_T = \frac{1}{T}\int_{t_0}^{t_0+T}\phi(x,t)dt.$$

The formal averaging procedure results in a hierarchy of equations that requires closure by models for averages of products of fluctuations. Extensive efforts have

gone into deriving closure models, simply called turbulence models. Key elements of these models are parameters obtained both from fitting solutions of the equations to experimental data and from detailed computations. Both experiments and DNS have played important parts in deriving and calibrating turbulence closure models, see e.g. [79] and [64]

2.4 Compressible Turbulence

Significant effects of compressibility appear when the Mach number $M_\infty = \frac{U_\infty}{c}$, the ratio between flow speed U_∞ and the speed of sound c, exceeds approximately 0.3. Turbulent compressible flows have been significantly harder to compute, due to the increased complexity introduced by the energy equation and the need for modeling its closure terms. The turbulence phenomena where compressibility plays an important role are for instance: in (1) transonic and higher speed aerodynamics, (2) high-speed combustion, and (3) magneto hydrodynamic (MHD) flows related to nuclear fusion physics. Here we focus on non-MHD compressible turbulent flows. While the classical turbulence studies have been conducted on incompressible flows, see e.g. [66] for a good review, LES for compressible turbulence e.g [19], [52], [39], [58], [16] and [54], is becoming a field of its own. Methods and theory concerning incompressible flows have been modified to treat compressible flows. Compressible high-speed flow involves, apart from turbulence, also non-smooth features such as shocks and shocklets, [62], making the requirements on the numerical method harder than for incompressible flow problems. The instabilities due to turbulence occur on many scales, although in some sense, averaged quantities may be stable and coherent large-scale features may evolve. The interplay between the large-scale and small-scale motions dominates the problem and compressibility affects this interplay, [17].

2.5 Numerical Approaches for Compressible Flows

Below a brief description of general numerical approaches and the most common methods for flow computations is presented.

Finite Element Methods

The finite element method (FEM) is based on a variational principle, where the solution is sought in a finite dimensional function space. FEM can be used on basically any kind of mesh. The solution will be a linear combination of functions from this function space. For large classes of problems FEM is supported by rigorous mathematical theory and error-estimates. Adaptivity can easily be included into the method. Application of FEM in the time domain gives rise to systems of equations that have to be solved in each time step. The most notable effort in FEM for compressible flows is the Discontinuous Galerkin method [12], where the

basis functions are discontinuous, either in time or space. The efficiency of FEM relies heavily on efficient linear solvers and effective preconditioners for these. When strong shocks are present the computational burden can become significant; for an analysis of this fact for a scalar non-linear conservation law see [10].

Spectral Methods

Spectral methods are a special kind of FEM method see e.g. [11]. The basis functions are typically based on Fourier approximations or Chebyschev polynomials and have non-local support. These methods are very accurate (exponential convergence rate for $\mathcal{C}^\infty$ solutions) and are mainly used for DNS. The difficulties with discontinuities in the solution , such as shock waves, can be handled with so called pseudo-spectral methods. Another limitation is that these methods admit only very simple geometries, making them inapplicable for flow problems around or inside complex bodies. Spectral methods owe efficiency to the FFT algorithm, and its implementation: For parallel processor computations often the whole solution must be communicated across all processors several times per time step, e.g. [2]. Also, local mesh refinement is difficult and adaption to local features, such as inlet disturbances, may require expensive global mesh refinement.

Finite Volume methods

Finite volume methods are based upon the integral formulation (using cell-averaged quantities) of the PDE. The usage of methods dominate computational aeronautics since 1970's. Unstructured finite volume methods do not easily generalize to higher order methods, mainly because there is no straightforward method to make the difference approximations more accurate when cells (control volumes) are arbitrarily shaped. Adaptivity is also in principle easy, but there are no rigorous error estimates, such as with FEM, and one estimates the error using some *ad hoc* principle. Finite volume methods can be used on all kinds of meshes.

Finite Difference Methods

The finite difference method is based on the differential formulation of the PDE see e.g. [74]. If we interpret cell-averaged quantities in the finite volume method as point values we can interpret the method as a finite difference method. These methods are based on Taylor's formula to approximate the solution and do not work on discontinuous solutions, as finite volume schemes and discontinuous Galerkin schemes do. Finite difference methods are always used on structured meshes.

2.6 Computational Grids for CFD

The methods above must operate on some data representation to solve the computational task at hand. With data representation we mean type of grid and how the

handling of geometries within the computational domain is done.

2.6.1 Unstructured grids

Unstructured grids cover the computational domain with "arbitrarily" shaped, often tetrahedral or hexahedral, control volumes and approximate the integral formulation in each control volume, see Figure 2.5. Due to the arbitrary shapes allowed in the method, complex geometric objects are in easy to capture. Many software packages can generate such grids from a CAD-model e.g. ICEMCFD [35] and Gambit [36]. Unstructured grids are used in FEM and in Finite Volume methods. Due to the irregularity of the mesh Taylor expansion methods are not easily applicable.

With ordinary unstructured finite volume methods usually only second order (formal) accuracy is achieved. Of course the quality of the simulation depends on the mesh quality. A rule of thumb is that the cell-size should not increase or decrease with more that 20% from one control volume to another (which still is very much!), in order for the mesh to have acceptable quality. Mesh cell skewness also affects the grid quality. Another feature which makes unstructured grid methods slightly ineffective is that in order for a cell/point to know its neighbor, it has to look it up in a list or some similar data structure, and it is somewhat difficult to order the data in such a way that needed data for an operation is in the cache. We will refer to this as *data reference locality*. Much work has been put into ordering the datasets to exploit data locality for unstructured methods in order to limit the cons of indirect addressing. Moving boundaries can be treated with ALE e.g. [14], which admits movements of the boundary small enough to limit the deformation of the cells. Larger deformations need time consuming regridding.

2.6.2 Body-fitted grids

The computational domain is covered with a grid of indexed points (x_i, y_j, z_k) which can be mapped to a rectangular box in (ξ_i, η_j, ϕ_k) computational space. Derivatives can either be evaluated in physical or in mapped space. Single such grids can only cover simple geometries, e.g. C-grids and O-grids etc., see e.g. [76] and Figure 2.6. There are regularity requirements on the grid-to-computational domain mappings, to preserve accuracy. For high-order methods and complex geometries this can be an impossible requirement. With so called multi-block meshes see Figure 2.7 complex domains can be gridded, but constructing good body-fitted grids is time consuming. The rewards for this are grids on which computations can represent the physics of the problem accurately, especially boundary layers. This property is not easily duplicated in the unstructured gridding or in the Cartesian embedded boundary methods.

The data structures do not suffer from indirect addressing and adaptive mesh refinement (AMR), e.g. [5], [4] can be used to locally refine the solution.

The simulation of moving boundaries requires regridding in each time iteration or when the boundaries move.

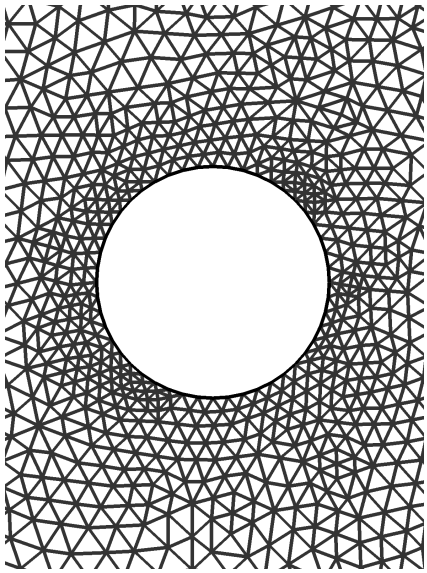


Figure 2.5. Example of an unstructured mesh around a cylinder.

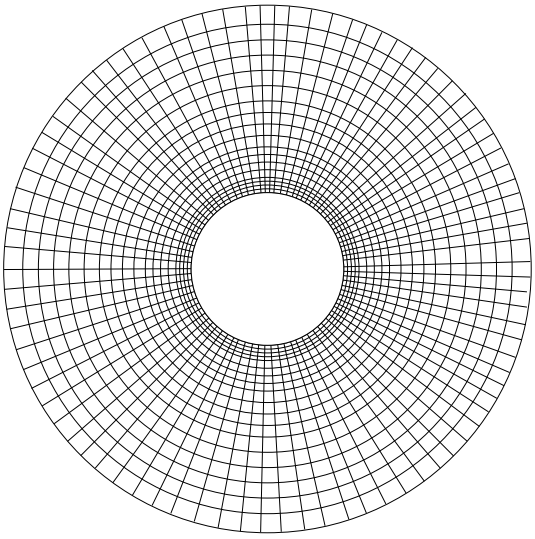


Figure 2.6. Example of body-fitted structured O-grid around a cylinder.

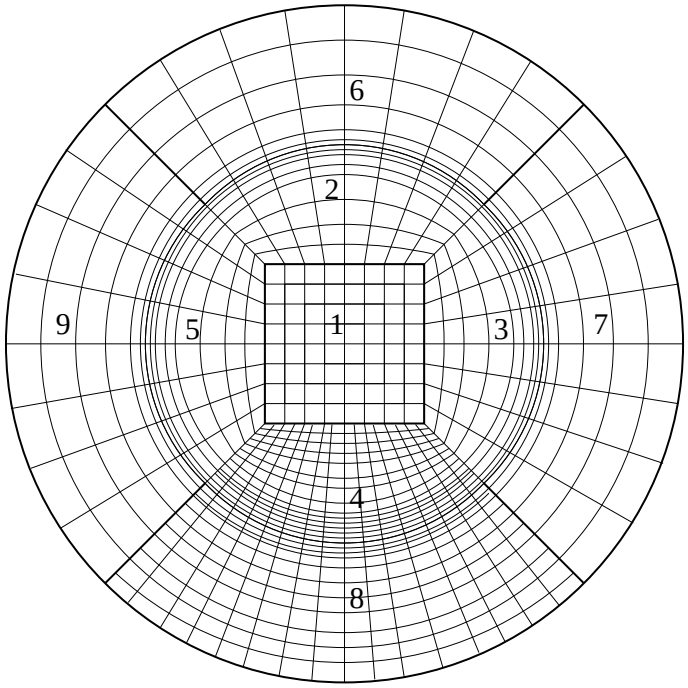


Figure 2.7. Example of a multi-block mesh of a jet. The grid is stretched somewhat towards the shear-layer located between the outer layer and the intermediate layer of grids. Note the hanging nodes in patches 1,4 and 8.

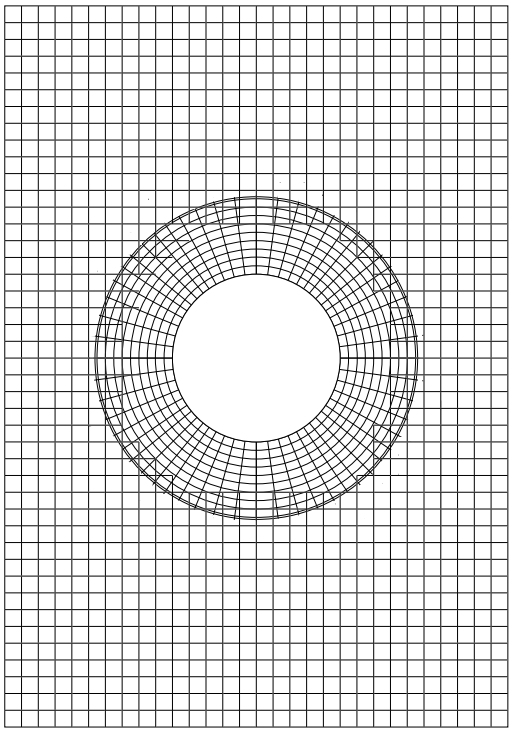


Figure 2.8. Overlapping a Cartesian mesh with a body-fitted structured mesh.

Overlapping structured grids

A very promising method to circumvent the limitations of structured grids is to use overlapping grids, also known as 'Chimera' grids e.g. [31]. A 'base' grid, covers the whole computational domain, and around objects body-fitted grids are introduced. In the overlapping domain between the 'base' grid and the body-fitted grids interpolation is used to connect the solution in all grids. A significant complexity of this method is the computation of intersection points for the interpolations between the grids.

Moving the boundary is handled by moving the mesh surrounding the boundaries only. The interpolation stencils between grids need to be updated and grid values previously unused must be assigned values in a consistent and stable manner, which is a general problem with moving meshes, for a solution see e.g. [31].

2.7 Uniform Cartesian grids with embedded boundaries

Uniform Cartesian grids have the highest quality with respect to computational accuracy, since there is no need to make any mappings from physical space to computational space. Most methods based upon Taylor-expansions are derived to be used on uniform Cartesian grids, so there is no need to make modifications to the numerical schemes due to deformed meshes. Classically, Cartesian grids could be used only for box-like geometries, where the boundaries of the object are aligned with the grid. The discretization methods do not need indirect addressing making them effective.

With embedded boundary methods e.g. [50], [13] and [49] (**Paper IV**), the use of Cartesian grids can be extended to solve problems in complex geometries, yielding an efficient and accurate method.

In a body-fitted grid, the grid spacing along the body surface can be much larger than the resolution in the surface normal direction in order to resolve boundary layers in viscous computations. This means that the cells near the body have high aspect ratios, typically 1:10 for time dependent calculations and 1:100 or even 1000 in steady state computations. While this is computationally efficient, accuracy can suffer due to the scale separation that in practice is enforced. For example, if a shock or flame hits the body at an angle to the grid lines, the effective resolution will be the coarser of the two resolutions and not the finer.

One of the former difficulties with embedded boundaries has been the so called "small-cell stiffness", caused by boundary-gridline intersecting very close to grid-points.

The boundary location is defined by signed distance function (SDF). Very complicated geometries can be generated in seconds, straight from CAD-files, see [59], by simple calls to the signed distance function. Movement of the boundary is also made simple, no regridding is required, only calls to the SDF, updating of the ghost point interpolation stencils and initialization of previously unused grid values.

2.7.1 Virtual cell embedding

The virtual cell embedding is a Cartesian gridding technique used for generating grids for very complex boundaries [50]. A complex geometry is specified as a union of a number of simples shapes. Each shape, whether given as an analytic function, a surface-panel representation or some form of bit map, must be accompanied by a subroutine that determines if a point lies inside or outside the geometry. In this method there is a distinction of cells into three categories; inner, outer and cells that are cut. The cells that are cut are further subdivided into smaller cells typically into 8^2 in 2D and 10^3 in 3D. Each one of these "subcells" is determined to be inside of the body, if the center of the "subcell" is inside the geometry. Thus this boundary

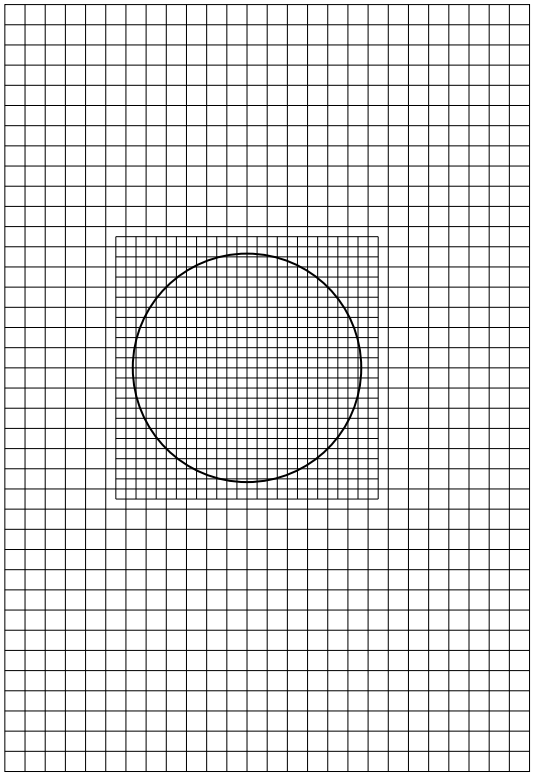


Figure 2.9. Example of a Uniform Cartesian mesh around a cylinder with local mesh refinement.

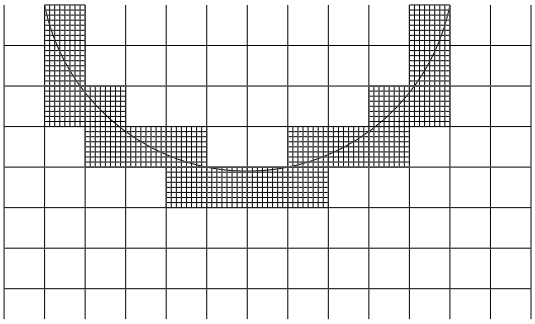


Figure 2.10. Example of Virtual Cell Embedding.

handling is in principle first order accurate, although the mesh is made much finer in the parts where needed.

2.7.2 Colellas embedded boundary method

In [13] an embedded boundary method based upon cell slicing is described for hyperbolic conservation laws. The same method to handle boundaries is used in [15] to solve the 2D compressible Navier-Stokes equations. The error for this method is $\mathcal{O}(h)$ in L_∞-norm and $\mathcal{O}(h^2)$ in L_1-norm. The method uses a linear combination of explicit conservative and non-conservative updates to set boundary values. This is combined with a flux redistribution procedure to maintain local conservation.

2.7.3 KP and SP embedded boundary methods

This work uses the Kreiss/Petersson (KP) [44, 45, 43] and SjÃ¶green/Petersson (SP) [70, 60] embedded boundary methods. They are more than first order accurate, empirically stable and do not suffer from "small-cell stiffness". They assign the ghost point values (set the boundary condition) through interpolation/extrapolation in the surface normal direction.

The KP methods are linear in the sense that the same interpolation stencil is used all the time during the calculation, whereas the SP method is slightly data-driven in the sense that it limits the slopes of the interpolants in a similar manner as is done in [70] for conservation laws. These methods are described in more detail in chapter 5.

Chapter 3

Governing Equations and Models

Here we present the LES equations with turbulence models used to model compressible flow. We discuss briefly the influence of chemical reactions on the thermodynamics of the molecular internal degrees of freedom.

WHY? there is no discussion of the reactive flow computed by FOAM?

3.1 Compressible Navier-Stokes/Euler Equations

We consider the compressible Navier-Stokes equations for a perfect gas in two and three space dimensions, which can be written as (using Einsteins' summation convention):

$$\begin{aligned}
&\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho u_j\right) = 0, \\
&\frac{\partial (\rho u_i)}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho u_i u_j + p\delta_{ij} - \left(2\mu S_{ij} - \frac{2}{3}\mu\delta_{ij}S_{kk}\right)\right) = 0, \qquad i = 1,2,3, \\
&\frac{\partial e}{\partial t} + \frac{\partial}{\partial x_j}\left((e+p)u_j - \kappa\frac{\partial T}{\partial x_j} - \left(2\mu S_{ij} - \frac{2}{3}\mu\delta_{ij}S_{kk}\right)u_i\right) = 0,
\end{aligned} \tag{3.1}$$

where ρ is the density, $u_i,\ i = 1,2,3$ is the velocity in x, y and z direction respectively, μ is the shear coefficient of viscosity and κ is the thermal conductivity coefficient. The Euler equations are obtained when $\mu = \kappa = 0$. The viscous strain rate tensor is given by

$$S_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_i} + \frac{\partial u_j}{\partial x_i}\right),$$

and the total energy per unit volume

$$e = \rho\left(\frac{R_w}{M(\gamma-1)}T + \frac{u_i u_i}{2}\right),$$

which, by using the ideal gas law constitutes the relationship

$$e = \frac{p}{\gamma - 1} + \frac{\rho}{2} u_i u_i.$$

The temperature T is given by

$$T = \frac{Mp}{R_w \rho},$$

where M is the molar mass of the substance considered and R_w is the universal gas constant. The speed of sound c, is related to the pressure and density by

$$c^2 = \gamma \frac{p}{\rho},$$

where $\gamma = \frac{C_p}{C_v}$ is the ratio between specific heats and $C_p - C_v = \frac{R_w}{M}$. Both γ and $\frac{R_w}{M}$ are constants. The specific total energy $e/\rho = \underbrace{\frac{1}{2} u_k u_k}_{e_k} + e_I = e_k + e_R + e_v$ is the sum of the kinetic and of the rotational and vibrational energies of the molecules denoted e_R and e_v respectively. The specific internal energy is denoted e_I. For a perfect gas the hypothesis are made:

- $e_v = 0$,
- the velocities at a point (x, t) satisfy a Gaussian distribution law $aexp(-b\langle u_k \rangle \langle u_k \rangle)$, where a and b and u_i are functions of (x, t). The distribution comes from the theorem of Laplace that considers the molecular velocities as identically distributed random variables when the number of particles tends to infinity.
- the specific internal energy is made up among its different components in proportion with the degrees of freedom.

The hypothesis of the equi-partition of energy is pretty well verified when there are a few degrees of freedom, for example for monoatomic molecules (He), diatomic molecules (H_2, N_2, O_2) or rigid molecules $(H_20, CO_2, C_2H_2, C_2H_4)$. The more complex molecules are less rigid; they thus have more degrees of freedom, which are not equivalent from the energetic point of view.

Let β be the number of non-translational degrees of freedom. The hypothesis of equi-partition gives the following formula for each type of internal energy:

$$e_{k_1} = \ldots = e_{k_d} = \frac{1}{d} e_k, \qquad e_R = \frac{\beta}{d} e_k,$$

and thus $e_I = (d + \beta) e_{k_1}$.

The adiabatic exponent $\gamma = \frac{d+\beta+2}{d+\beta}$ results in the law of perfect gases

$$p = (\gamma - 1) \rho e_I.$$

The most common adiabatic exponents are 5/3 and 7/5 for $d = 3$ and 5/3 and 2 for $d = 2$ and 3 for $d = 1$. In applications air is considered to be a perfect gas for which $\gamma = 7/5$.

The Prandtl number Pr relates thermal conductivity to viscosity

$$Pr = \frac{\mu R_u \gamma}{\kappa(\gamma - 1)M}.$$

Many normalizations are possible. We let $\mathcal{L}$ be a typical length and let the ∞ subscript denote typical values of the independent variables, which in our applications are the free stream values. The physical quantities are denoted by a '$*$' superscript. In three space dimensions we have

$$t = \frac{t^* c_\infty}{\mathcal{L}}, \quad x_i = \frac{x_i^*}{\mathcal{L}}, \quad u_i = \frac{u_i^*}{c_\infty}, \quad i = 1, 2, 3$$
$$\rho = \frac{\rho^*}{\rho_\infty}, \quad p = \frac{p^*}{\rho_\infty c_\infty^2}, \quad e = \frac{e^*}{\rho_\infty c_\infty^2},$$
$$\mu = \frac{\mu^*}{\mu_\infty}, \quad \kappa = \frac{\kappa^*}{\kappa_\infty}$$
$$Re = \frac{\rho_\infty c_\infty \mathcal{L}}{\mu_\infty}, \quad Pr = \frac{C_p \mu_\infty}{\kappa_\infty},$$

where Re is the Reynolds number. The dimensionless Navier-Stokes equations are

$$\begin{aligned}
&\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j}(\rho u_j) = 0, \\
&\frac{\partial(\rho u_i)}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho u_i u_j + p\delta_{ij} - \frac{\alpha(T)}{Re}\left(2S_{ij} - \frac{2}{3}\delta_{ij}S_{kk}\right)\right) = 0, \quad i = 1, 2, 3, \\
&\frac{\partial e}{\partial t} + \frac{\partial}{\partial x_j}\left((e + p)u_j - \frac{\alpha(T)\gamma}{RePr(\gamma - 1)}\frac{\partial}{\partial x_j}T - \frac{\alpha(T)}{Re}\left(2S_{ij} - \frac{2}{3}\delta_{ij}S_{kk}\right)u_i\right) = 0,
\end{aligned} \tag{3.2}$$

where $T = p/\rho$. The physical viscosity $\mu^* = \alpha(T)\mu_\infty = \alpha(T)\frac{\rho_\infty c_\infty \mathcal{L}}{Re}$ and the physical thermal conductivity $\kappa^* = \frac{\alpha(T)\gamma R_w \rho_\infty c_\infty \mathcal{L}}{MRePr(\gamma - 1)}$ are either taken constant ($\alpha \equiv 1$) or calculated using Sutherland's law

$$\alpha(T) = \left(\frac{T}{T_\infty}\right)^{1.5} \frac{T_\infty + S_1}{T + S_1}.$$

with $S_1 = 110.4K$. In planar flow all derivatives with respect to x_3 are set to zero and $u_3 \equiv 0$.

3.1.1 Boundary Conditions

We use boundary conditions of the form:

$$\text{wall (no-slip):} \begin{cases} u_i = 0, \ (i = 1, 2, 3) \\ \beta T + \alpha \frac{\partial T}{\partial n} = f_T \\ p \text{ :extrapolated} \end{cases} \qquad \text{wall (slip):} \begin{cases} u_i n_i = 0, \ (i = 1, 2, 3) \\ \beta T + \alpha \frac{\partial T}{\partial n} = f_T \\ p \text{ :extrapolated} \end{cases}$$

(3.3)

where n_i denote the i:th component of the normal vector. For an adiabatic wall $\alpha = 1$, $\beta = 0$ and $f_T = 0$ and for a thermal wall $\alpha = 0$ and $\beta = 1$.

At inflow and outflow boundaries at the edge of the computational domain, we impose characteristic boundary conditions e.g. [34].

Chapter 4

LES and Sub-GridScale Models

In this work we consider the Smagorinsky sub-grid model and an implicit sub-grid model. 'Implicit' subgrid models (ILES and MILES) are given by the leading order term of the truncation error of the numerical scheme. It is argued in the ILES/MILES community see e.g. [61] that there are four fortunate circumstances that make turbulence simulations and in particular ILES/MILES work:

- **The shape of the Kolmogorov spectrum:**
  The average kinetic energy decreases fast enough for the scales containing most of the energy to be resolvable current computing hardware. This means that there is enough energy for the small scales to mix large-scale inhomogeneities as fast as the large-scale flows can produce them.

- **Energy transfer through local interactions:**
  Turbulent energy is transferred by a turbulent cascade that passes from large eddies to the small scales where it is eventually dissipated. Simulations have shown that the energy transfer in the inertial range is dominated by local (in wavenumber space) interactions, and not deposited directly from the large, energy-containing scales into the small scales.

  The relatively smooth transfer of scales implies that there is a portion of the inertial sub-range (see Figure 2.2) where the behavior of the fluid dynamics is essentially scale invariant. This is the reason why the region with -5/3 spectrum slope is the acceptable place to match a subgrid model to the resolved-scale model. In the ILES/MILES context, this corresponds to choosing the grid size in this region.

- **Dynamics on the large scales:**
  There is an apparent lack of important dynamics occurring at scales even a factor of ten or more larger than the classically defined Kolmogorov scale. The dissipation at scales larger than the Kolmogorov scale is sufficiently strong

that little structure survives to reach the small scales. In practice this reduces the need for spatial resolution.

- **Behavior of monotone algorithms at the grid-scale cut-off:** Monotone schemes have a modified equation whose leading order truncation error resembles the local nonlinear dissipation that connects the large, resolved, energy-containing scales to the unresolved subgrid scales and provide a built-in measure of the dissipation required.

One important difference between LES and ILES/MILES concerns the computational mesh. In LES, one emphasizes the independence of the equations, particularity that of the explicit subgrid scale models from the mesh. However, in ILES/MILES, the mesh is analogous to an experimental apparatus; there one expects the simulation results to depend on the choice of mesh. Although at first glance this may seem a philosophical point, it has important practical implications. In particular, the implicit subgrid scale models of ILES depend both on the length scales and the geometry of the computational mesh.

4.0.2 The Smagorinsky subgrid model

The Smagorinsky subgrid model dates back to the 1960's, [73]. If we solve the Navier-Stokes equations $u_t + f(u)_x = 1/Re u_{xx}$, then the Smagorinsky model can be seen as the method of explicitly added artificial viscosity:

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = \frac{\partial}{\partial x}\left(\left(\frac{1}{Re} + \underbrace{\varepsilon\left|\frac{\partial u}{\partial x}\right|}_{Artificial}\right)\frac{\partial u}{\partial x}\right), \tag{4.1}$$

where ε is the Smagorinsky model parameter. This simple example does not show the physical reasoning behind the model, but as a corresponding one dimensional example illustrating the actual effect of the model this is precisely what the model is, which is easily seen from the full LES-Smagorinsky equations in d dimensions:

$$\begin{aligned}
&\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_j}(\rho u_j) = 0 \\
&\frac{\partial \rho u_i}{\partial t} + \frac{\partial}{\partial x_j}\left\{\rho u_i u_j + \delta_{ij} p - \left(\frac{\alpha(T)}{Re} + C_s^2 \rho \Delta^2 \|S\|\right)\left(2S_{ij} - \frac{2\delta_{ij}}{3} S_{kk}\right) + \right. \\
&\qquad \left. \frac{2\delta_{ij}}{3} C_I \rho \Delta^2 \|S\|^2\right\} = 0, \quad i = 1, \ldots, d \\
&\frac{\partial e}{\partial t} + \frac{\partial}{\partial x_j}\left\{(e+p)u_j - \left(\frac{\alpha(T)\gamma}{RePr(\gamma-1)} + \frac{\gamma R}{\gamma - 1} C_s \frac{\Delta^2 \rho \|S\|}{Pr_T}\right)\frac{\partial}{\partial x_j}(T) - \right. \\
&\qquad \left. \left(2S_{ij} - \frac{2\delta_{ij}}{3} S_{kk}\right) u_j\right\} = 0,
\end{aligned} \tag{4.2}$$

where Δ is the filter width parameter and the parameters $C_s = 0.16$ and $C_I = 0.09$ according to [52] and $\|S\|$ is the Frobenius norm of the strain rate tensor, which is $\sim |u_x|$ in (4.1). Realizability conditions on the model parameters, e.g for positive semi-definiteness of the SGS stress tensor, are given in [78]:

$$C_I \geq \frac{\sqrt{3}}{2} C_s^2.$$

Pr_T is the turbulent Prandtl number usually taken to be in the interval $Pr_T \in [0.7,\ 0.9]$. For wall-bounded flows the most used approach is to turn off the LES contribution near the wall resulting is wall-resolved LES. Δ is given by

$$\Delta(x, y, z)a = \min(\Delta, C(\text{wallDist}(x, y, z))) \tag{4.3}$$

where $C(\text{wallDist}(x, y, z))$ turns off the subgrid model close to a solid wall and Δ is a measure of the grid spacing and $\text{wallDist}(x, y, z)$ is the closest distance to any wall. Δ can be computed in many ways e.g.

$$(\Delta x \Delta y \Delta z)^{1/3}, \quad \max(\Delta x, \Delta y, \Delta z), \quad \min(\Delta x, \Delta y, \Delta z), \quad (\text{cell volume})^{1/3}.$$

For Cartesian grid methods $\Delta = h$, where h is the (uniform) cell width, is usually chosen.

4.0.3 The implicit subgrid model (ILES)

In implicit LES (ILES) and Monotone Integrated LES (MILES) methods the truncation error of the method acts as LES turbulence model. Monotonicity is a property of certain schemes for conservation laws [51] using flux limiting and artificial viscosity. The local effect of the non-linear flux-limiting is interpreted as an effective subgrid turbulence model. Since the truncation error is not viewed separate from the modeling, these methods are at least second order accurate for solving their modified equation, i.e. the equation a certain scheme solves to $\mathcal{O}(h^p)$ including the leading order error term. There are empirical indications that ILES/MILES works e.g. [29], [27] and [26]. In many studies e.g. [48] (**Paper III**), [22] (**Paper II**) ILES/MILES is one of many LES subgrid models tested. A common conclusion is that the solution is not much affected by the choice of subgrid model, indicating either that the solution algorithm has poor accuracy, so modeling effects do not show in the solution, or that the spatial resolution was insufficient.

4.0.4 Convergence

A high-Reynolds number flow is unstable, even when the full turbulent cascade is being resolved down to the Kolmogorov scale. Two faithful simulations of the same problem with different methods will deviate progressively with time as a result of arbitrarily small differences in initial and/or boundary conditions, round off errors, methods, etc.. Now, convergence of a sequence of numerical simulations means that

the solutions approach a limit, the grid-converged solution, which by the procedure becomes defined point-wise. If the methods are convergent under grid refinement, as is customarily assumed, one could compare the *limiting solutions*.

Turbulent flows are chaotic by nature and convergence cannot be point-wise in space and time, but as in the definition of Hinze, see Ch x, only in averaged quantities. It follows that the numerical simulations will not converge.

How to assess the accuracy of LES using $\Delta = C \cdot h$, i.e. how to define the limiting solution, is perplexing; increase in spatial resolution uncovers more structure in the computed flow as unresolved scales become resolved. One might define convergence to the limit solution as *convergence of all relevant averages*. and address the problem of selection of these.

An alternative is to use a *fixed* Δ, specific for the problem under study, dictated by the physics and the quantities of interest: forces, mixing properties, etc.. As an example, a physical length scale can be provided by the "equivalent grain size" in flows over rough walls. [47] (**Paper V**) uses this interpretation to analyze the performance of the EB boundary conditions.

4.0.5 Near-wall-boundary Conditions for LES

Like all numerical approximations to the flow equations LES requires the setting of boundary conditions in order to fully determine the system and obtain a mathematically well-posed problem. Here we discuss questions of determining suitable boundary conditions for LES especially near solid walls.

General problem

The LES equations can be of an order different from that of the original Navier-Stokes equations. This is trivially verified by considering the differential interpretations of the filters: the resolved equations are obtained by applying a differential operator of arbitrarily high order to the basic equations.

The changed degree of the discretized equations raises the problem of determining the associated boundary conditions, because those associated with the equations governing the evolution of the exact solution can no longer be used in theory for obtaining a well-posed problem [66]. This problem is generally not considered, arguing the fact that the higher-order terms appear only in the form of $\mathcal{O}(\Delta^p),\ p \geq 1$ perturbations of the Navier-Stokes equations and the same boundary conditions are used for the LES and DNS of the Navier-Stokes equations. Moreover, when the filter is unknown, it is no longer to derive suitable boundary conditions strictly, which also leads to the use of the boundary conditions of the basic problem.

The boundary conditions, along with the similarity parameters of the equations determine the solution. These conditions represent the whole fluid domain beyond the computational domain. To specify the solution completely, these conditions must apply to all of its scales, i.e. to all space-time modes it comprises.

In order to characterize a particular flow, the amount of information in the boundary conditions ia a function of the number of degrees of freedom of the boundary condition system. This poses the problem of representing a particular solution, in order to represent it numerically. We have a new modeling problem; to model the physical test configuration.

This difficulty is increased for LES and DNS, due to the large number of degrees of freedom and require a precise space-time deterministic representation of the solution at the computational domain boundaries.

Solid wall problem

We describe the problem in the ideal framework of a flat-plate, turbulent boundary layer, without pressure gradient. The external flow is in the x_1 direction and the x_2 direction is normal to the wall. The external velocity is denoted U_e. The boundary layer thickness δ is defined as the distance from the plate beyond which the fluid becomes irrotational, and thus where the fluid velocity is equal to the external velocity. The friction velocity u_τ is defined as

$$u_\tau = \sqrt{\frac{\tau_w}{\rho}}, \tag{4.4}$$

where $\tau_w = \frac{\partial u_{tangential}}{\partial n}$ and we can define a Reynolds number

$$Re_\tau = \frac{\delta \rho u_\tau}{\mu}. \tag{4.5}$$

The reduced velocity u_i^+, expressed in wall units, is defined as $u_i^+ = u_i/u_\tau$. The wall coordinates x_i^+ are obtained by the transformation $x_i^+ = \frac{x_i \rho u_\tau}{\mu}$.

The boundary layer is divided into two parts: the inner region ($0 \leq x_2 \leq 0.2\delta$) and the outer region ($0.2\delta \leq x_2$). In the inner region the dynamics is dominated by the viscous effects. In the outer region, it is controlled by the turbulence.

The structure of the (turbulent) boundary layer flow has certain characteristics:

- Low-speed streaks in the region $0 < y^+ \leq 10$. The flow is highly agitated close to the wall, consisting of pockets of fast and slow fluid that organize in ribbons parallel to the outer velocity.
- Ejections of low-speed pockets fluid outward from the wall. These are subject to an instability that make the explode near the outer edge of the inner region.
- An ejection is followed by sweeps of high-speed fluid toward the wall in the near-wall region, almost parallel to it.
- Vortical structures of several proposed forms.
- Strong internal shear layers in the wall zone ($y^+ \leq 80$).

- Near-wall pockets, observed as areas clear of marked fluid in certain types of flow visualizations.
- Backs: surfaces (of scale δ, where δ is the boundary layer thickness) across which the streamwise velocity changes abruptly.
- Large-scale motion in the outer layers (including bulges, superlayers and deep valleys of free-stream fluid).

These highly intermittent events induce a strong variation in the unsteady Reynolds stresses and originate a very large part of the production and dissipation of the turbulent kinetic energy. These variations produce fluctuations in the subgrid dissipation that can reach 300 % of the average value and can make it change sign, [66]. The above features call for a special treatment in the framework of LES. Analyzes of DNS result [66] indicate that the maximum turbulent energy production is at $x_2^+ \approx 15$, which gives rise to a high backward energy cascade and associated with the sweeping type events. The forward cascade is associated with the ejections.

In the outer regions of the boundary layer where the viscous effects no longer dominate the dynamics, the energy cascade mechanism is predominant. Both cascade mechanisms are associated preferentially with the ejections.

Modeling or resolving the near-wall behavior in LES?

The problems of applying the LES framework to the flow near walls are that the mechanisms creating the turbulence, i.e. the driving mechanisms, are associated with fixed characteristic length scales on the average. Also, the turbulence production is is associated with a backward cascade mechanism in certain regions of the boundary layer. These two factors make the present subgrid models inoperative because they cannot represent these driving mechanisms. There are two approaches to this dilemma:

- Resolving the near-wall dynamics directly. The subgrid models do not account for the turbulence production mechanisms in the near-wall region. By turning off the subgrid model in the near-wall region and using a sufficiently fine resolution the near-wall dynamics are resolved. This is called wall-resolved LES, and is illustrated in Figure 4.1. The solid wall is represented by a no-slip condition. This implicitly implies that the mean free path of the molecules is small compared to the characteristic scales of the motion, and that these scales are large compared with the distance of the first grid point from the wall. In actual simulations, this is achieved by placing the first grid point at $x_2^+ \in [0, 1]$. In [24, 67] a typical required resolution for boundary layers using DNS/LES is reported: Arguing that the non-isotropic modes must be resolved it is argued in [3] that number of degrees of freedom in space for the solution scales as Re_τ^2. This becomes prohibitively expensive from a computational perspective for high-Reynolds number flows. In [46] (**Paper VI**), we have experimentally made investigations how to the turn off the LES SGS-model

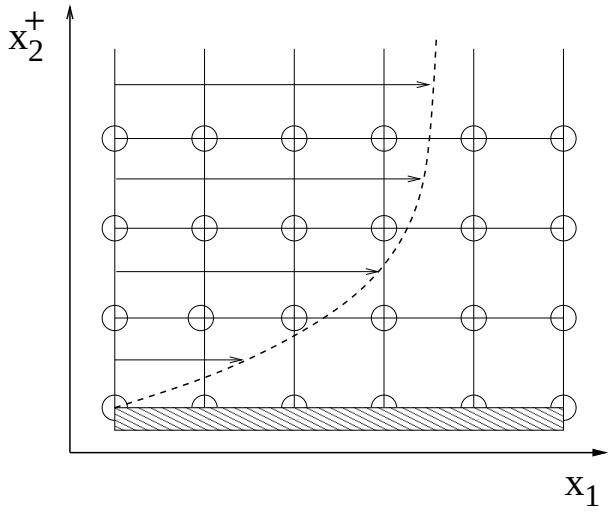


Figure 4.1. No-slip type grid.

	DNS	Wall-resolving LES	LES with wall-model
Δx_1^+ (streamwise)	10-15	20-50	100-600
Δx_2^+ (wall-normal)	1	1	30-150
Δx_2^+ (spanwise)	5	10-20	100-300
No. of points in $0 < x_2^+ < 10$	3	3	-

Table 4.1. Typical mesh size (in wall units) for DNS and LES for boundary layer flow, [67].

contribution close to the wall. However these simulations were too unresolved in the near-wall region, so that nothing conclusive can be said.

- Modeling the near-wall dynamics. The inner region is a model representing the dynamics of the zone between the wall and the first grid point outside the wall, see Figure 4.2. This is a special subgrid model called a wall-model. Since, usually the first grid point is located at a distance greater than the characteristic length scales of the modes existing in the modeled region, the no-slip condition can no longer be used. The boundary condition will apply to the values of the velocity components and/or their gradients, which will be provided by the wall-model. This approach makes it possible to place the first grid point in the logarithmic layer (in practice $x_2^+ \in [20, 200]$). The main advantage is that the number of degrees of freedom in the simulation is greatly reduced, but since part of the dynamics is modeled, an additional source of error is also introduced.

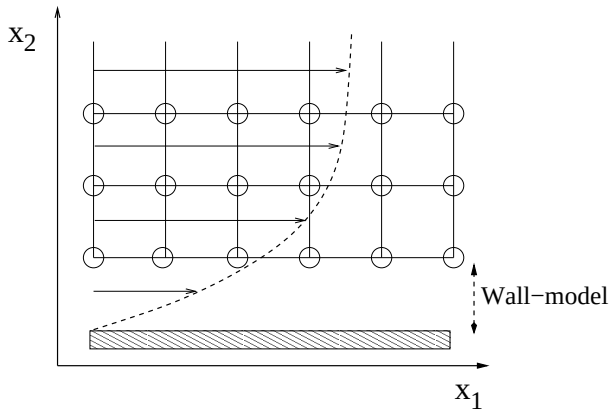


Figure 4.2. Wall-model type grid.

A wall-model for compressible LES

For incompressible flows there are numerous wall-models proposed for LES, see e.g. [66] for a comprehensive summary. In [55] a wall-model for the incompressible LES equations is presented. The method is based on a local integration from the first computational grid point near the wall to the surface. This approach does not require the use of an internal discretization of the boundary layer or the use of an auxiliary solver. Therefore it can be implemented into an existing flow solver. Below we present this wall-model adapted to compressible flow and the embedded boundary method used in this thesis.

We assume (for simplicity of presentation) that the tangential velocity is u_1 and that the normal direction from the wall is x_2. The wall-model is given as a solution to

$$\frac{d}{dx_2}\left((\mu+\mu_T)\frac{du_1}{dx_2}\right) = F, \tag{4.6}$$

where F can be the pressure gradient and/or convective terms in the NS/LES equations.

$$\mu_T(x_2) = \rho(x_2)L_m^2(x_2)\left|\frac{\partial u_1}{\partial x_2}(x_2)\right|.$$

The mixing length L_m is given by

$$L_m(x_2) = \kappa x_2\left(1-\exp\left(-\frac{x_2^+}{26}\right),\right)$$

where $\kappa = 0.42$ is the von Karman constant. The model is derived by integrating (4.6) in the wall-normal direction to yield:

$$\frac{du^+}{dx_2^+} = \begin{cases} \frac{2(1+F^+x_2^+)}{\frac{\mu_I}{\mu_w}+\sqrt{\frac{\mu_I^2}{\mu_w^2}+4\frac{\rho_I}{\rho_w}L_m^2(x_2^+)(1+F^+x_2^+)}} & \text{if } 1+F^+x_2^+ \geq 0 \\ \frac{2(1+F^+x_2^+)}{\frac{\mu_I}{\mu_w}+\sqrt{\frac{\mu_I^2}{\mu_w^2}-4\frac{\rho_I}{\rho_w}L_m^2(x_2^+)(1+F^+x_2^+)}} & \text{if } 1+F^+x_2^+ < 0, \end{cases} \tag{4.7}$$

where the $\cdot_I$-subscript denotes values at x_I and $\cdot_w$ denotes values on the wall boundary and $F^+ = F\frac{\mu}{\rho^2 u_\tau^3}$. The velocity profile is given by

$$u^+\left(x_2^+(u_\tau), F^+(u_\tau)\right) = \int_0^{x_2^+} \frac{du^+}{d\xi} d\xi.$$

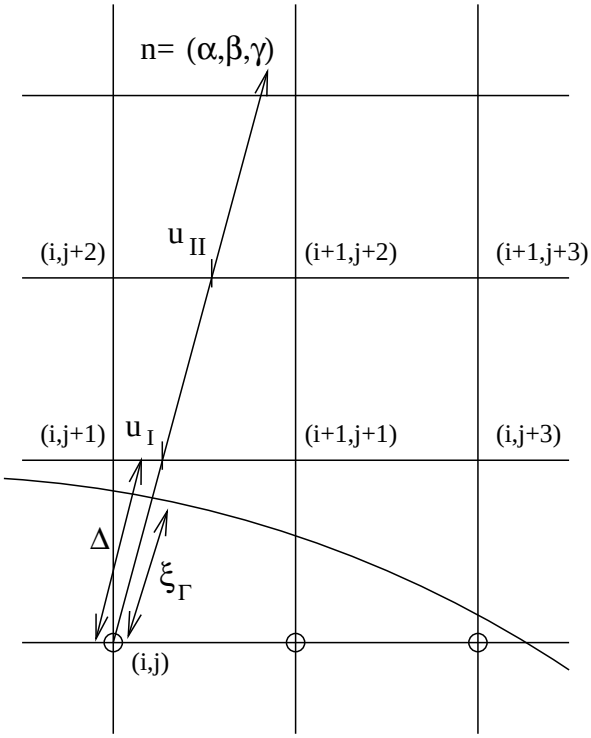


Figure 4.3.

In order to solve for the unknown friction velocity u_τ we have that $x_2 = \Delta - \xi_\Gamma \Rightarrow x_2^+ = \frac{(\Delta-\xi_\Gamma)u_\tau}{\mu\kappa}$

$$G(u_\tau) = u_\tau u^+\left(x_2^+(u_\tau), F^+(u_\tau)\right) - u_I = 0, \tag{4.8}$$

where u_I is the *known* velocity at x_I. The relation (4.8) can be solved for numerically with the Secant method:

$$u_\tau^{n+1} = u_\tau^n - \frac{u_\tau^n - u_\tau^{n-1}}{G(u_\tau^n) - G(u_\tau^{n-1})} G(u_\tau^n).$$

This of course needs two initial guesses to work.

The definition of friction velocity is:

$$u_\tau = \sqrt{\frac{\mu_w \left|\frac{\partial u}{\partial x_2}\right|_{x_2=0}}{\rho_w}}$$

Once the friction velocity is known, we get a Neumann boundary condition for the tangential velocity from

$$\left(\frac{\partial u}{\partial x_2}\right)_{x_2=0} = \frac{\rho_w u_\tau^2}{\mu_w} \tag{4.9}$$

where the u_τ is the *known* friction velocity. After this the system needs to be transformed back to the original coordinates.

In the embedded boundary technique the boundary can intersect the grid in an arbitrary way. Specially there will be computational points really close to the boundary, which are not considered boundary points. This means that we wish to investigate the effect of the wall boundary condition when $x_2^+ \to 0$. Taking the limit shows that there is no u_τ dependence on $\frac{du^+}{dx_2^+}$, which means that for small enough x_2^+ and arbitrary u_I there is no parameter to be solved for, and hence the procedure breaks down. This means that we are in the viscous sublayer in the boundary layer and we can switch into using the condition

$$u^+ = x_2^+$$

instead, which is used to get $u_\tau = \frac{\mu\kappa u_I}{\Delta - \xi_\Gamma}$.

Comment

One should note that the above boundary condition is *not* to be used for the convective part of the operator, but rather only to the viscous operator. In effect we have performed a splitting of the operators. The original equations are split from

$$\partial_t u + \partial_{x_i}(f(u)) + \partial_{x_i}(f_v(u)) = 0 \quad \text{with no-slip condition on velocity.}$$

into

$$\begin{aligned} \partial_t u + \partial_{x_i}(f(u)) &= 0 \quad \text{with slip condition on velocity i.e. } \vec{u}\cdot\vec{n} = 0 \\ \partial_{x_i}(f_v(u)) &= 0 \quad \text{with Neumann condition (4.9)} \\ &\qquad\quad \text{on tangential velocities.} \end{aligned}$$

Chapter 5

A Cartesian Embedded Boundary Method for Compressible Flow

This chapter presents the numerical methods used in this thesis and describes two Cartesian Embedded boundary methods for the compressible Navier-Stokes/LES equations; the Kreiss/Petersson (KP) [44, 45, 43] method and the SjÃ¶green/Petersson (SP) [70, 60] method. The presentation will be done for the two dimensional case, to simplify notation. The methods have been generalized to three space dimensions and are described in [49] (**Paper IV**). We also limit this description to the KP3 method, but we have investigated KP4 and KP5, which are higher order boundary procedures, but these are not described here.

5.1 FOAMs compressible solver suite

The following section is included for completeness of description of the methods used during this doctoral thesis work. The methods in FOAM for compressible flows are implicit up to second order accurate methods with a Courant-number restriction of ≈ 0.3 (determined experimentally).

Lately the time-integrator in FOAM has been re-implemented to using Runge-Kutta. Results using this solver is presented in [6] for the forward facing step problem [80] and simple Burgers equation.

Recently there has been work done in the direction of well-known methods that work [23] for hyperbolic conservation laws, showing remarkable improvement in both accuracy and effectivity compared to previously existing FOAM implementations for compressible flow.

The solvers used in this theses have produced good results for axisymmetric baseflow [22] (**Paper II**), when compared to experimental data [32], but nothing conclusive could be said regarding choice of subgrid model or numerical scheme, since results are nearly indistinguishable in [22] (**Paper II**) and [48] (**Paper III**).

The conclusion drawn was that FOAM, at least in its former status, was a too dull instrument to assess the question of how good LES really is and/or what subgrid model strategy to use.

5.2 Description of the methods

Below we describe the internal discretization scheme used to approximate solutions to the Navier-Stokes/LES equations.

5.2.1 Solving the Riemann problem

In the one-dimensional case, the particular problem when initial data u_0 is piecewise constant with a single jump discontinuity at some point x_0,

$$u_0(x) = \begin{cases} u_L, & \text{if } x < x_0 \\ u_R, & \text{if } x > x_0 \end{cases} \tag{5.1}$$

is called the *Riemann problem*. The solution of this problem has a central role in the construction of numerical methods for hyperbolic conservation laws. For hyperbolic problems the solution to the Riemann problem is typically a similarity solution, a function of x/t alone, and consists of a finite set of waves that propagate away from x_0 with constant wave speeds. For linear hyperbolic systems, $u_t + Au_x = 0$, the Riemann problem is easily solved in terms of the eigenvalues and eigenvectors of the matrix A, see e.g. [51]. This simple structure also holds for nonlinear systems of equations and the exact solution (or arbitrarily good approximations) to the Riemann problem can be constructed, described below.

Numerical methods are usually derived using Taylor series to establish appropriate expressions for the values of the dependent variables at the next time level. Differences in the spatial direction are also based upon the requirement of having certain accuracy using a series expansion. Taylor-series work very well when conditions for convergence of the series are met. In fact, the series will converge everywhere, provided the function that is approximated is sufficiently smooth. In the case of a finite-difference method, we assume that a series expansion is an appropriate means of obtaining a difference approximation and the functions are continuous and have continuous derivatives at least through the order of the difference approximation. This is certainly not the case when shock waves (in the inviscid case or unresolved sharp gradients in the viscous case) or other discontinuities are present. Godunov [25] proposed a solution to this problem by avoiding the differentiability requirement by using a finite-volume approximation in solving the conservation equations and evaluating the flux terms at the cell interfaces by the solution of a Riemann problem. The state variables are assumed to be constant in control volumes, which is sufficient for first order schemes.

The Euler equations in 1D are:

$$\begin{aligned} \rho_t + (\rho u)_x &= 0 && (5.2)\\ (\rho u)_t + (\rho u^2 + p(\rho, e_I)) &= 0 && (5.3)\\ \left(\rho\left(e_I + \frac{1}{2}u^2\right)\right)_t + \left(\rho\left(e_I + \frac{1}{2}u^2\right)u + pu\right)_x &= 0. && (5.4) \end{aligned}$$

The velocity u can take any real values, but ρ and e_I must be positive. The following description is made easier in the non-conservative form:

$$\begin{aligned} \rho_t + u\rho_x + \rho u_x &= 0 && (5.5)\\ u_t + uu_x + \rho^{-1}p_x &= 0 && (5.6)\\ (e_I)_t + u(e_I)_x + \rho^{-1}pu_x &= 0. && (5.7) \end{aligned}$$

The matrix of this system is

$$A = uI_{3\times 3} + \begin{pmatrix} 0 & \rho & 0 \\ \rho^{-1}p_\rho & 0 & \rho^{-1}p_{e_I} \\ 0 & \rho^{-1}p & 0 \end{pmatrix}.$$

The eigenvalues are solutions to $(\lambda-u)^3 = (\lambda-u)(p_\rho + \rho^{-2}pp_{e_I})$. In the form (5.5) we see that the system has a singularity all over the plane $\rho = 0$. This corresponds to the fact that, when $\rho = 0$, the conservative variables $(\rho, \rho u, e)$ are not independent of each other, since they are all zero together, resulting in a singular point $(0, 0, 0)$. The density being zero on an interval expresses the fact that this interval is free of gas. We cannot exclude this state in the solution of the Riemann problem, which introduces an indeterminacy in the variables which describe the flow. It is clear that vacuum has zero density, energy and pressure, but on the other hand velocity is not defined ($'0/0'$), which prevents giving sense to the energy flux. This issue is not pursued further here and we shall assume that $\rho > 0$.

The system is hyperbolic if and only if $p_\rho + \rho^{-2}pp_{e_I} > 0$, which we in the following will assume.

The eigenvectors and eigenvalues of A are expressed as functions of the speed of sound $c = \sqrt{p_\rho + \rho^{-2}pp_{e_I}}$:

$$\lambda_1 = u - c, \quad \lambda_2 = u, \quad \lambda_3 = u + c,$$

$$r_1 = \begin{pmatrix} -\rho \\ c \\ -\rho^{-1}p \end{pmatrix}, \quad r_2 = \begin{pmatrix} p_{e_I} \\ 0 \\ -p_\rho \end{pmatrix}, \quad r_3 = \begin{pmatrix} \rho \\ c \\ \rho^{-1}p \end{pmatrix}.$$

We have that $d\lambda_j \cdot r_j = c + \rho^{-2}p(\rho c)_{e_I} = \frac{1}{2}(\gamma+1)c > 0,\ j = 1, 3$, i.e. the first and third fields are of the same nature and genuinely non-linear. The speed of sound $c = \gamma(\gamma-1)e_I = \frac{\gamma p}{\rho}$. The second field is always linearily degenerate $d\lambda_2 \cdot r_2 = 0$. The Riemann-invariants for the 1-waves are $(S, u - \frac{2c}{\gamma-1})$, for the 2-waves (u, p) and for the 3-waves $(S, u + \frac{2c}{\gamma-1})$.

Solution to the Riemann problem

Given a state to the left (p_L, u_L, ρ_L) and a state to the right (P_R, u_R, ρ_R) the solution to the Riemann problem is to seek two intermediate states, indicated by subscripts 1 and 2, and three waves linking these four states. The central wave being a contact discontinuity (since 2-wave is linearily degenerate), we have $p^* = p_1 = p_2$ and $u^* = u_1 = u_2$. There is a 1-wave from (p_L, u_L, ρ_L) to (p^*, u^*, ρ_1) and a 3-wave from (p^*, u^*, ρ_2) to (p_R, u_R, ρ_R). This results in

$$\begin{aligned} \rho_1 &= \sigma(p^*; p_L, \rho_L), && (5.8)\\ u^* &= u_L - \tau(p^*; p_L, \rho_L), && (5.9)\\ \rho_2 &= \sigma(p^*; p_R, \rho_R), && (5.10)\\ u^* &= u_R + \tau(p^*; p_R, \rho_R), && (5.11) \end{aligned}$$

where

$$\tau(p^*; p_-, \rho_-) = \begin{cases} (p^* - p_-)\sqrt{\frac{\rho_-}{2}\left((\gamma+1)p^* - (\gamma-1)p_-\right)} & \text{if } p^* > p_-, \\ \frac{2c_-}{\gamma-1}\left(\left(\frac{p^*}{p_-}\right)^{\frac{\gamma-1}{2\gamma}} - 1\right) & \text{if } p^* \le p_-. \end{cases}$$

We cancel u^* from the calculations by observing

$$u_L - u_R = \tau(p^*; p_L, \rho_L) + \tau(p^*; p_R, \rho_R),$$

which is a scalar non-linear equation for the unknown p^*. Once this is solved, (5.8) and (5.10) yield values of ρ_1 and ρ_2. Finally u^* is given by either of (5.9) or (5.11). The σ function is not presented due to ease of notation.

The Godunov method

The solution to the Riemann is used in the flux computation in the Godunov method. In the method the Riemann problem is formulated at all cell interfaces $(x_{i+1/2})$ and integrated in space and time $[x_i, x_{i+1}] \times [t_n, t_n + \Delta t_n]$, over which the flux is computed.

Comment

Solving the Riemann problem is computationally expensive, especially in many dimensions (strictly speaking it is difficult to generalize the concept of Riemann problems from one dimension; we mean dimension-by-dimension splitted Riemann problems). Also the exact solution of the Riemann problem is seldom needed (one example is when computing very strong shock waves). Therefore approximate Riemann solvers were developed.

5.2.2 Roe Approximate Riemann Solver For the Euler equations

The Roe method [65] can be seen as a generalization of the upwind scheme to systems of equations. For a scalar conservation law $(u_t + f(u)_x = 0)$ the numerical flux would be

$$h^n_{j+1/2} = \frac{1}{2}\left(f_{j+1} + f_j\right) - \frac{1}{2}|a_{j+1/2}|\left(u^n_{j+1} - u^n_j\right).$$

For systems $(\mathbf{u}_t + \mathbf{F}(\mathbf{u})_x = 0)$ the local wave speed(s)

$$a_{j+1/2} = \begin{cases} \frac{f(u_{j+1})-f(u_j)}{u_{j+1}-u_j}, & u_j \neq u_{j+1}, \\ f'(u_j), & u_j = u_{j+1}, \end{cases} \tag{5.12}$$

is generalized using the eigenvalues of a Jacobian matrix. A matrix

$$A_{j+1/2} = A(\mathbf{u}_j, \mathbf{u}_{j+1}),$$

with $A(\mathbf{u}, \mathbf{u}) = A(\mathbf{u}) = \partial\mathbf{F}/\partial\mathbf{u}$ is defined, and the scheme becomes

$$\mathbf{h}^n_{j+1} = \frac{1}{2}\left(\mathbf{F}_{j+1} + \mathbf{F}_j\right) - \frac{1}{2}|A_{j+1/2}|\left(\mathbf{u}^n_{j+1} - \mathbf{u}^n_j\right),$$

where the absolute value of the matrix is defined as

$$|A| = R|\Lambda|R^{-1}, \qquad |\Lambda| = DAG(|\lambda_1|, |\lambda_2|, \ldots, |\lambda_d|),$$

for a system with d equations. Here λ_j are the eigenvalues and R is the matrix with the eigenvectors as columns. This can be viewed upon as a local diagonalization of the system. The matrix can be computed in many ways e.g. $A_{j+1/2} = A\left(\frac{1}{2}(\mathbf{u}_j + \mathbf{u}_{j+1})\right)$ or $A_{j+1/2} = \frac{1}{2}\left(A(\mathbf{u}_j) + A(\mathbf{u}_{j+1})\right)$, but Roe [65] required that it should satisfy the following condition

$$\mathbf{F}(\mathbf{u}_{j+1}) - \mathbf{F}(\mathbf{u}_j) = A_{j+1/2}(\mathbf{u}_{j+1} - \mathbf{u}_j),$$

which is a generalization of (5.12) in the scalar case. This means that the Rankine-Hugoniot conditions will be correct i.e. shocks are treated correctly, but rarefaction waves are not seen. This issue will be addressed later in the section considering entropy fixes. The Euler equations are obtained by letting $Re \to \infty$ in (3.2), i.e. $\mu = \kappa = 0$ in (3.1). For the Euler equations in one dimension ($d = 3$) with the equation of state

$$e = \frac{p}{\gamma - 1} + \frac{\rho}{2}\left(u_i u_i\right),$$

the Roe matrix is obtained by evaluation of the Jacobian at a weighted average state $\hat{\mathbf{u}}$:

$$A_{j+1/2} = A(\hat{\mathbf{u}}) = A(\theta\tilde{\mathbf{u}}_j + (1-\theta)\tilde{\mathbf{u}}_{j+1}), \quad \theta = \frac{\sqrt{\rho_j}}{\sqrt{\rho_j} + \sqrt{\rho_{j+1}}},$$

where $\tilde{\mathbf{u}} = [\rho, u, h]^T$, where $h = \frac{e+p}{\rho}$ is the enthalpy. The numerical flux for the Roe method for the Euler equations may be written as

$$\mathbf{h}_{j+1/2} = \frac{1}{2}\left((\mathbf{F}_{j+1} - \mathbf{F}_j - \sum_{k=1}^{3} \alpha_{k,j+1/2} |\lambda_{k,j+1/2}| \mathbf{r}_{k,j+1/2}), \right)$$

where $\alpha_{k,j+1/2}$ are the wave strengths obtained by solving $R\alpha = \mathbf{u}_{j+1} - \mathbf{u}_j$. The wave speeds, i.e., the eigenvalues of A are

$$\hat{\lambda_1} = \hat{u} - \hat{c}, \quad \hat{\lambda_2} = \hat{u}, \quad \hat{\lambda_3} = \hat{u} + \hat{c},$$

where $\hat{u}$ is the normal velocity and $\hat{c}$ is the speed of sound.

Comment

The difference scheme given by Roe is found by solving the linearized equations. For some choices of initial data the scheme becomes unstable, even though a solution exists, i.e. certain Riemann problems are not linearizable [18]. Given $u_L = (\rho, -rhou, e_I)$ and $u_R = (\rho, rhou, e_I)$ three cases can be distinguished:

- $\frac{4\gamma\rho e_I}{3\gamma-1} - \rho^2 u^2 \leq 0$ vacuum occurs in the solution.
- $\frac{4\gamma\rho e_I}{3\gamma-1} - \rho^2 u^2 > 0$ and $(\gamma-1)\rho e_I - \rho^2 u^2 \leq 0$, the problem has a solution with positive density and internal energy, but is not linearizable.
- $\frac{4\gamma\rho e_I}{3\gamma-1} - \rho^2 u^2 > 0$, the problem has a solution with positive density and internal energy and is linearizable.

The above shows that for certain special cases the approximate Riemann solvers might fail.

5.2.3 Artificial viscosity and entropy fix

In practice, one cannot use the wave speeds as given above; close to regions where $\hat{u} = 0$, e.g. stagnation points, the linear eigenvalues $\hat{\lambda_2}$ approaches zero, and near sonic lines the non-linear eigenvalues $\hat{\lambda_1}$, $\hat{\lambda_3}$ approach zero. In these cases the numerical flux cannot break up entropy violating shock waves into rarefaction waves and an entropy fix is needed to produce physically relevant solutions.

Viewing this from another point of view and recalling the similarity between the upwind method and the Roe method, one says that a scheme is Total Variation

Diminishing or TVD (strictly a property governed by scalar conservation laws), when

$$\sum_{i=-\infty}^{\infty} |u_{i+1}^{n+1} - u_i^{n+1}| \leq \sum_{i=-\infty}^{\infty} |u_{i+1}^{n} - u_i^{n}| \tag{5.13}$$

is fulfilled. A scheme with the numerical flux function given in the viscosity form

$$h_{j+1/2} = h(u_j, u_{j+1}) = \frac{1}{2}\left(f_i + f_{i+1} - \lambda^{-1} Q_{j+1/2}(u_{j+1} - u_j),\right)$$

where $\lambda = \frac{\Delta x}{\Delta t}$, is TVD if and only if

$$\lambda |a_{j+1/2}| \leq Q_{j+1/2} \leq 1. \tag{5.14}$$

The Lax-Wendroff scheme has $Q_{j+1/2} = \lambda^2 a_{j+1/2}$ and is not TVD, the Lax-Friedrichs scheme has $Q_{j+1/2} = 1$, which is the upper TVD-limit, and hence very diffusive. The *upwind* scheme has

$$Q_{j+1/2} = \lambda |a_{j+1/2}|,$$

which is the lower-TVD in (5.14). The upwind scheme does not satisfy the entropy condition, since it does not contain enough artificial viscosity to break expansion shocks into expansion waves. The sensitive points are the u-values for which $f'(u) = 0$, i.e., for points where the numerical viscosity vanishes. It is possible to eliminate this kind of entropy violation by simply modifying the viscosity coefficient $Q_{j+1/2} = Q(\lambda a_{j+1/2})$ near $a_{j+1/2} = 0$ so that it is positive. There are many possibilities of choosing the entropy fix, some are shown below in (5.15)

$$Q(x) = \begin{cases} (i) & \begin{cases} \frac{x^2}{2\varepsilon} + \varepsilon, & \text{for } |x| < 2\varepsilon \\ |x|, & \text{else} \end{cases} \\ (ii) & \sqrt{x^2 + \varepsilon^2}, \quad 0 < \varepsilon < 1 \\ (iii) & |x| + \varepsilon \\ (iv) & \max(|x|, \varepsilon) \end{cases} \tag{5.15}$$

where ε is a parameter given by the user. The choice of ε is problem dependent and experience has shown that for a specific problem when really fine grids are used ε needs to be increased slightly compared to the value used on a coarser grid. For steady state computations when iterative solvers that make use of the Jacobian of the solution are used (i) is preferably used (Harten entropy fix [30]), since it makes the viscosity coefficient a C^1 function of its arguments. Disadvantages of the methods are (i) does not vectorize, i.e. an if-then-else statement must be done in the innermost loop, taking the square-root in (ii) is computationally expensive, in (iii) viscosity is added everywhere (even where not needed) and (iv) is not C^1. In the computations made we have used (iv) because it vectorizes and we do not use any iterative solvers.

Artificial viscosity for systems

We enforce entropy by having non-zero artificial viscosity. In the following a choice between using local eigenvalues or using the maximum eigenvalue over all points/cells can be made. In our computations we have chosen to use local eigenvalues. The spectral radius of A is denoted $\rho(A) = |u| + c$. In our computations we use the entropy fix (iv) in (5.15), which for the three-dimensional problem is $(\hat{\lambda_1^d} = \hat{u}_d - \hat{c}, \quad \hat{\lambda_{2,3,4}^d} = \hat{u}_d, \quad \hat{\lambda_5^d} = \hat{u}_d + \hat{c})$ using local eigenvalues

$$\begin{aligned} |\tilde{\lambda}_1^d| &= \max\left\{|\hat{\lambda}_1^d|, \varepsilon_n^d \Phi_n^d\right\}, \\ |\tilde{\lambda}_{2,3,4}^d| &= \max\left\{|\hat{\lambda}_{2,3,4}^d|, \varepsilon_l^d \Phi_l^d\right\}, \qquad d = 1,2,3, \\ |\tilde{\lambda}_5^d| &= \max\left\{|\hat{\lambda}_5^d|, \varepsilon_n^d \Phi_n^d\right\}, \end{aligned} \tag{5.16}$$

where ε_l and ε_n are parameters to be chosen by the user. The subscripts 'l' and 'n' denote linear and non-linear eigenvalues respectively. The Φ_l and Φ_n are functions of the absolute values of the eigenvalues of the flux Jacobians used to scale the $\tilde{\lambda}$:s in (5.16) so that the CFL condition is not violated. In our computations the following choice was used:

$$\Phi_l^d = \Phi_n^d = |u_d| + c. \tag{5.17}$$

Choice of ε_l and ε_n depend mainly on the grid (if grid is deformed, stretched etc.), the geometry (corners, singularities, bluff-body etc.) and the type of flow (supersonic, hypersonic etc.).

5.2.4 MUSCL scheme

The numerical flux for the first order Roe method is

$$\mathbf{h}_{j+1}^n = \mathbf{h}(\mathbf{u}_i, \mathbf{u}_{i+1}) = \frac{1}{2}\left(\mathbf{F}_{j+1} + \mathbf{F}_j\right) - \frac{1}{2}|A_{j+1/2}|\left(\mathbf{u}_{j+1}^n - \mathbf{u}_j^n\right).$$

The first order scheme is generalized to second order by using limited piecewise linear reconstruction in the flux:

$$\mathbf{h}_{j+1}^n = \mathbf{h}(\mathbf{u}_i + \frac{1}{2}\mathbf{s}_i, \mathbf{u}_{i+1} - \frac{1}{2}\mathbf{s}_{i+1}), \tag{5.18}$$

where s_i are the slopes of the piecewise linear reconstruction. The slopes will introduce new extrema in the solution, violating the TVD property (5.13) and some sort of limiting must be done using e.g. the minmod limiter:

$$\mathbf{s}_i = minmod(\Delta_+\mathbf{u}_i, \Delta_-\mathbf{u}_i) = \begin{cases} 0 & \text{if } \Delta_+\mathbf{u}_i\Delta_-\mathbf{u}_i < 0 \\ sign(\Delta_+\mathbf{u}_i)\min(\Delta_+\mathbf{u}_i, \Delta_-\mathbf{u}_i) & \text{else,} \end{cases}.$$

The width of the numerical stencil increases from three to five points.

5.2.5 Approximating the strain rate tensor in Navier-Stokes equations

A finite difference approximation of the space derivatives in Navier-Stokes equations can be written as

$$\frac{d}{dt}\mathbf{u}_{i_1,i_2,i_3} + \sum_{s=1}^{d} \frac{\mathbf{h}_{i_s+1/2} - \mathbf{h}_{i_s-1/2}}{\Delta x_s} = \sum_{s=1}^{d} \frac{(\mathbf{g}_v^{x_s})_{i_s+1/2} - (\mathbf{g}_v^{x_s})_{i_s-1/2}}{\Delta x_s}$$

The viscous fluxes $(\mathbf{g}_v^x)_{i_1-1/2,i_2,i_3}, (\mathbf{g}_v^y)_{i_1,i_2-1/2,i_3}$, and $(\mathbf{g}_v^z)_{i_1,i_2,i_3-1/2}$ contain first derivatives. For example, the x-direction viscous fluxes are, with the temporary notation of half-integer subscripts, $f_{im} = \frac{f_i+f_{i-1}}{2}$,

$$(\mathbf{g}_v^x)_{i_1 m,i_2,i_3} = \begin{pmatrix} 0 \\ \frac{4\mu}{3} D_-^x u_{i_1,i_2,i_3} - \frac{2\mu_{i_1 m,i_2,i_3}}{3}(D_0^y v_{i_1 m,i_2,i_3} - D_0^z w_{i_1 m,i_2,i_3}) \\ \mu_{i_1 m,i_2,i_3}(D_0^y u_{i_1 m,i_2,i_3} + D_-^x v_{i_1,i_2,i_3}) \\ \mu_{i_1 m,i_2,i_3}(D_0^z u_{i_1 m,i_2,i_3} + D_-^x w_{i_1,i_2,i_3}) \\ f_5 + k_{i_1 m,i_2,i_3} D_-^x T_{i_1,i_2,i_3} \end{pmatrix}.$$

where

$$f_5 = u_{i_1 m,i_2,i_3}(\mathbf{g}_{v,2}^x)_{i_1-1/2,i_2,i_3} + v_{i_1 m,i_2,i_3}(\mathbf{g}_{v,3}^x)_{i_1-1/2,i_2,i_3} + w_{i_1 m,i_2,i_3}(\mathbf{g}_{v,4}^x)_{i_1-1/2,i_2,i_3},$$

This means that second derivatives are approximated by standard finite difference formulas e.g.

$$u_{xx}(x_{i_1}, y_{i_2}, z_{i_3}) = D_+^x D_-^x u_{i_1,i_2,i_3} + \mathcal{O}(h^2) = \frac{u_{i_1+1,i_2,i_3} - 2u_{i_1,i_2,i_3} + u_{i_1-1,i_2,i_3}}{h^2} + \mathcal{O}(h^2) \tag{5.19}$$

and

$$u_{xy}(x_{i_1}, y_{i_2}, z_{i_3}) = D_0^x D_0^y u_{i_1,i_2,i_3} + \mathcal{O}(h^2) = D_+^x \frac{D_0^y u_{i_1,i_2,i_3} + D_0^y u_{i_1-1,i_2,i_3}}{2} + \mathcal{O}(h^2). \tag{5.20}$$

5.2.6 Wavelet sensor for detecting singularities

[72, 81] describes an adaptive low-dissipation scheme based on high-order central differences. It uses multi-resolution wavelet analysis to post-process the solution after each time step to pinpoint where numerical dissipation needs to be added. Below we describe the wavelet analysis technique, since it is useful in general to detect discontinuities/singularities in the solution and it is used in [49] (**Paper IV**) to control the order of an interpolant in grid to grid interpolation, since it is

well-known that high-order polynomial interpolation of non-smooth data gives rise to unwanted over- and undershoots.

The technique is based upon estimating the Hölder exponent α

$$|f(x) - P(x - x_0)| \leq C|x - x_0|^{\alpha}. \tag{5.21}$$

of the discrete solution $f(x)$. $P(x)$ is a polynomial. For precise mathematical details see e.g. [20].

The estimated α is fed into a switch $\tau(\alpha)$, which decides if any action needs to be taken. In [72] several switches are investigated and the following switch yields satisfactory results:

$$\tau(\alpha) = \begin{cases} 1, & \text{if } \alpha \leq \alpha_0 \text{ (take action)} \\ 0, & \text{else,} \end{cases} \tag{5.22}$$

where α_0 is chosen to be 0.5.

A Demonstration

Below we demonstrate the need to detect where the solution is non-smooth in application to grid-to-grid interpolation. In our example a jump discontinuity is being transported from left to right, the two outermost grid point values on the embedded grid need interpolation from the underlying grid. When applying the higher order interpolation routine with the discontinuity is inside interpolation stencil we observe the well known over- and undershoots, see Figure 5.1. When using the wavelet-indicator, the order of the interpolant is lowered point-wise when $\tau = 1$. This approach gives no over- or undershoots as shown in Figure 5.1

5.3 KP and SP embedded boundary methods

Among the greatest challenges of computational fluid dynamics is arguably the accurate prediction of flows in the vicinity of complex geometries. Traditionally, this problem is tackled by discretizing the governing equations on unstructured or structured body-fitted curvilinear grids, causing the geometric boundaries to coincide with those of the computational domain. The main advantage of these methods is the relatively straight-forward implementation of boundary conditions. However, the difficulty of representing complex geometries in the computational domain is now being transferred to generating adequate grids and keeping track of neighboring grid points (logistics problem) and merging simpler grid-blocks (connectivity problem). Furthermore, if moving boundaries are present, frequent regeneration and merging of grids will be necessary, weighing heavily on computational cost and maintenance of grid quality. In addition, highly distorted grids reduce the accuracy of a numerical method. In fact, constructing a high-order method for such grids is extraordinarily difficult. These problems hamper the development of solution algorithms for flows in complex geometries.

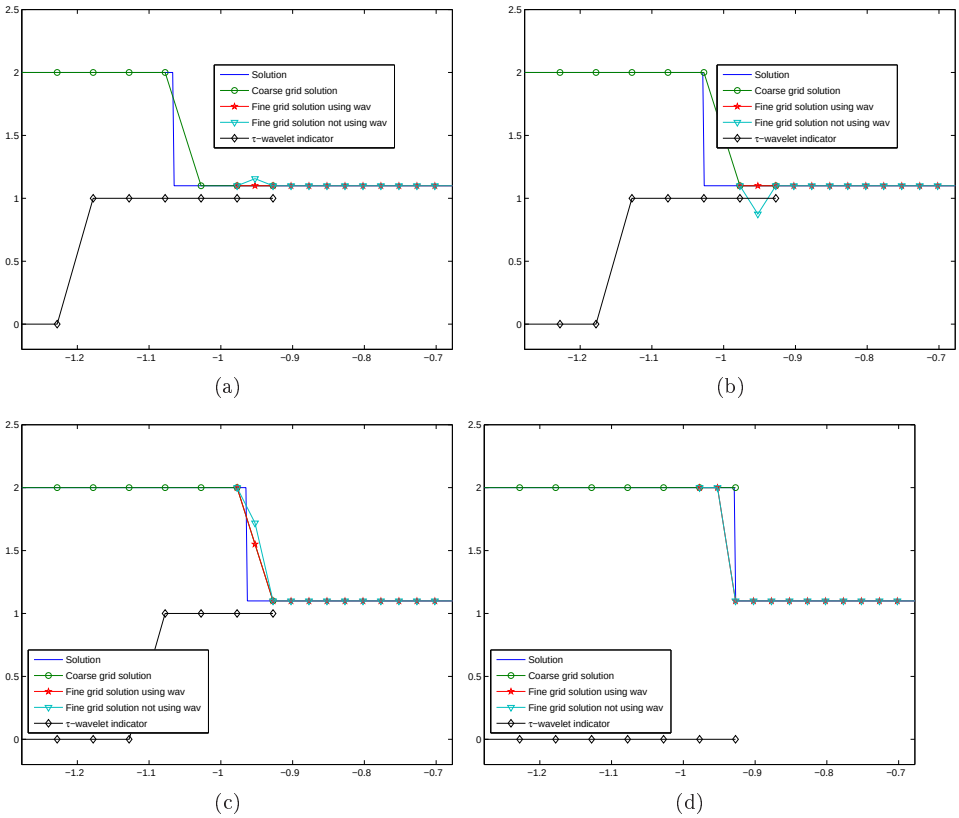


Figure 5.1. The grid-grid interface is located at $x = 1$. The two left-most fine grid point values need to be interpolated from the coarse grid. At locations where $\tau = 1$, the order of interpolation is lowered. Notice the over- and undershoots of the high-order interpolant.

The approach presented here is a fixed Cartesian grid method. As a consequence, geometric boundaries can not always conform to computational domain boundaries, thus forming *embedded boundaries*, whose presence has to be accounted for in another way. Furthermore, boundary conditions at the embedded boundary are not necessarily enforced at grid points. The domain Ω is covered by a Cartesian grid with step size h, where the grid points are located at $(x_i, y_j) = (ih, jh)$, and the boundary Γ is allowed to cut through the grid in an arbitrary manner, see Figure 5.2. The use of Cartesian grids greatly simplifies grid generation and implementation of high-order methods[1] The separation of the embedded boundary and the computational grid removes the need for re-meshing strategies when dealing with moving boundaries. An appropriate Cartesian grid method also enables efficient code parallelization by taking advantage of the structured nature and time-independence of the underlying grid.

The obvious complication associated with Cartesian grid methods is the implementation of appropriate boundary conditions at embedded boundaries and their subsequent representation on the computational grid. Below we describe the approach we have chosen. This is the main contribution of this thesis.

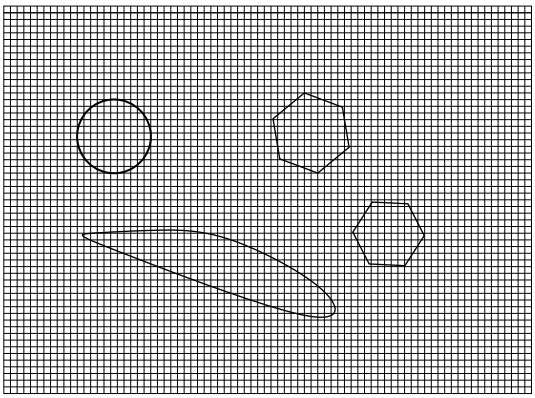


Figure 5.2. The boundary is allowed to cut the grid in an arbitrary manner in the embedded boundary method.

5.3.1 KP Embedded boundary method

The KP embedded boundary method was originally developed for the second order wave equation [45, 43, 44]. In the following we present how the method was extended for solving the compressible Navier Stokes equations [49, 47, 46] (**Paper IV,V,VI**).

[1] High-order refers to the formal order of accuracy of the numerical scheme without the presence of embedded boundaries.

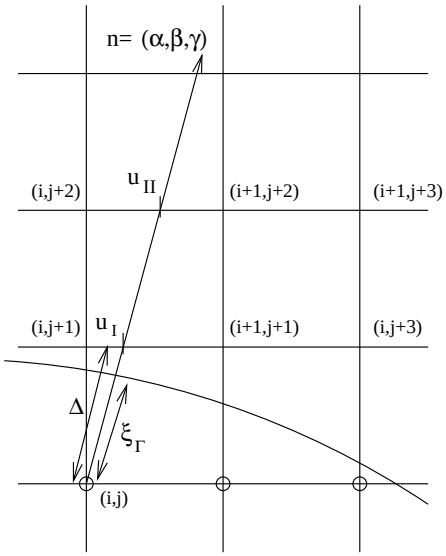


Figure 5.3. KP embedded boundary method. The indices denote the domain of dependence for the embedded boundary procedure for ghost point (i,j).

To evaluate the differencing operator associated with Navier Stokes equations at all grid points inside the computational domain, we use ghost points "just" outside the domain, see Figure 5.3. We construct a Lagrange interpolation between three points along the normal: $(0, \phi_{i,j})$, (ξ_I, ϕ_I) and (ξ_{II}, ϕ_{II}) to aid in the approximation of the Dirichlet boundary condition. The ϕ_I and ϕ_{II} are approximated by Lagrangian interpolation along grid lines y_{j+1} and y_{j+2}. For the case illustrated $\xi_{II} = 2\xi_I = \left|\frac{\beta}{\alpha}\right|$, where α and β are the x and y components of the normalized normal. These formulas hold when the angle θ between the x-axis and the normal satisfies $\theta \in [\pi/4, \pi/2]$. In the KP method, [43], the expressions in the remaining three quadrants are simply obtained by reflections in index space, leading to a total of 8 different cases to treat all possible directions of the boundary in two space dimensions. In three dimensions there are 24 different cases. The author of this thesis has included four additional (in 2D) interpolation stencils to make the interpolation more continuous when changing stencil, all stencils are shown in Figure 5.4. In 3D the number of interpolation stencils are increased to 38. For higher order accurate boundary approximations more points have to be included in the approximation.

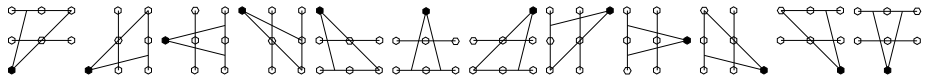


Figure 5.4. All different interpolation stencils in 2D for the KP3 embedded boundary method. The ghost point is marked black and the bounds of the normal is shown.

We define

$$L_h u \equiv g_0(\xi)u_0 + g_I(\xi)u_I + g_{II}(\xi)u_{II} \tag{5.23}$$

The boundary may intersect the grid so that ξ_Γ is arbitrarily close to ξ_I, i.e., $g_0(\xi_\Gamma) = \frac{(\xi_\Gamma - \xi_I)(\xi_\Gamma - \xi_{II})}{\xi_I \xi_{II}}$ can be arbitrarily close to zero. This is referred to as the small-cell stiffness problem, due to the finite volume analogy of cutting cells to arbitrarily small sizes, yielding an arbitrarily small cell affecting the ($\Delta t \sim C\Delta x$) CFL condition see. Hence, if naive Lagrange interpolation would be used to approximate the boundary condition, the time-stepping would become very stiff. An artificial term is added to the Lagrange interpolant to mitigate the stiffness

$$B_h\phi(t) \equiv L_h\phi(t) + \eta(\phi_{i,j} - 2\phi_I + \phi_{II}) = g_D(x_\Gamma, y_\Gamma, t), \tag{5.24}$$

where the constant $\eta > 0$ and g_D is the Dirichlet value on the boundary. The artificial term is an undivided second difference in the normal direction, so it inflicts an $\mathcal{O}(h^2)$ error in the boundary condition approximation.
[43] shows that the truncation error in the boundary condition can oscillate wildly between consecutive grid points along the boundary and can degrade the rate of convergence.

The artificial term η in the boundary condition bounds the coefficient in front of $\phi_{i,j}$ away from zero, since

$$\eta \leq g_0(\xi_\Gamma) + \eta < 1 + \eta,$$

and hence the small cell stiffness problem is removed. The spectrum of the one-dimensional convection-diffusion equation has been estimated, to yield that the eigenvalue with the largest magnitude is independent of small cells near the boundary when $\eta \geq 0.25$. For all our numerical computations (using KP3) we have used $\eta = 0.25$, which works well in practice. It is big enough to allow time steps independent of the small cells near the boundary, and small enough to prevent the artificial term from dominating the error in the numerical solution.

The Neumann Problem

For Neumann conditions, e.g. when there is an adiabatic wall, the formulas of the previous section change somewhat. In [45] this procedure is described for the

second order wave equation and we present the major ideas from that paper here, since they are identical to what we use. The Neumann condition is

$$\frac{\partial \phi}{\partial n} = g_N(x, y, t), \quad (x, y) \in \Gamma, \quad t > 0. \tag{5.25}$$

The Lagrange interpolant (without added artificial error) is differentiated to yield a second order approximation of the (outward) normal derivative. It is shown in [45] that the Neumann problem does not suffer from the small cell stiffness problem, since $g_0'(\xi) = \frac{2\xi - \xi_I - \xi_{II}}{\xi_I \xi_{II}}$ is bounded away from zero in $\xi \in [0, \xi_I + \xi_{II})$.

5.3.2 SP Embedded boundary method

When discontinuities are present in the solution, special care has to be taken to make the boundary interpolation robust. The SP method is of lower accuracy, but more suited for dealing with shock waves, and is outlined in Figure 5.6. It uses more values along the normal than the KP method, but the tangential interpolation is linear. When the normal has positive y-component and the angle between the normal and the x-axis is between $\frac{\pi}{4}$ and $\frac{\pi}{2}$, the normal will always intersect the grid line $y = y_{j+1}$ between x_i and x_{i+1}. There are two different cases when the normal intersects the $y = y_{j+2}$ grid line (between x_i and x_{i+1} or between x_{i+1} and x_{i+2}) and similarly three different cases where the normal intersects the $y = y_j + 3$ grid line. A quarter of all interpolation stencils for the SP method are shown in Figure 5.5

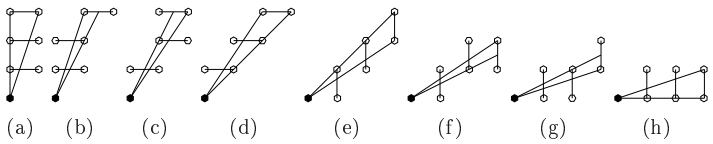


Figure 5.5. A quarter of all different interpolation stencils in 2D for the SP embedded boundary method. The ghost point is marked black and the bounds of the normal is shown.

Denote the distance between the boundary and the ghost point by b and let the distance between the ghost point and grid line $y = y_{j+1}$ along the normal be Δ (see Fig 5.6). Define new points u_{b_1} and u_{b_2} placed equidistantly along the normal by linear interpolation along the normal at distances $b + \Delta$ and $b + 2\Delta$ from the ghost point respectively,

$$u_{b_1} = \frac{b}{\Delta} u_{II} + \left(1 - \frac{b}{\Delta}\right) u_I, \quad u_{b_2} = \frac{b}{\Delta} u_{III} + \left(1 - \frac{b}{\Delta}\right) u_{II}.$$

A limited boundary slope is defined,

$$s_D := S_{minmod}\left(u_{b_1} - g_D, u_{b_2} - u_{b_1}\right),$$

where

$$S_{minmod}(x,y) = \begin{cases} x, & \text{if } |x| < |y| \text{ and } xy > 0, \\ y, & \text{if } |y| < |x| \text{ and } xy > 0, \\ 0, & \text{otherwise} \end{cases} \tag{5.26}$$

is the well-known min-mod limiter. The Dirichlet boundary condition is approximated by extrapolation using the limited boundary slope,

$$u_{i,j} = g_D - \frac{\xi_\Gamma}{\Delta} s_D. \tag{5.27}$$

The above construction is always well-defined, since $h \leq \Delta \leq \sqrt{d}h$, where d is the number of space dimensions. The Neumann boundary condition is imposed by:

$$u_{i,j} = \left(\frac{4}{3} - \frac{\xi_\Gamma}{3\Delta}\right) u_{b_1} - \left(\frac{1}{3} - \frac{\xi_\Gamma}{3\Delta}\right) u_{b_2} - \frac{2\Delta(2\xi_\Gamma + 1)}{3} g_N \tag{5.28}$$

Numerical boundary conditions are imposed using extrapolation

$$u_{i,j} = u_I - S_{minmod}\left(u_{III} - u_{II}, u_{II} - u_I\right). \tag{5.29}$$

The above numerical boundary condition is equivalent to setting $\frac{\partial^2 u}{\partial n^2} = 0$, if the solution is smooth enough not to trigger the limiter. Otherwise it is equivalent to the first order approximation of $\frac{\partial u}{\partial n} = 0$.

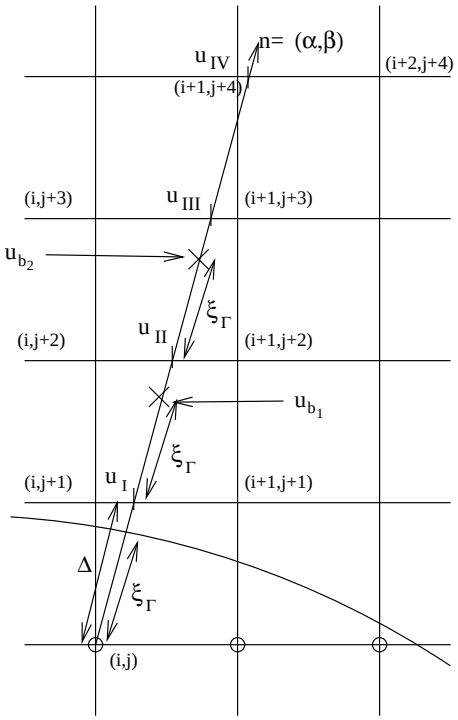


Figure 5.6. SP embedded boundary method. The indices denote the domain of dependence for the embedded boundary procedure for ghost point (i,j).

5.3.3 Comments

For parallel execution special care needs to be taken for the ghost points whose interpolation stencils extend outside the local processor's domain. For objects that extend outside the grid, then alternatively the interpolation stencil is made smaller, so that it will fit inside the grid, or values from an underlying coarser grid are used. If the object extends outside the computational domain and values cannot be retrieved from any grid, then the assignment of ghost point values can use the exterior boundary condition. This has however not been implemented or tested.

5.3.4 Treatment of corners and thin bodies

Sharp corners demand special treatment. This approach generalizes to thin bodies, i.e. embedded objects that are only one grid point wide in some direction.

For ghost-points with more than one normal to the surface, we overload l solution values $\mathbf{u}_m$, $m = 1, \ldots, l$, each associated with a unique surface normal $\mathbf{n}_m \neq \mathbf{n}_n$, $m \neq n$. Each $\mathbf{u}_m$ is updated using the KP method as described in e.g. [49] (**Paper IV**).

The difference stencil uses a value associated with the correct surface normal with the direction of the using point.

There are interior points where associated surface normals are not well-defined. For example, the difference stencil for point $\mathbf{x}_\Gamma$ uses $\mathbf{x}_{i,j,k}$ see Figure 5.7. We can associate two boundary values to $\mathbf{x}_{i,j,k}$: $\mathbf{u}_1$ and $\mathbf{u}_2$, and use a weighted average of $\mathbf{u}_1$ and $\mathbf{u}_2$.

A difference stencil at $\mathbf{x}_\Gamma$, which uses $\mathbf{x}_{i,j,k}$ which, in here, does not have a unique normal, performs averaging:

$$\mathbf{u}^\Gamma_{i,j,k} = \frac{d_1}{d_1+d_2+\ldots+d_l}\mathbf{u_1} + \frac{d_2}{d_1+d_2+\ldots+d_l}\mathbf{u_2} + \ldots + \frac{d_l}{d_1+d_2+\ldots+d_l}\mathbf{u_l}, \tag{5.30}$$

where $d_m = \max((\mathbf{x}_\Gamma - \mathbf{x}_{i,j,k})\cdot\mathbf{n}_m, 0)$, $m = 1,2,\ldots,l$. The averaging is only performed if two or more $d_m > 0$, i.e. the point is in the averaging zone, see Figure 5.7.

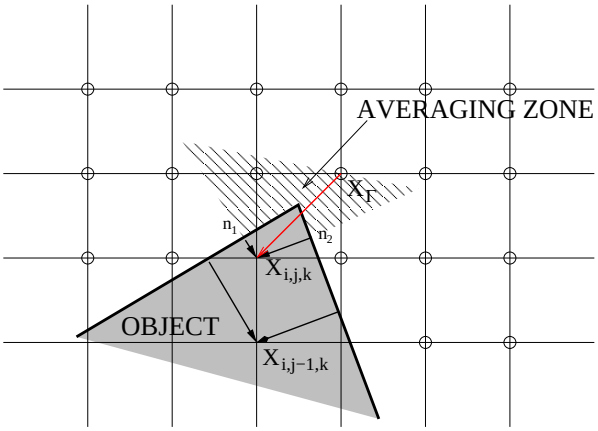


Figure 5.7. Corner points are averaged: $l = 2$.

The average (5.30) is a convex combination of the ghost-point values $\mathbf{u_m}$. Note that the above procedure can be used for thin bodies, since the $x_{i,j,k}$:s associated with the 'backside' surface will have $d_m < 0$. The values from point $x_{i,j-1,k}$ in Figure 5.7 are never averaged. In the case of a 'concave' corner we average the values of the nearby ghost points, see Figure 5.8

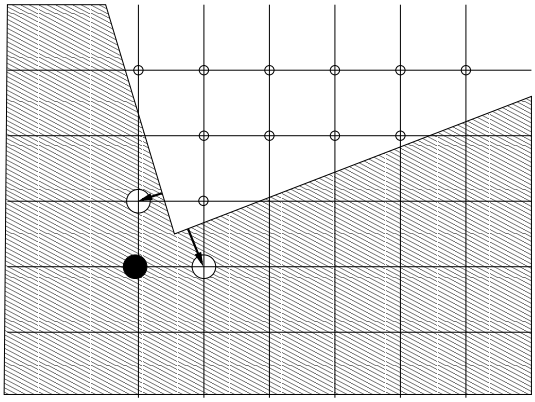


Figure 5.8. The black ghost point does not have a normal associated with any surface, but its value is needed in the discretization. The black ghost point is assigned the average value of the two white ghost points.

5.3.5 Higher Order of Accuracy

In general, higher-order methods are better than lower order methods. The higher order methods can only yield better answers for scale lengths that are longer than a few cells. Interpolation, no matter how high the order, also cannot reproduce unresolved variations occurring within a computational cell. This is a fundamental limitation set by the resolution of the representation. It is independent of the order of accuracy, type of expansion, and algorithms used.

The most obvious way to increase the order of accuracy for setting boundary conditions is to involve more points in the interpolation formulas. High-order polynomial interpolation assumes that the solution is smooth and if it is not smooth enough, unwanted oscillations may be introduced. The above interpolation methods except possibly the SP method are not data driven, in the sense that the interpolation stencil is fixed. The SP method chooses a limited slope to the linear interpolation, which is data-driven. Future research should investigate whether data driven higher-order interpolation such as Essentially Non-Oscillatory/Weighted Essentially Non-Oscillatory (ENO/WENO) type interpolation e.g. [1] could be used to set the boundary conditions. Alternatively, regions of non-smoothness can be detected using wavelets to compute the Hölder exponent, and the interpolating polynomial can be adjusted according to the (non-)smoothness of the solution. In [49] (**Paper IV**) wavelets are used to control the degree of an interpolating polynomial in grid/grid interpolation, but it can be used for the selection of interpolation stencil for the embedded boundary also. The only difference is that one-sided approximations must be used when computing the Hölder exponent near the boundary.

Chapter 6

Computational results

This chapter summarizes the computational results from **Papers I-VI**.

6.1 Large Eddy Simulations of a Turbulent Jet Diffusion Flame using FOAM (Paper I)

In this paper a comparison between LES and new type of experimental measurement technique is made. The experiment/simulation is of a non-premixed propane jet with Reynolds number 10000 into ambient air. The LES calculations are performed with a presumed β-PDF for a mixture fraction. As a SGS-model for the flow we have used the Smagorinsky and the One Equation Eddy Viscosity Model.

The comparison is made at intermediate stage by examining schlieren images, interferograms and phase maps consrtucted from the LES and experimental data respectively. This provides a novel approach for comparing simulations and experiments. Reasonable good agreement between measured and predicted flame characteristics and properties were found, although the details near the nozzle are not sufficiently accurate. The burning occurs primarily at the outer edges of the jet shear layer in the convoluted interface between reactants and products where most of the diffusive mixing takes place.

6.2 Large Eddy Simulation of Supersonic Axisymmetric Baseflow using FOAM (Paper II)

The paper describes LES of a rockets afterbody with and without mass injection. Several sub-grid models and grids are used to conclude that LES/MILES can be used with success for this type of problem, since the agreement with experimental data is good. It is shown that the results are independent of subgrid model making it impossible to distinguish numerical errors from model errors.

Simulations with different bleed rates are performed and comparison of C_p at the base are made with experimental data. C_p is however systematically overpredicted by 5%. A potential source of error may be underresolution and/or the difference in approach boundary-layer thickness between the experiments and the simulations; a thick boundary-layer is thought to have a base pressure enhancing effect, similar to that of base-bleed.

6.3 Large Eddy Simulation of a Turbulent Non-premixed Flame using FOAM (Paper III)

This paper considers a reactive and a non-reactive jet. For the reactive jet, we compare two different approaches to combustion modeling: the conserved scalar (mixture fraction) approach together with a presumed PDF and a single step finite rate chemistry model also known as Arrhenius type model. For the non-reacting case very good qualitative and quantitative agreement between predictions and measure ment data are obtained. For the LES calculations we find virtually no differences in results obtained using the different subgrid models. Grid refinement does not affect any of the investigated first and second order statistical profiles, but provides a more detailed picture of the fluid dynamics. For the reacting case the flow field is reasonably well predicted, independent of subgrid models. Refining the grid increases the resolution of the flow variables, but does not affect the statistical moments investigated here. The simulation of combustion, however, is sensitive to which model is used. The simple one-step irreversible one-step global reaction meachnism used in this study is too simple to describe all the couplings between the fluid dynamics and the chemistry.

6.4 2D Supersonic flow around a cylinder (Paper IV)

In [49] (**Paper IV**), the ability of the embedded boundary technique to resolve boundary layers is investigated by computing skin-friction profiles along the surfaces of the embedded objects. The accuracy is assesed by comparing the computed skin-friction profiles with those obtained by a body fitted discretization with the solver developed in [81] for the compressible MHD equations with magnetic fields set to zero.

For low Reynolds numbers one can resolve the flow. We measure convergence by C_f, C_p and temperature distributions on the embedded surface (which may be used to solve for the temperature distribution inside the embedded body). The C_P and C_f distributions are important, since they dictate what aerodynamic forces $\mathbf{F}$

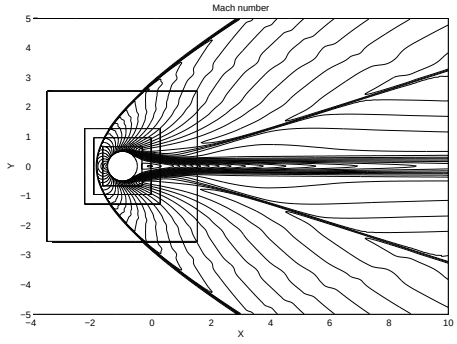


Figure 6.1. Two dimensional computations of Mach 3 flow past a cylinder. Velocity magnitude contours. Results from the KP method.

act upon the body by:

$$\mathbf{F} = \iint d\mathbf{F} = -\iint \underbrace{p\mathbf{n}}_{\approx C_p} dS + \iint \underbrace{\tau}_{\approx C_f} \mathbf{t} dS, \tag{6.1}$$

where $\mathbf{n}$ and $\mathbf{t}$ are the normal and tangential vectors respectively.

We here compute supersonic flow around a cylinder with radius 0.5 with Mach number 3 and Reynolds numbers 500 in the two dimensional domain $(x, y) \in [-10, 10] \times [-5, 5]$. The center of the cylinder is located at $(-1, 0)$. These simulations are time accurate, and resolved in time and space. As initial data, we impose free stream conditions in the entire domain. The discretization on the Cartesian grid is efficient because it has a simpler memory access pattern than an unstructured method and requires less metric information (and thereby less memory accesses and less arithmetic operations) than an approximation on a curvilinear grid. In fact the grid is never used in the computation.

The computations was run until steady-state. In the computations the timestep had to be restricted by the stability requirement of the viscous operator. We take this as an indication that the flow is resolved.

6.4.1 Description of the body fitted solver

The domain is discretized by the overset grid configuration displayed in Fig. 6.2 for the cylindrical flow problem. There are four grids, a base grid that covers the entire domain, a curved grid around the bow shock, a fine polar grid near the cylinder surface, and a fine grid that covers the wake region. We used the overset grid generator Xcog [63] to generate the grids and the interpolation information.

We discretized the Navier-Stokes equations by a sixth order accurate finite difference scheme with summation-by-parts boundary modification of the difference operators on all component grids except the bow shock grid, where we used a TVD

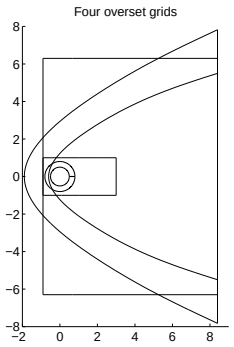


Figure 6.2. Overset grid domains used for computations with body fitted grids.

type difference scheme. These are standard finite difference methods, for details about the scheme and the code see [81, 71]. The solution was time marched to steady state, first with a TVD scheme on all grids, and later when the solution is fully developed, with the sixth order method on three of the grids, as described above.

Interesting questions about overall accuracy and error propagation from the bow shock are outside the scope of this work. However, it was observed in [81] that the actual grid convergence rate at the body boundary is close to 2nd order.

6.4.2 Results

In Fig. 6.6 we have collected the C_f curves from the finest grids in Figs. 6.3–6.5. The body fitted method and the KP embedded method give results that are indistinguishable in the plot. We conclude that the KP embedded boundary approach gives more accurate results than the SP embedded boundary method, and furthermore that the accuracy of the KP embedded boundary method is comparable to the accuracy of the body fitted method on the medium and fine resolution grids. On the coarsest grid, Figs. 6.3 and 6.5 show that the body fitted method is more accurate.

It is not unexpected that the KP method is more accurate than the SP method, because the SP method switches between a first and second order accurate boundary condition, whereas the KP method is always of high formal accuracy. The SP method uses limiters to handle shock waves, but the KP method uses centered interpolation stencils. Nevertheless, the KP method gave solutions that were free from unphysical oscillations, since a resolved boundary layer does not contain discontinuities.

The formal order of accuracy is very important as shown in Figure 6.7, which shows results from the computation using the first order extrapolation of the slopes

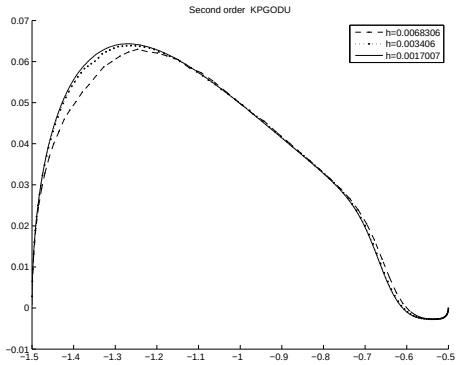


Figure 6.3. C_f along the upper half of the cylinder computed with the KP embedded boundary method for Mach number 3 and Reynolds number 500.

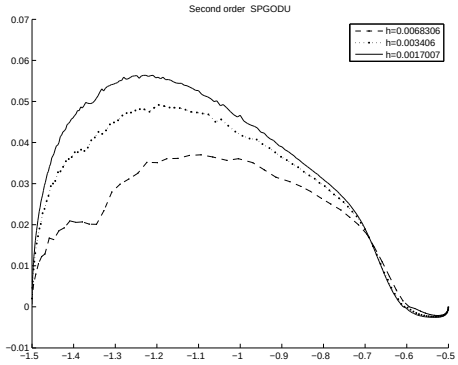


Figure 6.4. C_f along the upper half of the cylinder computed with the SP embedded boundary method for Mach number 3 and Reynolds number 500.

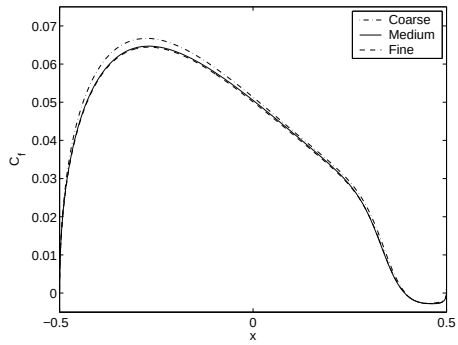


Figure 6.5. C_f along the upper half of the cylinder computed with the body fitted method for Mach number 3 and Reynolds number 500.

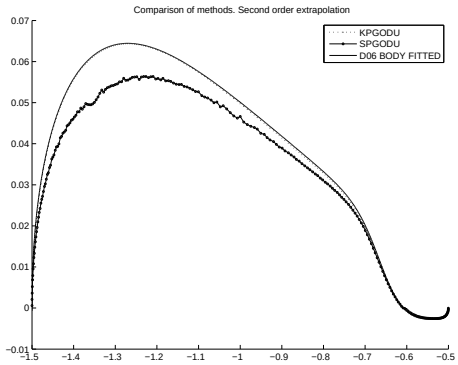


Figure 6.6. C_f along the cylinder surface with the KP embedded boundary method, the SP embedded boundary method, and the body fitted method. Mach number 3, Reynolds number 500. The finest grid size is $h = 0.0017007$. Using second order extrapolation of the slopes in the Godunov fluxes.

in (5.18)

$$\mathbf{s}_{i_{bp},j,k} = \mathbf{s}_{i_{bp}\pm 1,j,k} \tag{6.2}$$

instead of the second order extrapolation (results shown in Figure 6.6)

$$\mathbf{s}_{i_{bp},j,k} = 2\mathbf{s}_{i_{bp}\pm 1,j,k} - \mathbf{s}_{i_{gp}\pm 2,j,k}. \tag{6.3}$$

We conclude that when the physical viscosity is not resolved, which is the case for simulations using the coarse mesh then the skin friction cannot be expected to be accurate raising the need for local grid refinement.

It is well-known that for attached boundary layers, one obtains good resolution by stretching the grid towards the body. The coarser resolution in the tangential direction saves computational work. The ability to coarsen the grid in the direction tangential to the body is clearly absent in the embedded boundary method. We conclude that for attached laminar boundary layers, this feature makes the body fitted approximation considerably more efficient. However, when resolution is equal in both directions, Fig. 6.6 shows that the embedded boundary method gives results of similar quality as with the body fitted method. Equal resolution in all direction is needed in direct simulation of turbulent separating flows. Furthermore, with complicated geometries it might not be known a priori at which locations the boundary layer is attached and therefore it would not be possible take advantage of body fitted stretched grids.

The temperature on the boundary

The adiabatic wall condition imposes $\frac{\partial T}{\partial n} = 0$. We evaluate the accuracy of the Neumann boundary condition by plotting the temperature on the surface. Fig. 6.8

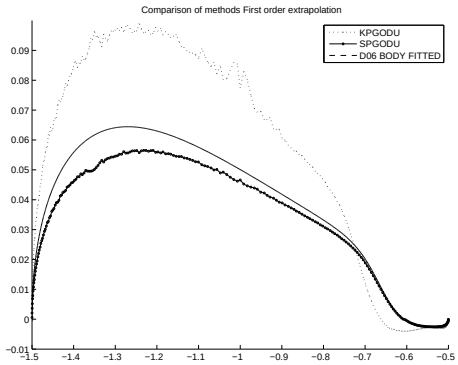


Figure 6.7. C_f along the cylinder surface with the KP embedded boundary method, the SP embedded boundary method, and the body fitted method. Mach number 3, Reynolds number 500. The finest grid size is $h = 0.0017007$. First order extrapolation (6.2).

shows the wall temperature obtained with the KP method and Fig. 6.9 displays the wall temperature obtained with the SP method. Similarly to the C_f plots, the KP method appears to be more accurate than the SP method.

6.5 Embedded Boundaries and Roughness

The specific errors of the embedded boundary method are assessed by comparing two almost identical simulations: one grid aligned and one 'tilted' simulation, see Figure 6.10.

When the flow is unresolved, typically for high Reynolds numbers, the truncation errors from the EB method can be seen as acting effectively as a rough wall. This can be seen by a 'reconstruction' resembling a finite difference operator of the boundary from the solution based on the tangential velocity. We find θ such that

$$\theta u_{ghostPoint} + (1-\theta) u_{FirstInteriorPoint} = 0, \tag{6.4}$$

and use the θ to evaluate the location of the 'zero'-contour

$$\vec{x}_{\Gamma_{Eff}} = \theta \vec{x}_{ghostPoint} + (1-\theta)\vec{x}_{FirstInteriorPoint},$$

The deviation from the true boundary is called protrusion and its height is measured as $k = \min_{\vec{x}\in\Gamma} |\vec{x}_{\Gamma_{Eff}} - \vec{x}_{\Gamma}|$.

From Figure 6.12 the following observations are made:

- The protrusion height $k = C \cdot h$,
- the friction velocity $u_\tau \sim constant$ increases with finer mesh,

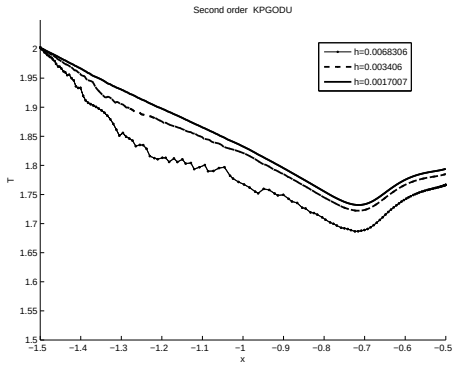


Figure 6.8. Temperature on the boundary using the KP embedded boundary method. Mach number 3, Reynolds number 500.

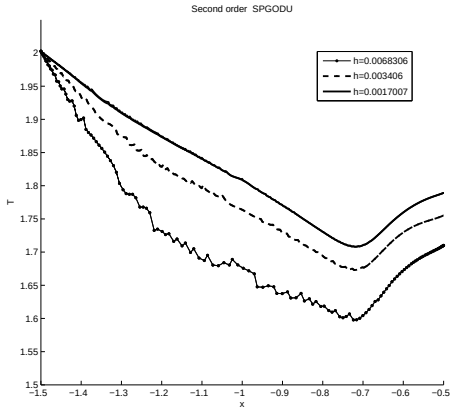
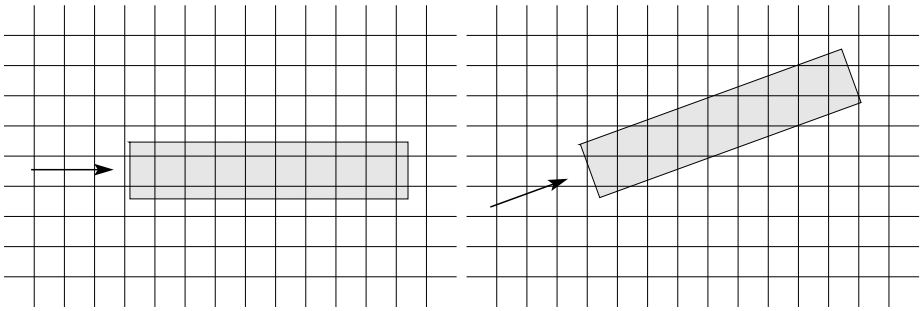
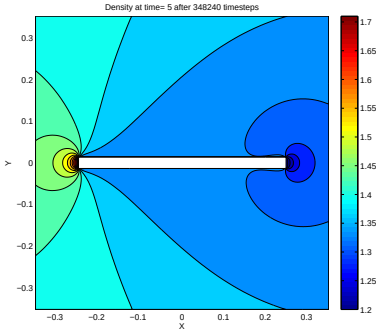


Figure 6.9. Temperature on the boundary using the SP embedded boundary method. Mach number 3, Reynolds number 500.

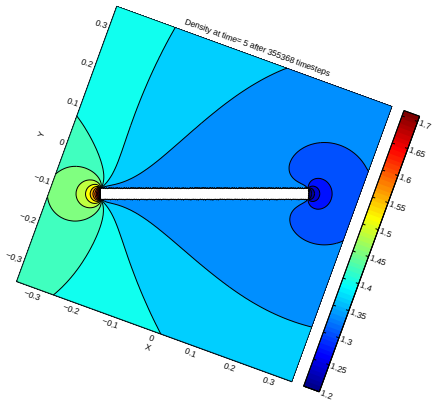


(a) 0^o (b) 20^o

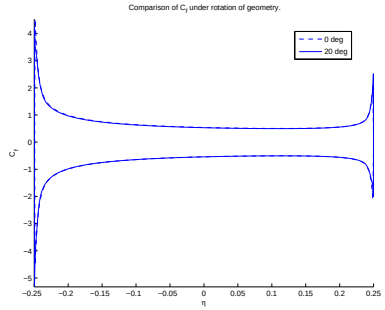
Figure 6.10. An illustration of the experiment with different grid alignments.



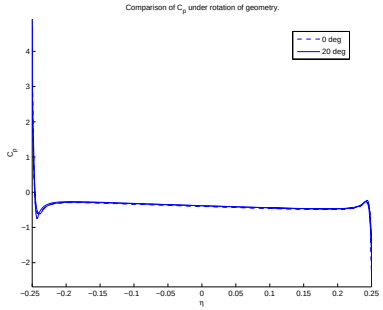
(a) 0^o



(b) 20^o

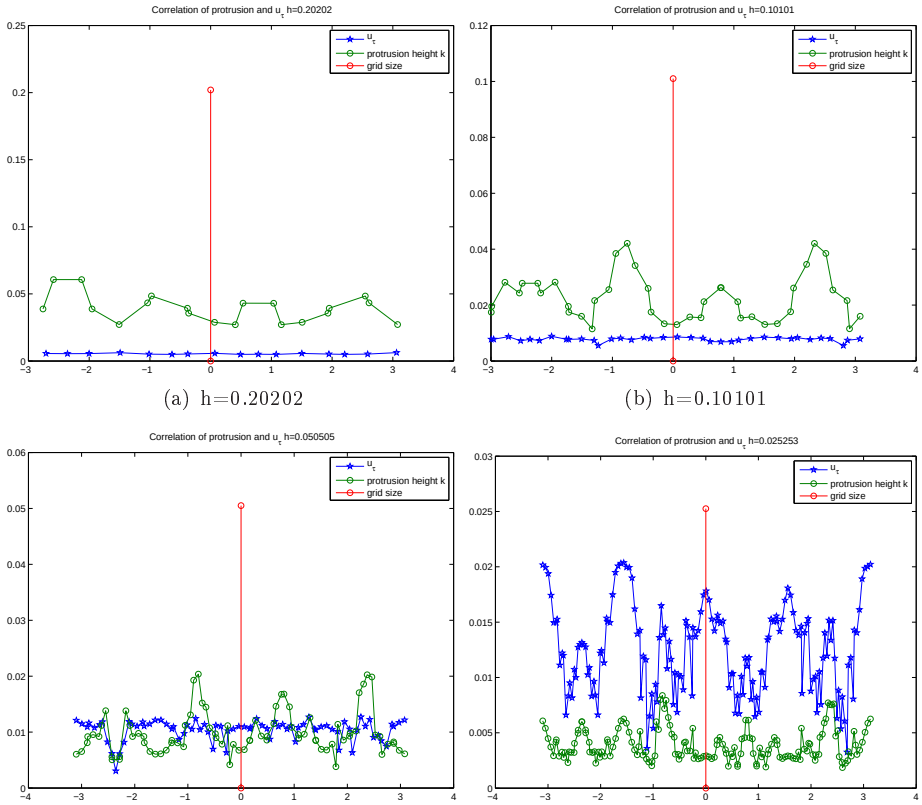


(c) C_f distribution.



(d) C_p distribution.

Figure 6.11. The density for $h_{min} = 1.50754 \cdot 10^{-3} Re_a = 50$.



(a) h=0.20202 (b) h=0.10101

(c) h=0.050505 (d) h=0.025253

Figure 6.12. The protrusion height estimated using (6.4) together with the friction velocity u_τ. The vertical bar illustrates the grid size h.

- there exists correlation between u_τ and k.

In cases where protrusions are of size $\delta_{laminar}$ they are all contained in the laminar sublayer, i.e. if $k < \delta_{laminar}$, the wall may be considered hydraulically smooth [69]. We have $\delta_{laminar} \sim C \cdot \nu/u_\tau$. The dimensionless roughness factor $k/\delta_{laminar} \sim ku_\tau/\nu$, a roughness Reynolds number Re_k based on the protrusion size and friction velocity, is a good measure for the relative roughness.

Circular pipes covered with sand of a definite grain size, k_s, glued on the wall have been used in experiments to determine three regimes of Re_{k_s} [69]:

1. Hydraulically smooth: The size of the roughness is so small that all protrusions are contained within the laminar sub-layer, $Re_{k_s} \in [0, 5]$,

2. Transition regime: Protrusions extend partly outside the laminar sub-layer and additionally when comparing to smooth pipes there are mainly effects on the resistance from the protrusions in the boundary layer, $Re_{k_s} \in [5, 70]$.

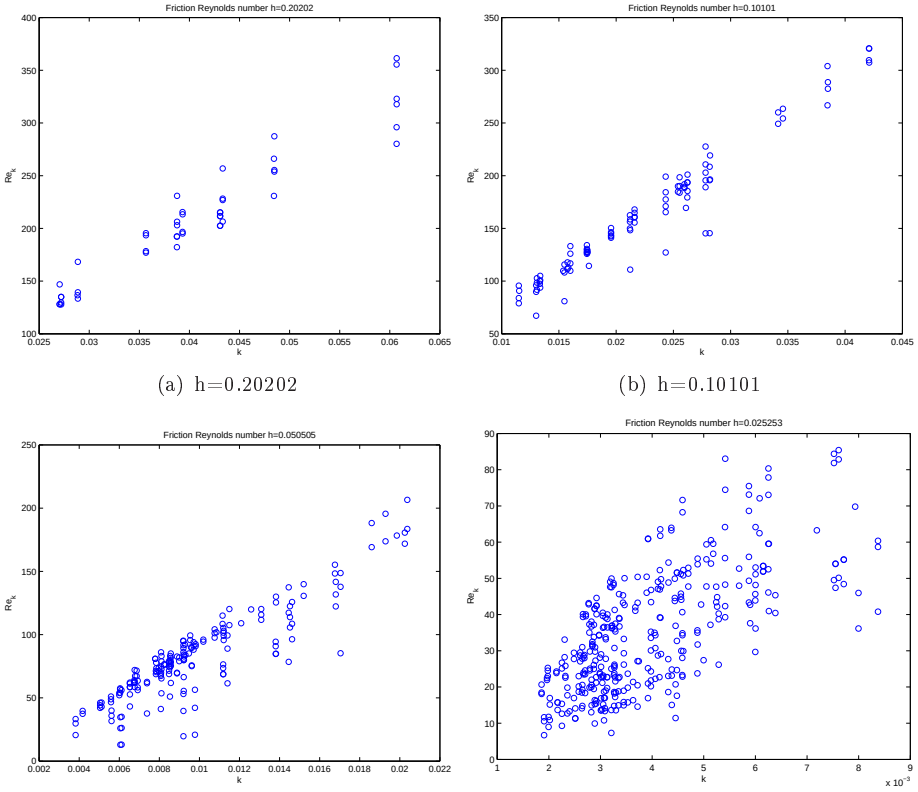
3. Completely rough regime: All protrusions reach outside the laminar sub-layer and the largest part of the resistance to flow is due to form drag which acts on them, $Re_{k_s} \in [70, \infty)$.

The condition for hydraulical smoothness is also valid for flat plates at zero incidence [69]. Assuming that $k \approx$ equivalent grainsize k_s, we believe that our estimate can be used e.g. as an indicator in a grid adaptation algorithm. Below, see Figure 6.13, we have computed Re_k based upon actual simulation data. It shows that when the grid is coarse the geometry is completely numerically rough and after grid refinement the bulk of the points shift towards the transition regime. Much smaller k (i.e. mesh sizes) are required to reach the hydraulically smooth regime.

Varying Reynolds Number

This test is performed using a fixed discretization and varying the acoustic Reynolds number. In this way we can assess how far from resolved the simulation can be using this embedded boundary method and still obtain similar solutions after rotation of the geometry, see Figure 6.10.

The C_f distributions are very similar for $Re_a = 3125$, 12500 and somewhat similar for $Re_a = 25000$ where the 20^o case has some oscillations superimposed on the 0^o C_f distribution. It is somewhat surprising that for $Re_a = 6250$ the C_f distribution differs so much, due to the fact that 0^o discretization predicts separation on the leading edge, while the 20^o does not. For all $Re_a > 6250$ there is leading edge separation. With $Re_a = 50000$ both 0^o and 20^o C_f distributions show great oscillations showing no similarity, except perhaps in the mean amount. The $Re_a = 50000$ flow is 'turbulent' and we cannot expect to get point-wise convergence, but should rather expect convergence in mean. The convergence of means is not investigated for the above cases.



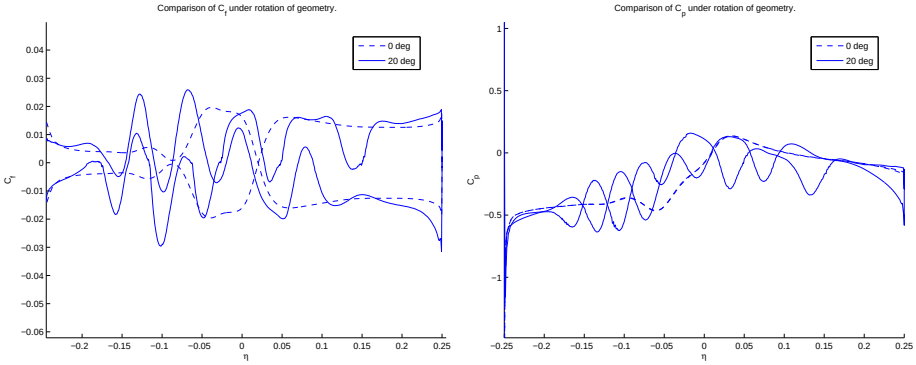
(a) h=0.20202

(b) h=0.10101

(c) h=0.050505

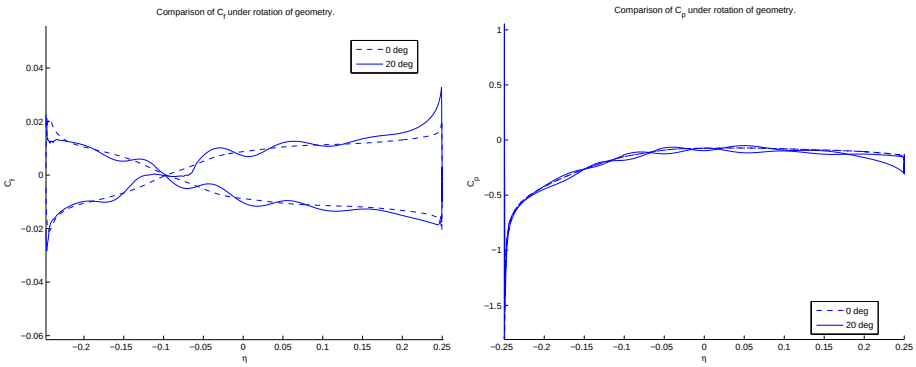
(d) h=0.025253

Figure 6.13. Correlation between friction Reynolds number Re_k and protrusions k computed from simulation data.



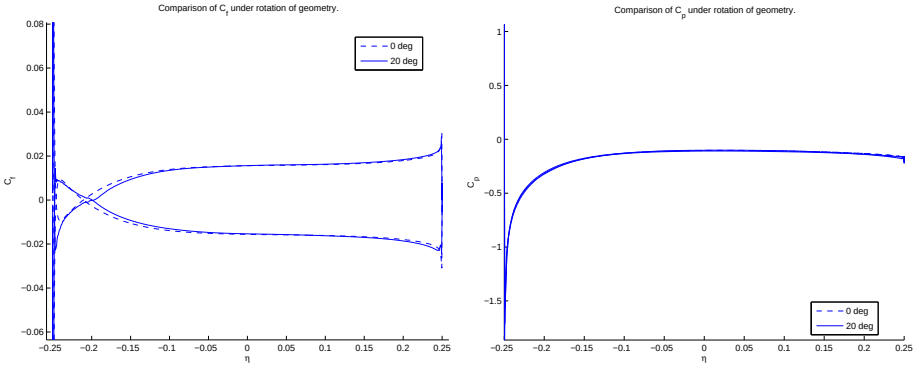
(a) Comparison of C_f distribution. (b) Comparison of C_p distribution.

Figure 6.14. Unsteady flow, instantaneous velocity at $t = 5$ $Re_a = 50000$ $h_{min} = 7.5188 \cdot 10^{-4}$. Notice that the 20^o simulation has more wiggles, which are due to the wagging wake.



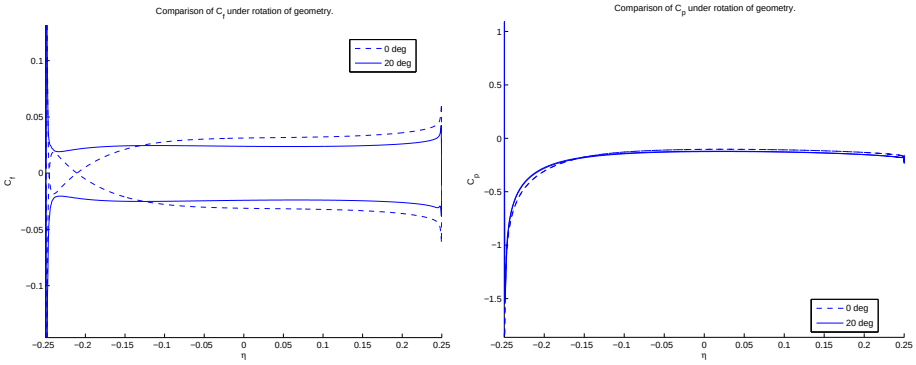
(a) Comparison of C_f distribution. (b) Comparison of C_p distribution.

Figure 6.15. Unsteady flow, instantaneous velocity at $t = 5$ $Re_a = 25000$ $h_{min} = 7.5188 \cdot 10^{-4}$. Note that the 20^o simulation has more unsteadiness which are caused by pressure waves from the unsteady wake.



(a) Comparison of C_f distribution. (b) Comparison of C_p distribution.

Figure 6.16. Instantaneous velocity at $t = 5$ $Re_a = 12500$ $h_{min} = 7.5188 \cdot 10^{-4}$. Notice the leading edge separation.



(a) Comparison of C_f distribution. (b) Comparison of C_p distribution.

Figure 6.17. Instantaneous velocity at $t = 5$, $Re_a = 6250$ $h_{min} = 7.5188 \cdot 10^{-4}$. Notice the 0^o the leading edge separation, but attached flow for 20^o.

The C_p distributions seem to be the same independent of the Re_a number, and for all Re_a, except $Re_a = 50000$ which would require higher resolution, the C_p distributions are very similar for 0^o and 20^o.

High Reynolds number flow ($Re_a = 10^6$)

We consider this a high Reynolds number case, where we successively refine the grid size by a factor of two in three computations. This is an illustration of a highly unresolved computation using IB/EB. The solution contains a lot of eddies emanating from the boundary. This illustrates the resolution needed to resolve the flow and highlights the need for wall-models.

For $Re_a = 10^6$ the flow is 'turbulent' and we cannot expect to get point-wise convergence. Both the C_f and C_p distributions are very oscillatory/unsteady for all resolutions. Therefore we examine the time averages $\langle C_f \rangle$ and $\langle C_p \rangle$ and examine the impact the embedded boundary has on the solution. It is discovered that the truncation error from the EB is manifested as roughness, estimated by linear reconstruction of the boundary. The size of the roughness is estimated by the friction velocity Reynolds number Re_τ. We compare the obtained Re_τ with Re_τ for rough pipes, [69].

6.6 EB applied to supersonic baseflow

Physical experiments of supersonic axisymmetric baseflow are compared to LES using embedded boundaries in [46] (**Paper VI**). The results are summarized here.

In the experiments [32, 53, 9, 8, 37] the centerline velocity downstream of a cylinder and the pressure coefficient on the cylinder base were measured. Numerical simulations of supersonic baseflow were performed in [22] (**Paper II**)using FOAM and we wanted to compare those results with results from the embedded boundary code. We used the simple Smagorinsky model and tried to do wall-resolved LES, which is too costly for $Re \approx 10^6$. The experiment however shows that the embedded boundary method works efficiently in 3D. The computational cost is significantly lower for the EB, even when using the very expensive Riemann solver, than for the unstructured solver FOAM. Below we show some obtained results; from Figures 6.19 and 6.20 we see that grid refinement has the greatest impact on the solution. The tuning of model parameters such as C_s does not change much. This is much due to the low order (and dissipative) 2:nd order Godunov scheme. This test case shows the EBM extended to three dimensional problems with the Smagorinsky sub grid scale model. The geometry contains sharp corners, for which our proposed strategy (see section 5.3.4) seems to work well.

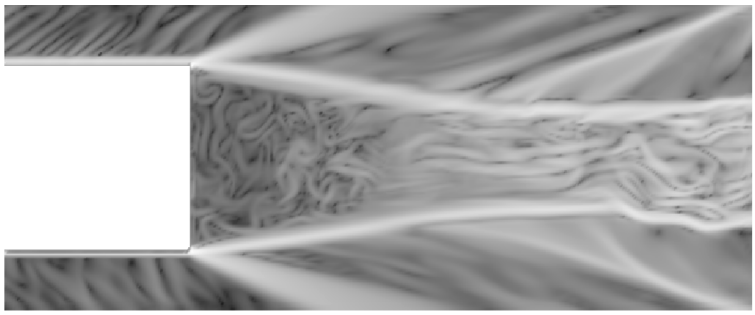


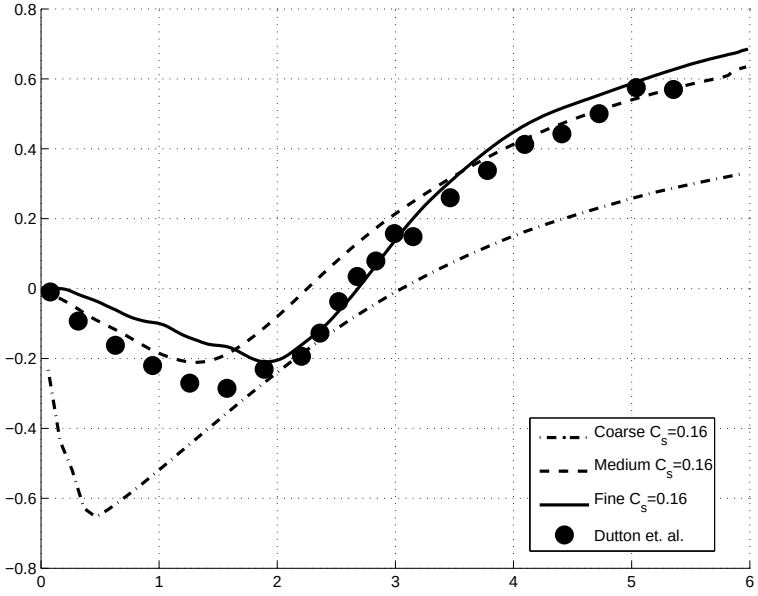
Figure 6.18. Instantaneous visualization of $\log(\|\nabla\rho\| + \varepsilon)$. The characteristic features of supersonic baseflows can be seen such as the unsteady nature of such flows with the presence of numerous(?) turbulent scales. The separation point is fixed by the geometry at the corner. A centered expansion fan turns the separated shear layer towards the axis. Further downstream, due to axisymmetric constraints, the mixing layer is bent to realign the flow with the axis in the mean. This region exhibits a strong adverse pressure gradient as evident by the presence of unsteady recompression shocks and Mach-lines coalescing into shocks. In this region, the incoming fluid that lacks the momentum to overcome the pressure gradient is pushed upstream into a recirculation zone. Downstream of the stagnation region, a turbulent wake with larger coherent structures develops.

6.7 Some Preliminary Results using EBM together with a Wall-model

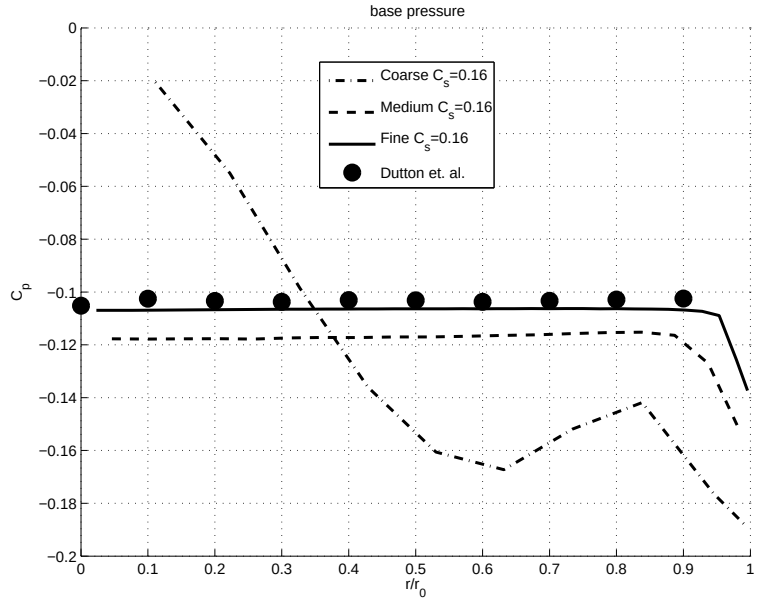
As we have postulated multiple times, wall-models are needed for high Reynolds number flows. Here we briefly summarize some preliminary findings using the wall-model described in section 4.0.5. We make a similar comparison as in section 6.5 of an embedded rectangle in $Re_a = 10^6$, $Ma = 0.3$ flow using the same discretization as above.

Results are shown in figures (6.22-6.23), where the coarsest grids do not resolve the flow at all, which is seen by the too smooth solutions. As the resolution is increased more 'turbulent' structures become resolved.

We observe that the friction velocity u_τ computation in the wall-model depends on y^+ and hence $h^+ = \frac{u_\tau h}{\nu}$ is more non-smooth using this wall-model, than just imposing the standard no-slip boundary condition, compare Figures 6.21 and 6.24.

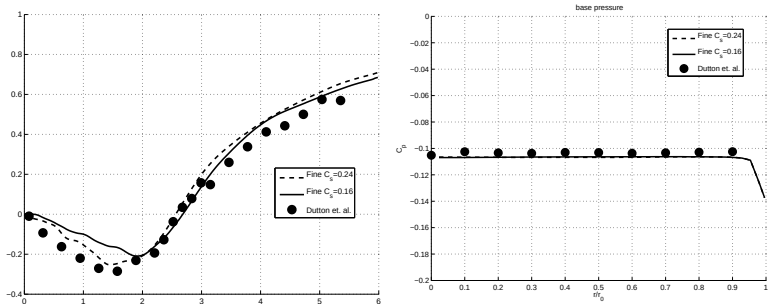


(a) $Pr_T = 0.7$ $C_s = 0.16$



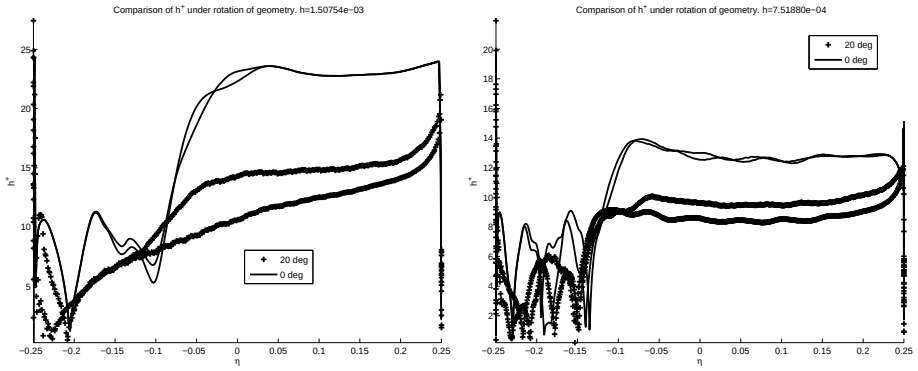
(b) $Pr_T = 0.7$ $C_s = 0.16$

Figure 6.19. Grid convergence study of velocity $\langle u \rangle$ along the base centerline and base pressure coefficient $\langle C_p \rangle$.



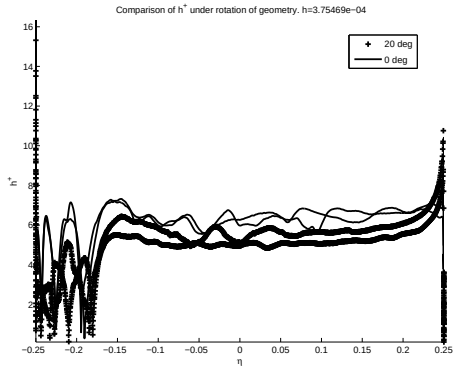
(a) Fine mesh. (b) Fine mesh.

Figure 6.20. Velocity along the base centerline $\langle u \rangle$ and $\langle C_p \rangle$ using $C_s = 0.16$ and 0.24. The model parameter affects the solution only in very small details, which is a consequence of the too dissipative Godunov scheme.



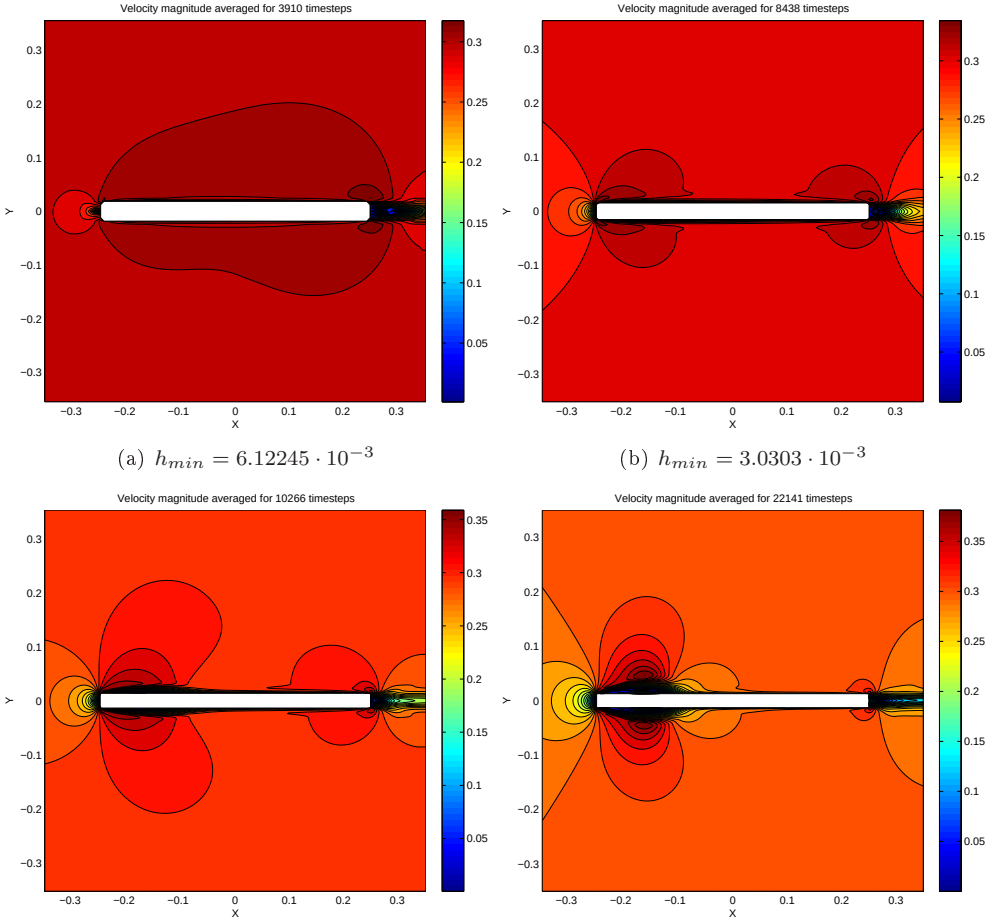
(a) $h_{min} = 1.50754 \cdot 10^{-3}$

(b) $h_{min} = 7.5188 \cdot 10^{-4}$



(c) $h_{min} = 3.75469 \cdot 10^{-4}$

Figure 6.21. $\langle h^+ \rangle$, i.e. the effective discretization size along the rotated and non-rotated object under grid refinement.



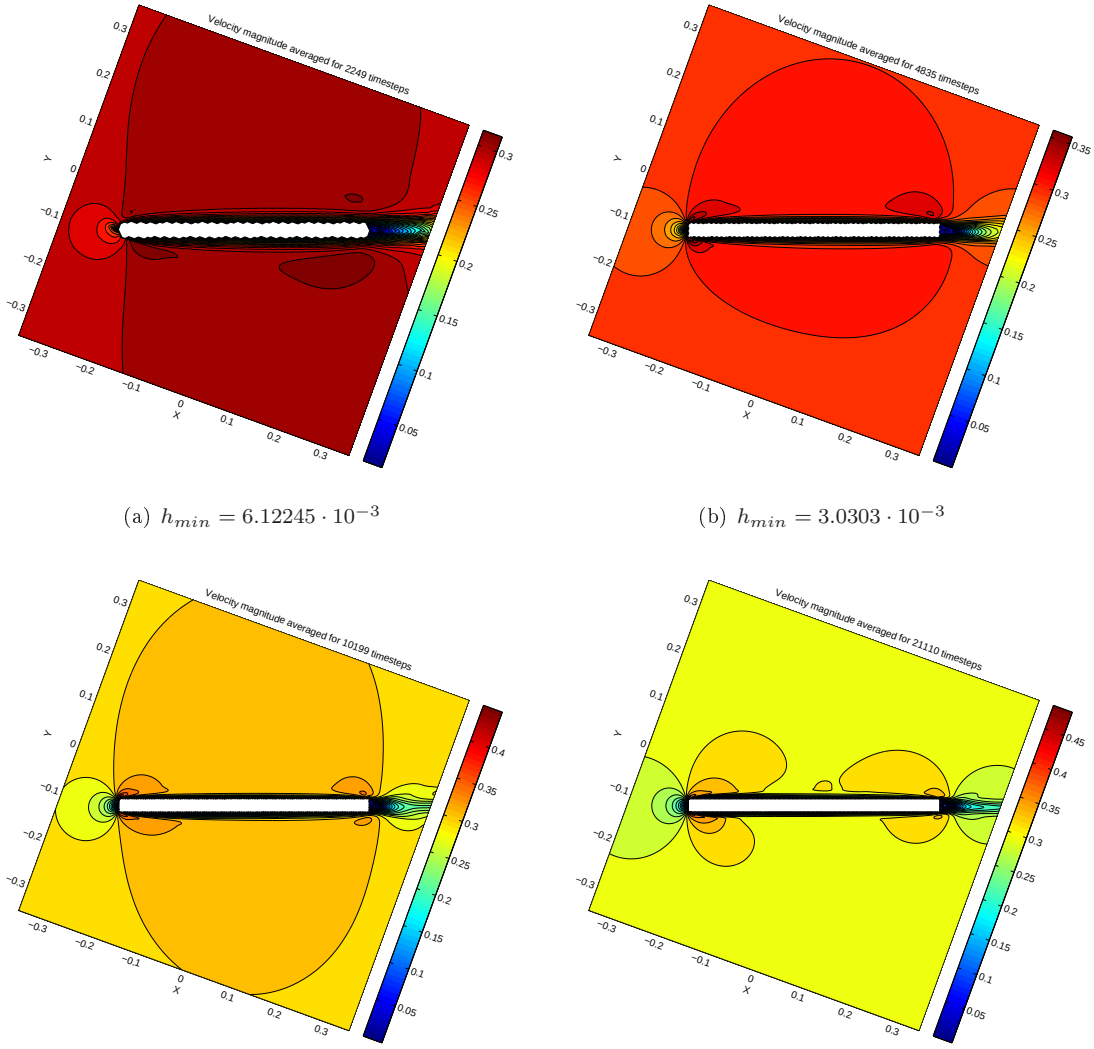
(a) $h_{min} = 6.12245 \cdot 10^{-3}$

(b) $h_{min} = 3.0303 \cdot 10^{-3}$

(c) $h_{min} = 1.50754 \cdot 10^{-3}$

(d) $h_{min} = 7.5188 \cdot 10^{-4}$

Figure 6.22. $\langle \|u\| \rangle$ 0^o using the wall-model desribed in section 4.0.5 and LES (standard Smagorinsky).



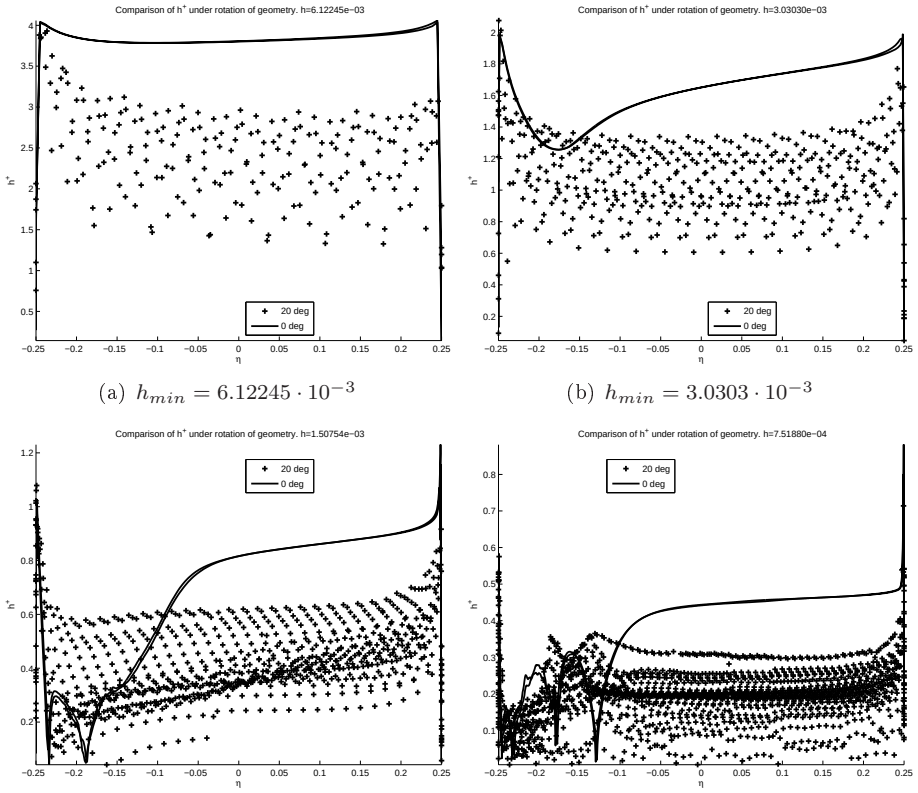
(a) $h_{min} = 6.12245 \cdot 10^{-3}$

(b) $h_{min} = 3.0303 \cdot 10^{-3}$

(c) $h_{min} = 1.50754 \cdot 10^{-3}$

(d) $h_{min} = 7.5188 \cdot 10^{-4}$

Figure 6.23. $\langle \|u\| \rangle$ 20^o using the wall-model desribed in section 4.0.5 and LES (standard Smagorinsky).



(a) $h_{min} = 6.12245 \cdot 10^{-3}$

(b) $h_{min} = 3.0303 \cdot 10^{-3}$

(c) $h_{min} = 1.50754 \cdot 10^{-3}$

(d) $h_{min} = 7.5188 \cdot 10^{-4}$

Figure 6.24. $\langle h^+ \rangle$, i.e. the effective discretization size along the rotated and non-rotated object under grid refinement.

References

[1] F. Aràndiga and A. M. Belda. Weighted ENO interpolation and applications. *Communications in Nonlinear Science and Numerical Simulation*, 9:187–195, 2004.

[2] P.-O. Åsen. A parallel code for direct numerical simulations of pipe Poiseuille flow. Technical report, NADA/CSC Royal Institute of Technology, 2007.

[3] J. S. Baggett, J. Jimenez, and A. G. Kravchenko. Resolution requirements for large-eddy simulations of shear flows. *CTR Annual Research Briefs*, page 51, 1997.

[4] M. J. Berger and P. Colella. Local adaptive mesh refinement for shock hydrodynamics. *Journal of Computational Physics*, 82:64–84, 1989.

[5] M. J. Berger and J. Oliger. Adaptive mesh refinement for hyperbolic partial differential equations. *Journal of Computational Physics*, 53:484–512, 1984.

[6] M. Berglund. *Large Eddy Simulation of Complex Turbulent Flows*. PhD thesis, Lund Institute of Technology, 2006.

[7] J. P. Boris, F. F. Grinstein, E. S. Oran, and R. L. Kolbe. New insights into large eddy simulation. *Fluid Dynamics Research*, 10:199–228, 1992.

[8] C. J. Bourdon and J. C. Dutton. Planar visualisations of large-scale turbulent structures in axisymmetric supersonic separated flows. *Physics of Fluids*, 11(1):201–213, 1999.

[9] C. J. Bourdon, J. C. Dutton, K. M. Smith, and T. Mathur. Planar visualisations of large scale turbulent structures in axisymmetric supersonic base flows. *AIAA Paper No. 98-0624*, 1998.

[10] A. Bourgeat and B. Cockburn. The TVD-projection method for solving implicit numerical schemes for scalar conservation laws: A numerical study of a simple case. *SIAM Journal on Scientific and Statistical Computing*, 10:253–273, 1989.

[11] J. P. Boyd. *Chebyshev and Fourier Spectral Methods*. Dover, New York, 2001.

[12] B. Cockburn and C.-W. Shu. Runge-Kutta discontinuous Galerkin methods for convection-dominated problems. *Journal of Scientific Computing*, 16(3):173–261, 2001.

[13] P. Colella, D. T. Graves, B. J. Keen, and D. Modiano. A Cartesian grid embedded boundary method for hyperbolic conservation laws. *Journal of Computational Physics*, 211(347-366), 2006.

[14] J. Donea, A. Huerta, J.-P. Ponthot, and A. Rodríguez-Ferran. *Encyclopedia of Computational Mechanics*, volume 1, chapter Arbitrary Langrangian-Eulerian Methods. John Wiley & Sons, Ltd., 2004.

[15] Z. Dragojlovic, F. Najmabadi, and M. Day. An embedded boundary method for viscous, conducting compressible flow. *Journal of Computational Phyics*, 216(1):37–51, 2006.

[16] F. Ducros, V. Ferrand, F. Nicoud, C. Weber, D. Darracq, C. Gacherieu, and T. Poinsot. Large-eddy simulation of shock/turbulence interaction. *Journal of Computational Physics*, 152, 1999.

[17] J. P. Dussauge. Compressible turbulence and energetic scales: What is known from experiments in supersonic flows? *Flow, Turbulence and Combustion*, 66:373–391, 2001.

[18] B. Einfeldt, C. D. Munz, P. L. Roe, and B. Sjğreen. On Godunov-type methods near low densities. *Journal of Computational Physics*, 92:273–295, 1991.

[19] G. Erlebacher, M. Y. Hussaini, C. G. Speziale, and T. A. Zang. Toward the large-eddy simulation of compressible turbulent flows. *Journal of Fluid Mechanics*, 238:155–185, 1992.

[20] M. Farge, N.-R. Kevlahan, V. Perrier, and K. Schneider. Turbulence analysis, modelling and computing using wavelets. In J. C. van den Berg, editor, *Wavelets in Physics*. Cambridge University Press, 2004.

[21] C. Fureby and F. F. Grinstein. Recent progress on MILES for high re flows. *Journal of Computational Physics*, 181:68, 2002.

[22] C. Fureby and M. Kupiainen. Large eddy simulation of supersonic axisymmetric baseflow. *Proceedings to Turbulent Shear Flow Phenomena 3, Sendai*, 3, 2003.

[23] L. Gasparini. Some results using a new compressible flow solver for openfoam. Downloadable from http://openfoamwiki.net/index.php/TestLucaG.

[24] L. Georges. *Development and validation of a LES methodology for complex wall-bounded flows*. PhD thesis, Universite catholique de Louvain, 2007.

[25] S. K. Godunov. Finite difference method for numerical computation of discontinous solution of the equations of fluid dynamics. *Matematicheskii Sbornik*, 47:271, 1959.

[26] F. Grinstein and C. Fureby. *High-Resolution Schemes for Convection-Dominated Flows: 30 Years of FCT*, chapter On Monotonically Integrated Large Eddy Simulation of Turbulent Flows Based on FCT Algorithms. Springer, 2005.

[27] F. Grinstein, C. Fureby, and C. DeVore. On MILES based on flux-limiting algorithms. *International Journal of Numerical Methods in Fluids*, 47:1043–1051, 2005.

[28] F. F. Grinstein and C. Fureby. Recent progress on MILES for high reynolds number flows. *Journal of Fluids Engineering*, 124:848–861, 2002.

[29] F. F. Grinstein, L. G. Margolin, and W. J. Rider, editors. *Implicit Large Eddy Simulation, Computing Turbulent Fluid Dynamics*. Cambridge University Press, 2007.

[30] A. Harten. High resolution schemes for hyperbolic conservation laws. *Journal of Computational Physics*, 49:357–393, 1983.

[31] W. D. Henshaw and D. W. Schwendenman. Moving overlapping grids with adaptive mesh refinement for high-speed reactive and non-reactive flow. *Journal of Computational Physics*, 216:744–779, 2006.

[32] J. L. Herrin and C. J. Dutton. Supersonic base flow experiments in the near wake of a cylindrical afterbody. *AIAA Journal*, 1994.

[33] J. O. Hinze. *Turbulence*. McGraw-Hill Book Company, second edition, 1975.

[34] C. Hirsch. *Numerical Computation of Internal and External Flows*, volume 1 and 2. J. Wiley and Sons, 1999.

[35] http://www.ansys.com/products/icemcfd.asp. Internet, 2008.

[36] http://www.fluent.com/software/gambit/. Internet, 2008.

[37] J. R. Janssen and J. C. Dutton. Time-series analysis of supersonic base-pressure fluctuations. *AIAA Journal*, 42(3):605–613, 2004.

[38] R. M. Kirby and G. E. Karniadakis. Under-resolution and diagnostics in spectral simulations of complex-geometry flows. In D. Drikakis and B. J. Geurts, editors, *Turbulent Flow Computation*, volume 66. Kluwer Academic Publishers, 2002.

[39] D. Knight, G. Zhou, N. Okong'o, and V. Shukla. Compressible large eddy simulation using unstructured grids. *AIAA*, Paper 98-0535, 1998.

[40] A. N. Kolmogorov. Local structure of turbulence in an incompressible fluid for very large Reynolds numbers. *Comptes rendus (Doklay) de l'Academie des Sciences de l'U.R.S.S*, 31:301–305, 1941. Reprinted in: S.K. Friedlander and L. Topper (editors) 1961. Turbulence: Classic Papers on Statistical Theory, New York: Interscience Publishers.

[41] A. N. Kolmogorov. On degeneration of isotropic turbulence in an incompressible viscous liquid. *Comptes rendus (Doklay) de l'Academie des Sciences de l'U.R.S.S*, 31:538–540, 1941. ibidem.

[42] A. N. Kolmogorov. A refinement of previous hypotheses concerning the local structure of turbulence in a viscous incompressible fluid at high Reynolds number. *Journal of Fluid Mechanics*, 13:82–85, 1962.

[43] H.-O. Kreiss and N. A. Peterson. A second order accurate embedded boundary method for the wave equation with dirichlet data. *SIAM Journal on Scientific Computing*, 2004.

[44] H.-O. Kreiss, N. A. Peterson, and J. Yström. Difference approximations for the second order wave equation. *SIAM Journal of Numerical Analysis*, 40(5):1940–1967, 2002.

[45] H.-O. Kreiss, N. A. Peterson, and J. Yström. Difference approximations of the neumann problem for the second order wave equation. *SIAM Journal on Numerical Analysis*, 2003.

[46] M. Kupiainen. LES baseflow simulations using a cartesian embedded boundary method. Technical report, NADA/CSC, 2008.

[47] M. Kupiainen. On the accuracy of an immersed/embedded boundary method for compressible turbulent flow. Technical report, NADA/CSC, 2008.

[48] M. Kupiainen and C. Fureby. Large eddy simulation of a turbulent non-premixed flame. *Proceedings to Turbulent Shear Flow Phenomena 3, Sendai*, 3, 2003.

[49] M. Kupiainen and B. Sjögreen. A Cartesian embedded boundary method for the compressible Navier-Stokes equations. *accepted for publication in Journal of Scientific Computing*, 2008.

[50] A. M. Landsberg and J. P. Boris. The virtual embedding method: A simple approach for gridding complex geometries. *AIAA paper 97-1982*, 1997. Reston, VA: American Institute of Aeronautics and Astronautics.

[51] R. J. LeVeque. *Numerical Methods for Conservation Laws.* Birkhäuser, second edition, 1992.

[52] M. P. Martin, U. Piomelli, and G. V. Chandler. Subgrid-scale models for compressible large-eddy simulations. *Theoretical and Computational Fluid Dynamics*, 13:361–376, 2000.

[53] T. Mathur and J. C. Dutton. Base-bleed experiments with a cylindrical afterbody in supersonic flow. *Journal of Spacecrafts and Rockets*, 33, 1996.

[54] P. Moin, K. Squires, W. Cabot, and S. Lee. A dynamic subgrid-scale model for compressible turbulence and scalar transport. *Physics of Fluids A*, 3(11):2746–2757, November 1991.

[55] D. Monfort, S. Benhamadouche, and P. Sagaut. Meshless approach for wall treatment in large-eddy simulation. submitted to Physics of Fluids, 2007.

[56] A. S. Monin and A. M. Yaglom. *Statistical Fluid Mechanics: Mechanics of turbulence*, volume 1. Cambridge, 1971.

[57] K. Nakahashi and L. Kim. *Computational Fluid Dynamics 2004, Proceedings of the Third International Conference on Computational Fluid Dynamics, ICCFD3, Toronto, 12-16 July 2004*, chapter High-Density Mesh Flow Computations by Building-Cube Method, pages 121–126. Springer Berlin Heidelberg, 2006.

[58] M. Oberlack. Invariant modeling in large-eddy simulation of turbulence. *CTR Annual Research Briefs*, 1997.

[59] K. Okita, T. Tawara, and K. Ono. Shape representation by signed distance function for immersed boundary method. In *ECCOMAS 2008*, 2008.

[60] O. K. Olsen. Embedded boundary method for Navier-Stokes equations. Master's thesis, Royal Institute of Technology, 2005.

[61] E. S. Oran and J. P. Boris. *Numerical Simulation of Reactive Flows.* Elsevier, New York, 1987.

[62] D. Papamoschou. Evidence of shocklets in a counterflow supersonic shear layer. *Physics of Fluids*, 7(2):233–235, February 1995.

[63] A. Petersson. http://www.andrew.cmu.edu/user/sowen/software/xcog.html. Internet.

[64] S. B. Pope. *Turbulent Flows.* Cambridge, 2000.

[65] P. L. Roe. Approximate Riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics*, 43:357–372, 1981.

[66] P. Sagaut. *Large Eddy Simulation for Incompressible Flows.* Springer, 1998.

[67] P. Sagaut. *Implicit Large Eddy Simulation*, chapter Subgrid-Scale Modeling: Issues and Approaches. Cambridge University Press, 2007.

[68] Sandia. http://www.ca.sandia.gov/tnf/dataarch/sandchn.html. Internet, 2003.

[69] D. H. Schlichting. *Boundary-Layer Theory.* McGraw-Hill, seventh edition, 1987.

[70] B. Sjögreen and N. A. Peterson. A Cartesian embedded boundary method for hyperbolic conservation laws. *Communications in Computational Physics*, 2(6):1199–1219, 2007.

[71] B. Sjögreen and H. Yee. Variable high order multiblock overlapping grid methods for mixed steady and unsteady multiscale viscous flows. *Commun. Comput. Phys.*, 5:730–744, 2009.

[72] B. Sjögreen and H. C. Yee. Multiresolution wavelet based adaptive numerical dissipation control for high order methods. *Journal of Scientific Computing*, 20(2), 2004.

[73] J. Smagorinsky. General circulation experiments with the primitive equations. i. the basic experiment. *Mon. Weather Review.*, 91:99–164, 1963.

[74] J. C. Strikwerda. *Finite difference schemes and partial differential equations.* Chapman and Hall/CRC, 1999.

[75] H. Tennekes and J. L. Lumley. *A first course in turbulence.* Cambridge, 1972.

[76] J. F. Thompson, Z. U. A. Warsi, and C. W. Mastin. *Numerical Grid Generation, Foundations and Applications.* Elsevier Science Publishing Co.,Inc, 1985.

[77] A. A. Townsend. *The structure of turbulent shear flow.* Cambridge university press, second edition, 1976.

[78] B. Vreman, B. Geurts, and H. Kuerten. Realizability conditions fot the turbulent stress tensor in large-eddy simulation. *Journal of Fluid Mechanics*, 278:351–362, 1994.

[79] D. C. Wilcox. *Turbulence Modeling for CFD.* DCW Industries, Inc. La Canada, California, 1994.

[80] P. Woodward and P. Colella. The numerical simulation of two-dimensional fluid flow with strong shocks. *Journal of Computational Physics*, 54:115–173, 1984.

[81] H. C. Yee and B. Sjögreen. Development of low dissipative high order filter schemes for multiscale Navier-Stokes/MHD systems. *Journal of Computational Physics*, 225(1):910–934, 2007.

Part I

Pulsed TV Holography and Schlieren Studies, and Large Eddy Simulations of a Turbulent Jet Diffusion Flame

Part II

Large Eddy Simulation of Supersonic Axisymmetric Baseflow

Part III

Large Eddy Simulation of a Turbulent Non-premixed Flame

Part IV

A Cartesian Embedded Boundary Method for the Compressible Navier-Stokes Equations

Part V

On the Accuracy of an Immersed/Embedded Boundary Method for Compressible Turbulent flow

Part VI

LES Baseflow simulations using a Cartesian Embedded Boundary Method