Development of an efficient Navier-Stokes/LES solver on unstructured grids for high-order accurate schemes

Thesis submitted in fulfillment of the requirements for the award of the degree of Doctor in de Ingenieurswetenschappen (Doctor in Engineering) by

Matteo Parsani

November 2010

Advisor:  Prof. Dr. Ir. Chris Lacor
Abstract

Researchers are attempting to tackle problems which were considered too ambitious just a few years ago. Multidisciplinary analysis and design (MAD), computational aeroacoustics (CAA), large eddy simulation (LES) and direct numerical simulation (DNS) of turbulence are examples of what is being attempted today. Improvements to the efficiency of these solutions are necessary due to the complexity of such problems.

In the field of the computational fluid dynamics (CFD), the use of higher-order accurate spatial discretizations for unstructured grids offers a possible avenue for improving the predictive simulation capabilities for many modern applications. This is due to the fact that higher-order methods exhibit a faster asymptotic convergence rate in the discretization error than lower (second)-order accurate finite volume (FV) and finite difference (FD) methods. The expectation is that an efficient higher-order discretization may provide an alternate path for achieving high accuracy in a flow with a wide disparity of length scales at reduced cost, by avoiding the use of excessive grid resolution.

Although the formulation of compact discretization strategies for higher-order methods such as discontinuous Galerkin (DG), spectral volume (SV) and spectral difference (SD) methods are now fairly well understood, the development of techniques for efficiently solving the discrete equations arising from these methods has generally been lagging. This is partly due to the complex structure of the discrete equations originating from fairly sophisticated discretization strategies, as well as the current application of higher-order methods to problems where simple explicit time-stepping schemes are thought to be adequate solution mechanisms such as acoustic phenomena. Therefore, the development of optimal, or near optimal solution strategies for higher-order discretizations, including steady-state solutions methodologies, and implicit time integration strategies, remains
then one of the key determining factors in devising higher-order methods.

The main goal of the present PhD research is to develop an efficient Navier-Stokes/LES solver on unstructured grids for high-order accurate spatial discretizations, and build up the necessary know-how to make a high-order accurate solver for industrial purposes. In order to achieve that, the present research has been carried out in two parts. In the first part, two implicit time integration schemes, namely backward Euler (BE) scheme and second-order backward difference formula (BDF2), are coupled with a non-linear lower-upper Gauss-Seidel (LU-SGS) algorithm for efficiently solving the discrete equations arising from the spatial discretization with a SV or a SD method. The non-linear LU-SGS algorithm with the BE scheme is evaluated both with analysis and computation for both spatial operators and steady flow problems. In addition, the capabilities and the advantages of the SD method in combination with the implicit time integration/algebraic solver technique is demonstrated by solving several unsteady reference test cases. Good agreement between the present results and reference solutions is found, demonstrating the potential benefits of high-order accurate spatial methods.

In the second part, the SD method coupled with a LES approach is investigated. The wall-adapted local eddy-viscosity (WALE) model is chosen as a subgrid-scale model and a new idea is presented for the definition of the filter width in the closure of the LES equations. The method is successfully applied to compute two- and three-dimensional turbulent cases. Good agreement between the present numerical results and reference solutions is observed, showing the capability and the quality of the new coupling approach.
# Jury members

**President**

Prof. Hugo SOL  
Vrije Universiteit Brussel

**Vice-president**

Prof. Rik PINTELON  
Vrije Universiteit Brussel

**Secretary**

Prof. Patrick GUILLAUME  
Vrije Universiteit Brussel

**Internal members**

Prof. Gert DESMET  
Vrije Universiteit Brussel

**External members**

Prof. Wim DESMET  
Katholieke Universiteit Leuven  
Prof. Eli TURKEL  
Tel Aviv University

**Advisor**

Prof. Chris LACOR  
Vrije Universiteit Brussel
Contents

1 Introduction .................................................. 1

2 Literature survey ........................................... 7
   2.1 Spectral volume method .............................. 8
   2.2 Spectral difference method ......................... 9
   2.3 Time integration schemes ......................... 10
   2.4 Geometric and $p$-multigrid methods ............... 11

3 Governing equations ....................................... 13
   3.1 Compressible Navier-Stokes equations ............. 14
      3.1.1 Newtonian fluid ............................... 16
      3.1.2 Thermal conductivity ......................... 17
      3.1.3 Thermodynamic properties: ideal gas model .... 17
      3.1.4 Formulation in Cartesian space ............... 20
      3.1.5 Dimensionless numbers ......................... 23
   3.2 Large eddy simulation ............................... 24
      3.2.1 Formulation in Cartesian space ............... 26
      3.2.2 The wall-adapted local eddy-viscosity model .... 28
   3.3 Boundary conditions .................................. 30
      3.3.1 Far field ....................................... 30
      3.3.2 Inlet mass density and velocity ............... 32
      3.3.3 Pressure outlet ................................ 33
      3.3.4 Solid wall .................................... 33
   3.4 Aerodynamic coefficients ........................... 36
   3.5 Linear convection equation ......................... 37

4 Spatial discretization ................................... 39
   4.1 Spectral volume method ............................. 40
      4.1.1 Discretization of convective term ............. 40
      4.1.2 SV basis polynomials .......................... 43
4.1.3 Discretization of diffusive terms .............................. 45
4.1.4 Spectral volume partition .................................... 47
4.2 Spectral difference method ..................................... 52
  4.2.1 Discretization of convective term .......................... 52
  4.2.2 SD basis polynomials ....................................... 55
  4.2.3 Discretization of diffusive terms ............................ 56
  4.2.4 Component-wise flux point distribution ..................... 58
  4.2.5 Solution and flux points distribution ....................... 59
  4.2.6 Grid filter width for the subgrid-scale model ......... 62
4.3 Concluding remarks ............................................. 64

5 Time discretization ................................................. 67
  5.1 Backward Euler scheme ........................................ 70
  5.2 Second-order backward difference formula ................. 74
  5.3 Time step ..................................................... 75

6 Analysis of the non-linear LU-SGS algorithm ................... 77
  6.1 Summary of the methodology .................................. 77
  6.2 SV method for triangular cells ................................ 82
    6.2.1 Second-order SV method ................................ 83
    6.2.2 Third-order SV method .................................. 92
    6.2.3 Fourth-order SV method .................................. 94
  6.3 SD method for quadrilateral cells ............................ 95
    6.3.1 Second-order SD method ................................ 95
    6.3.2 Third-order SD method .................................. 96
    6.3.3 Fourth-order SD method ................................ 98
  6.4 Remarks ..................................................... 99

7 Application I: spectral volume method .......................... 101
  7.1 Two-dimensional laminar steady flow simulations .......... 102
    7.1.1 Flow over a circular cylinder ............................ 104
    7.1.2 Flow over a NACA0012 airfoil ......................... 110
    7.1.3 Flow in a channel with a backward-facing step ...... 116

8 Application II: spectral difference method ...................... 121
  8.1 Steady laminar flow simulations .............................. 122
    8.1.1 Flow over a NACA0012 airfoil ............................ 122
    8.1.2 Flow through a 90° bending square duct ............... 127
  8.2 Unsteady laminar flow simulations .......................... 133
    8.2.1 Flow over an open cavity ................................ 133
    8.2.2 Flow past a square cylinder ............................. 137
    8.2.3 Flow past a circular cylinder ............................ 141
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>BDF2</td>
<td>Second-order backward difference formula</td>
</tr>
<tr>
<td>BDF3</td>
<td>Third-order backward difference formula</td>
</tr>
<tr>
<td>BDF4</td>
<td>Fourth-order backward difference formula</td>
</tr>
<tr>
<td>BE</td>
<td>Backward Euler scheme</td>
</tr>
<tr>
<td>BR2</td>
<td>Second approach of Bassi and Rebay</td>
</tr>
<tr>
<td>CAA</td>
<td>Computational aeroacoustics</td>
</tr>
<tr>
<td>CEM</td>
<td>Computational electromagnetics</td>
</tr>
<tr>
<td>CFD</td>
<td>Computational fluid dynamics</td>
</tr>
<tr>
<td>CFL</td>
<td>Courant-Friedrichs-Lewy</td>
</tr>
<tr>
<td>CV</td>
<td>Control volume</td>
</tr>
<tr>
<td>DES</td>
<td>Detached eddy simulation</td>
</tr>
<tr>
<td>DG</td>
<td>Discontinuous Galerkin</td>
</tr>
<tr>
<td>DNS</td>
<td>Direct numerical simulation</td>
</tr>
<tr>
<td>DOF</td>
<td>Degree of freedom</td>
</tr>
<tr>
<td>E-RK</td>
<td>Explicit Runge-Kutta scheme</td>
</tr>
<tr>
<td>ESDIRK</td>
<td>Explicit-first-stage, single-diagonal-coefficient, diagonally-implicit Runge-Kutta scheme</td>
</tr>
<tr>
<td>FE</td>
<td>Forward Euler scheme</td>
</tr>
<tr>
<td>FV</td>
<td>Finite Volume</td>
</tr>
<tr>
<td>GMRES</td>
<td>Generalized minimal residual method</td>
</tr>
<tr>
<td>I-RK</td>
<td>Implicit Runge-Kutta scheme</td>
</tr>
<tr>
<td>LDG</td>
<td>Local discontinuous Galerkin</td>
</tr>
<tr>
<td>LES</td>
<td>Large eddy simulation</td>
</tr>
<tr>
<td>LSD</td>
<td>Local spectral difference</td>
</tr>
<tr>
<td>LSV</td>
<td>Local spectral volume</td>
</tr>
<tr>
<td>LU-SGS</td>
<td>Lower-upper symmetric Gauss-Seidel</td>
</tr>
<tr>
<td>N-S</td>
<td>Navier-Stokes</td>
</tr>
<tr>
<td>RANS</td>
<td>Reynolds averaged Navier-Stokes</td>
</tr>
<tr>
<td>RK</td>
<td>Runge-Kutta scheme</td>
</tr>
</tbody>
</table>
SD Spectral difference
SD-LES SD method coupled with LES approach
SGS Symmetric Gauss-Seidel
SSP Strong stability preserving
SV Spectral volume
TVD Total variation diminishing
WALE Wall-adapted local eddy-viscosity

Subscripts

c Reference quantity
|Ω| Boundary quantity
cc Current cell
gho Ghost value
I Imaginary part of a complex number
int Internal value
L Left cell
nb Neighbouring cells
R Real part of a complex number
R Right cell

Symbols

∩ Cross section
\vec{I}_k Unit vector in direction of the wave vector
\vec{\xi}_n Unit normal to a face in a cell-mapped coordinate system
α Damping factor in averaging operator for diff. term treatment
α_3 DOF of a 3rd-order SV partition/SD flux point distribution
α_4 DOF of a 4th-order SV partition
β Bias in the averaging operator for the diffusive term treatment
β_3 DOF of a 3rd-order SV partition
β_4 DOF of a 4th-order SV partition
\hat{W}_m Coefficients of expansion of \hat{W} in terms of eigenvectors \hat{V}_m
\hat{W}_0 Coefficients of initial solution expansion in terms of eigenvectors \hat{V}_m
\vec{\chi} Vector of general Cartesian coordinates \left[\chi_1, \chi_2, \chi_3\right]^T, m
\chi_1, \chi_2, \chi_3 General coordinates in Cartesian space, m
\Delta \Omega_j^{\vec{\xi}} Volume of CV j in the mapped coordinate system \vec{\xi}
\Delta B Length of \vec{B}_1, m
\delta W_{L(R)} Polynomial used in the BR2 lifting operator definition for SD
\Delta Grid filter width for LES
δ_4 DOF of a 4rd-order SV partition
δ jm Kronecker delta function

xii
\( \gamma_3 \) DOF of a 3rd-order SV partition
\( \gamma_4 \) DOF of a 4th-order SV partition
\( \Lambda \) Lifting operator used with the BR2 diff. term treatment
\( \hat{\nabla} \xi \) Divergence operator in mapped coordinate system
\( \hat{\Phi} \) Averaged gradient approximations on a face
\( \varphi \) Riemann flux upwinding parameter
\( \Phi \) Polynomial approximation of the conserved variable gradients
\( \psi \) 2D advection speed vector orientation angle
\( \sigma \) Courant-Friedrichs-Lewy (CFL) number, dimensionless
\( T \) Spatial discretization matrix for linear advection equation
\( T^{+1,0} \) Spatial discretization matrix for linear advection equation
\( T^{-1,0} \) Spatial discretization matrix for linear advection equation
\( T^{0,+1} \) Spatial discretization matrix for linear advection equation
\( T^{0,-1} \) Spatial discretization matrix for linear advection equation
\( T^{0,0} \) Spatial discretization matrix for linear advection equation
\( T' \) Spatial discretization matrix for linear advection equation
\( \tau \) Parameter of the BDF2 with variable time step
\( G_d \) Amplification matrix for direct inversion method
\( G_{QD} \) Amplification matrix for 1-st forward GS sweep for quadrilateral cells
\( G_{TR} \) Amplification matrix for 1-st forward GS sweep for triangular cells
\( G_{QD}^{f,m} \) Amplification matrix for \( m \)-th forward GS sweep for quadrilateral cells
\( G_{TR}^{f,m} \) Amplification matrix for \( m \)-th forward GS sweep for triangular cells
\( G_{SGS,1}^{QD} \) Amplification matrix for 1-st SGS sweep for quadrilateral cells
\( G_{TR}^{SGS,1} \) Amplification matrix for 1-st SGS sweep for triangular cells
\( G_{SGS,m}^{QD} \) Amplification matrix for \( m \)-th SGS sweep for quadrilateral cells
\( G_{TR}^{SGS,m} \) Amplification matrix for \( m \)-th SGS sweep for triangular cells
\( \theta \) 2D wave vector orientation angle
\( \Theta_m \) Eigenvalue of matrix \( T \)
\( \Theta'_m \) Eigenvalue of matrix \( T' \)
\( \varepsilon \) Small constant used for the calculation of Jacobians
\( \varepsilon_{mach} \) Machine zero
\( \xi \) Vector of mapped coordinates \([\xi_1, \xi_2, \xi_3]^T\)
\( \xi_1, \xi_2, \xi_3 \) Mapped coordinates
\( \xi_l \) Mapped coordinates of flux point \( l \)
\( \xi_j \) Mapped coordinates of solution point \( j \)
\( a \) Module of advection speed, \( ms^{-1} \)
\( C_D \) Drag coefficient, dimensionless
\( C_L \) Lift coefficient, dimensionless
\( C_P \) Pressure coefficient, dimensionless
\( C_f \) Skin friction coefficient, dimensionless
\( \vec{B}_1 \) Vector defining a 2D generating pattern, \( m \)
\( \vec{B}_1' \) Dimensionless vector defining a 2D generating pattern
\( \vec{B}_2 \) Vector defining a 2D generating pattern, \( m \)
\( \vec{B}_2' \) Dimensionless vector defining a 2D generating pattern
\( \rho \) Mass density, \( kg \, m^{-3} \)
\( \mathbf{w} \) Conserved variables
\( \text{vars} \) Number of the scalar conserved variables, \( \text{vars} = 4 \) in 2D, \( \text{vars} = 5 \) in 3D
\( \vec{m} \) Momentum, \( kg \, m^{-2} \, s^{-1} \)
\( |\vec{m}| \) Module of the momentum, \( kg \, m^{-2} \, s^{-1} \)
\( \vec{u} \) Velocity vector, \( m \, s^{-1} \)
\( |\vec{u}| \) Module of the velocity vector, \( m \, s^{-1} \)
\( \text{dim} \) Dimensions of the physical space, \( \text{dim} = 2 \) in 2D, \( \text{dim} = 3 \) in 3D
\( M \) Mach number
\( Pr \) Prandtl number, dimensionless
\( Pr_t \) Turbulent Prandtl number, dimensionless
\( Re \) Reynolds number
\( \Omega \) Computational domain
\( \partial \Omega \) Boundary of the computational domain
\( \mathbf{F}_{i,\xi} \) Polynomial approximation of \( f_{i,\xi} \)
\( f_{C,i} \) \( \xi_1 \)-components of convective flux vectors in mapped coordinate system
\( \tilde{f}_{C,i} \) Convective flux vector projected in mapped coordinate system
\( \hat{\Delta}_{i,l} \) Averaged filter width of the SD face flux point \( l \) in cell \( i \), for LES
\( \langle \cdot \rangle \) Favre filtered quantity
\( \langle \cdot \rangle \) Spatially filtered quantity
\( \vec{q} \) Heat flux vector, \( Pa \, m \, s^{-1} \)
\( \mathbf{f} \) Flux vector
\( \tilde{\mathbf{f}} \) Convective flux vector
\( \mathbf{f}_D \) Diffusive flux vector
\( \tilde{f}_\rho \) Mass density flux vector, \( kg \, m^{-2} \, s^{-1} \)
\( \tilde{f}_m \) Momentum flux tensor, \( Pa \)
\( \tilde{f}_e \) Total energy flux vector, \( Pa \, m \, s^{-1} \)
\( \varphi^R \cdot \vec{n} \) Riemann flux through a face with unit normal \( \vec{n} \)
\( G \) Amplification factor of a full discretization in space and time
Polynomial approximation of $\mathbf{g}_C^\xi$

$\xi_2$-components of convective flux vectors in mapped coordinate system

$\gamma$ Characteristic gas constant. It is the specific heat capacities ratio, for the ideal gas model

$R$ Specific gas constant defined by the ratio of the universal gas constant and the molar mass of the gas

$h_f$ Local length scale associated to a face, $m$

Polynomial approximation of $\mathbf{h}_C^\xi$

$\xi_3$-components of convective flux vectors in mapped coordinate system

$I$ Imaginary unit number, square root of $-1$

$\Delta_l$ Filter width of the SD internal flux point $l$ in cell $i$, for LES

$K$ Dimensionless wave number

$k$ Wave number, $m^{-1}$

$L^s_j$ Basis polynomial associated to solution point $j$

$L^f_l$ Basis polynomial associated to flux point $l$

$\bar{L}_{i,j}$ SV basis polynomial with index $j$ in cell $i$

$N^f$ Number of flux points in a cell

$N^s$ Number of solution variables in a cell

$N^s_{GP}$ Number of solution variables in a generating pattern

$N_{RK}$ Number of stage for a Runge-Kutta scheme

$\nabla$ Nabla/del operator, $m^{-1}$

$\frac{\partial}{\partial t}$ Partial derivative with respect to time, $s^{-1}$

$r$ Position vector in a general coordinate system, $m$

$A_c$ Reference surface of an object, $m^2$

$\mathbb{C}$ Set of complex number

$\mathbb{R}$ Set of real number

$St$ Strouhal number, dimensionless

$\mathbf{G}$ Stress tensor, $Pa$

$\bar{\sigma}$ Deviatoric stress tensor. It becomes the viscous stress tensor when $\zeta$ is set equal to the pressure $P$, $Pa$

$\bar{S}$ Symmetric velocity gradient tensor, $s^{-1}$

$\bar{I}$ Isotropic stress tensor characterized by the scalar $\zeta$

$\zeta$ Scalar number characterizing the isotropic stress tensor $\bar{I}$. It is set equal to the pressure $P$, $Pa$

$k^{sgs}$ Subgrid-scale kinetic energy, $J kg^{-1}$

$\bar{\tau}^{sgs}$ Subgrid-scale stress tensor, $Pa$

$\bar{q}^{sgs}$ Subgrid-scale heat flux vector, $Pa m s^{-1}$
Conductivity coefficient, $J m^{-1} K^{-1}$
Dilatation viscosity coefficient, $kg m^{-1} s^{-1}$
Dynamic viscosity coefficient, $kg m^{-1} s^{-1}$
Electro-chemical potential, $J mol^{-1}$
Entropy, $J K^{-1}$
Internal energy, $J$
Kinematic viscosity coefficient, $m^2 s^{-1}$
Turbulent kinematic viscosity coefficient (eddy-viscosity), $m^2 s^{-1}$
Number of moles in a single-component fluid
Pressure, $Pa$
Specific heat capacity at constant pressure, $J kg^{-1} K^{-1}$
Specific heat capacity at constant volume, $J kg^{-1} K^{-1}$
Specific entropy, $J kg^{-1} K^{-1}$
Specific internal energy, $J kg^{-1}$
Specific volume, $m^3 kg^{-1}$
Speed of sound, $m s^{-1}$
Temperature, $K$
Volume, $m^3$
Time, $s$
Time averaged quantity
Specific total energy, $J kg^{-1}$
Eigenvector of matrix $T'$
Complex amplitude of the numerical spatial Fourier wave
Averaged conserved variables, used for diffusive term treatment
Conserved variables in mapped coordinate system in cell $i$
SD solution variable, solution at solution point $j$ in cell $i$
Polynomial of degree $p + 1$, for gradient computation with SD
Solution polynomial in cell $i$
SV solution variable, averaged solution in CV $j$ in cell $i$
A complex number

**Superscripts**

* Latest available solution
0 Initial quantity
LES Quantity related to large eddy simulation (LES)
$s_{gs}$ Subgrid-scale quantity
$T$ Transpose
Chapter 1

Introduction

Basic forms of computational fluid dynamics (CFD) were established in 1960 with the research at the Courant Institute on hyperbolic systems of conservation laws which led to the development of the first second-order accurate dissipative methods by Lax and Wendroff. This class of methods forms the basis of the efficient explicit method developed in 1969 for the Navier-Stokes equations by MacCormack. At the Douglas Aircraft Company, the aerodynamic research group, led by A. M. O. Smith, developed the first panel method for three-dimensional, linear, potential flows, and the first implementation of Kellers box method for turbulent boundary layer flows. CFD began developing quickly in the 1980’s as the advances of computing and processor technology gave scientists the means to solve complex fluid dynamics problems. This has enabled progress at many fronts, including numerical algorithms for the Euler and Navier-Stokes equations, grid generation and adaptation, turbulence modeling, flow visualization as well as the basic understanding of complex phenomena and better design quality of technological applications. Nowadays, CFD has reached a high level of sophistication, and at the same time its range of applications is broadening, including diverse topics such as aerodynamics, aeroacoustics, combustion, global atmospheric modeling, oceanography etc. In some circumstances and especially in the early stages of the design process, experiments might be too expensive or even impossible to perform. Therefore, modeling is the only reasonable way to get answers and to study a range of parameters for optimal design. In addition, CFD is used routinely to complement wind tunnel tests for the final design of compressors, turbines, pumps as well as complete aircrafts.
In the field of CFD, spatially low-order (first- and second-order) numerical methods, typically based on finite volume (FV) or finite difference (FD) discretizations, are less accurate than their high-order counterpart, but they are generally more robust and reliable; as a result, they are routinely employed in commercial flow solver software packages. In fact, for many industrial application problems, these methods are often good choices, when considering the balance between computational speed, simplicity of coding, and resolution required. This statement is especially valid if sought after solutions are piecewise simple (almost linear) with several isolated discontinuities in between, for example, the solution of most Riemann problems. However, in certain modern applications with complicated geometries and complex physics, where the accurate resolution of small scales is required (e.g. direct numerical simulation (DNS), large eddy simulation (LES), computational aeroacoustics (CAA), computational electromagnetic (CEM), turbulent combustion etc.), the solution structures are so complicated and the time of evolution of these structures is so long that it is impractical to use low-order methods to obtain an acceptable resolution. Low-order methods are in fact too dissipative to resolve accurately rich structures in the smooth part of the solutions. Therefore, it should be decided, based on the problem at hand, whether to use a high-order scheme (order of accuracy >2) or to find a suitable first- or second-order scheme.

In addition, since CFD is increasingly used as an industrial design and analysis tool, high-order accuracy must be achieved on unstructured grids which are required for efficient meshing of complex geometries. With classical spatial discretizations, such as FV and FD methods, high-order accuracy can be obtained theoretically for an arbitrary unstructured grid by using high-order polynomial data reconstructions. However, higher than linear reconstructions are rarely used for three dimensions in practice. This is mainly because of the difficulty in finding valid (non-singular) stencils and the enormous memory required to store the coefficients used in the reconstruction. For each cell, the stencil is unique for an unstructured grid and its size increases non-linearly with the order of accuracy. A data reconstruction must be performed at each iteration for each element. This reconstruction step is the most memory consuming in higher than second-order schemes. In addition, the stencil size complicates numerical formulations near boundaries, increases the matrix bandwidth and can increase the communication time required by algorithms for parallel architectures.

All the needs mentioned above have been the driving force for the development of a new class of spatially high-order schemes for unstructured grids,
e.g. the discontinuous Galerkin (DG) method, the spectral volume (SV) method and the spectral difference (SD) method. Such methods approximate the solution by a polynomial of a certain degree in each cell by providing sufficient pieces of independent information, i.e. degrees of freedom (DOFs), in the cell itself. These schemes use piecewise continuous functions as the solution approximation space. They have a compact stencil, since only the data local to the cell and the data of its immediate neighboring cells are required for the evaluation of the fluxes. Consequently, they are easily parallelizable. In this thesis, two spatially compact methods, namely the SV and SD methods, are used for the spatial discretization of the fluid dynamics equations.

When high-order schemes are combined with classical solution methods, such as explicit Runge-Kutta (E-RK) solvers, they suffer from a restrictive Courant-Friedrichs-Lewy condition or CFL condition and hence a relatively slow convergence rate. In addition, the solver should also be able to deal with the geometrical stiffness imposed by the Navier-Stokes grids where high-aspect ratios occur near walls. In the case of compressible solvers there is an additional stiffness when solving for low speed flows caused by the disparate eigenvalues of the system. Therefore, the development of optimal, or near optimal solution strategies for higher-order discretizations, including steady-state solutions methodologies, and implicit time integration strategies, remains one of the key determining factors in devising higher-order methods which are not just competitive but superior to lower-order methods in overall accuracy and efficiency. Implicit time-integration schemes can be used to deal with these problems. They can advance the solution with significantly larger time steps compared to explicit methods. Two implicit time marching schemes, namely the backward Euler (BE) scheme and second-order backward difference formula (BDF2) with variable time step, are used here for the time discretization. The BE scheme, which is first-order accurate in time, is used to advance in (pseudo) time steady flow problems, whereas the BDF2 is used for unsteady flow simulations.

Implicit temporal schemes imply the solution of non-linear algebraic systems. Therefore, if the algorithm for such systems is not efficient, implicit schemes might be more expensive than explicit ones. In this work, an efficient algebraic solver, namely the non-linear lower-upper symmetric Gauss-Seidel algorithm (LU-SGS), is used to solve the non-linear algebraic systems associated to the implicit time discretizations. This algorithm was proposed a few years ago in combination with the BE scheme for the SD
method. It solves the non-linear algebraic system through multiple cell-wise symmetric forward and backward Gauss-Seidel sweeps. Because of the Gauss-Seidel nature of this algorithm, where the latest available solution in the neighboring cells is used to update the solution in a cell, information travels much faster across the domain than with a traditional explicit solver, where only the solution at the previous time step or stage is used to update the solution in a cell. Besides its cell-wise implicitness, the Gauss-Seidel nature is one of the reasons why the non-linear LU-SGS method is much more efficient than an explicit solver. The coupling of the non-linear LU-SGS solver with the SV and SD methods and the evaluation of its performance both with analysis and computation are two of the main cores of this thesis.

Although spatially high-order accurate numerical schemes guarantee a better resolution of small scales than low-order ones (the latter require more grid points to achieve the same level of accuracy), their application to the simulation of general turbulent flows implies that particular attention has still to be paid to subgrid models. Therefore, since this PhD work has been accomplished in the framework of the IWT Project SBO 050163 ("Simulation and design tools towards the reduction of aerodynamic noise in confined flows"), and its final goal was the development of an efficient N-S/LES solver for high-order accurate schemes, the SD scheme has been coupled with the wall-adapted local eddy-viscosity (WALE) model to perform large eddy simulations (SD-LES). The development and the evaluation of the accuracy and the reliability of the implicit SD-LES solver are also two main cores of this thesis. It should be noted that the coupling SV-LES was not treated because no stable three-dimensional SV partition for third-order accurate scheme seems to exist.

The remainder of this thesis is organized as follows. A survey of the available literature on the SV and SD methods, time integration/iterative solution approaches for spatially high-order methods and multigrid algorithms is included in Chapter 2. The governing equations that describe the flow problems considered in this thesis, namely the compressible Navier-Stokes equations and the filtered compressible Navier-Stokes equations for LES, are presented in Chapter 3. The WALE model for the closure of the subgrid-scale terms is also discussed. Chapter 4 is devoted to the description of the SV and SD methods and the coupling of the latter scheme with the WALE model through a new definition of the grid filter width. The non-linear LU-SGS algebraic solver, in combination with the BE scheme and the BDF2 is presented in Chapter 5. In the same chapter, the lo-
A Von Neumann stability analysis for the LU-SGS algorithm with the BE scheme and for general SV and SD methods, respectively on triangular and quadrilateral grids, is described in Chapter 6. In Chapter 7 the main results achieved by applying the SV method and the nonlinear LU-SGS algorithm with the BE scheme to problems governed by the two-dimensional steady compressible Navier-Stokes equations are presented. In order to assess the convergence properties of the non-linear LU-SGS algorithm, the performance of the latter scheme is compared with that of a family of E-RK smoothers. Both time marching schemes are coupled with a $p$-multigrid algorithm to accelerate the convergence to steady state solutions. Chapter 8 deals with the main results achieved by applying the SD method and the nonlinear LU-SGS algorithm to problems governed by 2D and 3D compressible Navier-Stokes and filtered Navier-Stokes equations. The convergence properties of the non-linear LU-SGS algorithm for steady flow simulations are assessed by comparing the algorithm’s performance with that of a Newton-Raphson GMRES algebraic solver. For the unsteady laminar and turbulent flow simulations, an extensive study of the accuracy and reliability of the implicit SD and SD-LES solvers is done by means of validation against experimental data and/or reference solutions available in literature. Finally, in Chapter 9, conclusions from the results presented this thesis are drawn and an outlook for future work is given.
Chapter 2

Literature survey

In this chapter an overview of the available literature on the spectral volume (SV) and the spectral difference (SD) methods, efficient time marching and algebraic solvers for spatially high-order methods and multigrid algorithms is given.

Both SV and SD methods are strongly related to the discontinuous Galerkin method (DG), which was introduced in 1973 by Reed and Hill [140] to solve the neutron transport equation. The development of the DG method for hyperbolic conservation laws was pioneered by Cockburn, Shu and their co-workers in a series of papers on the Runge-Kutta DG (RKDG) method [37, 39, 41, 43]. Nowadays, the DG method is the most popular and most developed high-order accurate method for unstructured grid. Bassi and Rebay demonstrated the capabilities of the DG method by achieving high-order accuracy for the compressible Euler and Navier-Stokes (N-S) equations [15, 16]. For a comprehensive review of the DG history and literature the interested reader is referred to Cockburn et al. [38].

Recently, Huynh [79, 80], Wang and Gao [52, 184, 185] and Haga et al. [61] have proposed a new high-order method for unstructured grids. This method is called lifting collocating penalty (LCP) approach and unifies several of the popular methods including the DG method, the SV method and the SD method with a technique that does not require the evaluation of any integrals. Consequently, the evaluation of the residuals is relatively cheap. If the parameters of the LCP method are chosen such that it is linearly equivalent to the DG method, then, like the DG method, it is linearly stable on general hybrid grids.
2.1 Spectral volume method

The SV method, with applications to one-dimensional (1D) scalar conservation laws, was proposed in 2002 by Wang \cite{183} as an alternative for the DG method. Further development of the SV method for two-dimensional (2D) and for non-linear hyperbolic systems, such as the Euler equations, was then reported in subsequent papers by Wang et al. \cite{186, 187, 191}. The SV method was successfully extended to 2D N-S equations, and three-dimensional (3D) Maxwell equations respectively by Sun et al. \cite{164} and Liu et al. \cite{104}. Chen \cite{32, 33} developed many high-order SV partitions for simplex in 2D and 3D with relatively small Lebesgue constants. The appropriate treatment of curved wall boundaries for all high-order methods, was addressed for the 2D SV method by Wang and Liu \cite{188}, by using a high-order geometric mapping of the SV cells near the boundaries. Comparisons between the SV and DG methods were made by Sun and Wang \cite{163} and by Zhang and Shu \cite{199}. The SV method was also applied to solve the 3D Euler and N-S equations by Haga et al. \cite{62} on Japan’s Earth Simulator. A positive step towards addressing the issue of stability was given by Van den Abeele et al. \cite{174, 176}, who performed Fourier analysis for both 1D and 2D SV methods, and identified a weak instability in several SV partitions. New partitions were derived which showed improved stability properties. In addition, Harris et al. \cite{67} developed a more efficient quadrature free implementation for the SV method, which was significantly faster than the standard quadrature-based SV method. Different approaches for the discretization of the diffusive terms in the N-S equations with the SV method, based on similar approaches that were developed for the DG method, were investigated in Kannan and Wang \cite{90}. In 2009, Van den Abeele et al. \cite{175} performed an extensive study of the variation of the stability properties of the SV method on tetrahedral grids. The study indicates that probably there is no partition that yields a stable third-order SV scheme for tetrahedral cells. In 2010, Harris and Wang \cite{66} have presented a constrained minimization approach in the design of 3D third-order SV schemes. Several new partitions were proposed which have a reduced maximum real part of the Fourier footprint by up to 20% over the original un-optimized partition proposed by Chen \cite{32}. Numerical simulations have shown that the strength of the instability has been weakened by about an order of magnitude for some cases by employing the constrained minimization approach. However, also in this case, no fully stable third-order SV scheme for tetrahedral cells has been found.
2.2 Spectral difference method

The SD method originated in the staggered grid multi-domain spectral method proposed by Kopriva and Kolas [94] and Kopriva [93]. It was generalized to simplex elements by Liu et al. [103], who applied the SD method to 2D scalar conservation laws and the Maxwell equations. Extension of the SD method to the Euler equations was described in Wang et al. [189], and to the N-S equations in May and Jameson [111] and Wang et al. [190]. A 3D N-S implementation of the SD method for hexahedral grids was presented by Sun et al. [165]. Different approaches for the discretization of the diffusive terms in the N-S equations with the SD method, based on similar approaches that were developed for the DG method, were investigated by Van den Abeele et al. [178]. Huang et al. [77] reported an implicit space-time implementation of the SD method for discontinuity capturing using adaptive polynomials. Huynh [79] showed that for quadrilateral and hexahedral cells, tensor product flux point distributions based on a 1D flux point distribution consisting of the end points and the Legendre-Gauss quadrature points, lead to stable schemes for arbitrary order of accuracy. In 2008, Van den Abeele et al. [177] showed an interesting property of the SD method, namely that it is independent of the positions of its solution points. Recently, this property has been proved by Jameson [85]. In the same paper Jameson also has proved that for the case of one dimensional linear advection the SD method is stable for all orders of accuracy in a norm of Sobolev type, provided that the interior fluxes collocation points are placed at the zeros of the corresponding Legendre-Gauss polynomial, as reported by Huynh [79].

The performance of the SD method for turbulent flow simulations was investigated by Liang and his co-workers [102] and Parsani et al. [129], with an unresolved DNS type approach. More recently, Parsani et al. [122, 124, 126] coupled, for the first time, a high-order SD scheme with the local eddy-viscosity (WALE) model to perform 2D and 3D large eddy simulations. The 3D LES was performed to simulate the turbulent flow in a muffler which is also one of the benchmark test cases of the IWT Project SBO 050163. The coupling of the SD scheme with the WALE model, through a new definition of the grid filter width for high-order SD schemes, is discussed in Section 4.2.6. The accuracy and the reliability of the SD-LES approach are discussed in Section 8.3, where the turbulent flow simulations are presented.

Application of the SD method in combination with a perfectly matched
layer (PML) approach was reported by Zhou and Wang [201], for the simulation of computational aeroacoustics (CAA) benchmark problems. More recently, Parsani and his co-workers [123, 125] have used the LES-high-order spectral difference method to provide the acoustic sources for aerodynamic sound field simulations with a Ffowcs-Williams Hawkings (FW-H) code.

2.3 Time integration schemes

High-order spatial operators are usually much stiffer than lower-order ones. For time accurate problems, the allowable Courant-Friedrichs-Lewy (CFL) number usually decreases with increasing order of accuracy for explicit schemes [73]. For viscous problems, where cells with high-aspect ratio occur near the walls, resolving the viscous boundary layer with an economical distribution of grid points is severely limited by the time step size for explicit high-order time marching schemes, and usually not competitive against low-order implicit methods in terms of efficiency, as demonstrated in Venkatakrishnan and Mavriplis [181]. In fact, the computational cost of high-order explicit time integration methods for many steady-state problems is so high that they become less efficient than low-order implicit methods in terms of the total CPU time, given the same level of solution error. Many types of implicit algorithms have been successfully developed for unstructured grid-based solvers in the last two and a half decades. Non-linear element-Jacobi, as well as linearized element-Jacobi and Gauss-Seidel schemes were used by e.g. Nastase and Mavriplis [115] for the DG method. Bassi and Rebay [15] successfully developed a preconditioned Newton-Raphson GMRES solvers for the DG method to solve the compressible Navier-Stokes equations. In Van den Abeele et al. [178, 179] a preconditioned Newton-Raphson GMRES approach was applied to SV and SD methods. A matrix-free Krylov method was applied to DG schemes by Rasetariner and Hussaini [139] and to SD schemes by May et al. [110]. The non-linear lower-upper symmetric Gauss-Seidel (LU-SGS), developed by Chen and Wang [34], was used with SD schemes by Sun et al. [166, 168] and Van den Abeele et al. [178], with SV schemes by Parsani et al. [128], Kannan et al. [90] and Haga et al. [63]. Line-implicit solvers were developed by Fidkowski et al. [48] for their DG method.

The present author performed an extensive study of the smoothing properties of the non-linear LU-SGS solver in combination with the backward Euler scheme for SV schemes. This analysis was reported in Parsani et al.
2.4. GEOMETRIC AND P-MULTIGRID METHODS

[127, 130] and discussed in Chapter 6. Recently, the non-linear LU-SGS algorithm with the BDF2 was used to perform 2D and 3D unsteady laminar and turbulent flow simulations in Parsani et al. [122, 129]. An application of the non-linear LU-SGS solver to implicit Runge-Kutta schemes was reported by Parsani et al. [128], where explicit-first-stage, single-diagonal-coefficient, diagonally-implicit Runge-Kutta (ESDIRK) schemes were used to advance the quasi-1D Euler equations in pseudo time. In the latter work a SV scheme was used for the spatial discretization. A short description of the non-linear LU-SGS solver in combination with ESDIRK schemes has been included in Appendix D.

2.4 Geometric and $p$-multigrid methods

One powerful solution strategy for solving large scale problems in fluid dynamics is multigrid [24, 25, 60, 172]. The standard multigrid algorithm (geometric multigrid or $h$-multigrid) has been used very effectively in computational fluid dynamics (CFD) to accelerate the rate of convergence to steady state. In its implementation, multiple levels of coarse grids are generated either from the finer grids or independently, with each coarser grid doubling the mesh size of the next finer mesh in all directions. Jameson and Caughey [86] demonstrated that an Euler solution for airfoil flows, converged to the level of the truncation error, could be obtained in 3-5 multigrid cycles. However, several factor contribute to the effectiveness of the multigrid approach. Multigrid methods for hyperbolic problems depend on two elements to accelerate convergence. One element is the smoothing of high-frequency components of the solution error. The choice of an iterative scheme for smoothing is crucial, since multigrid requires a smooth solution error to approximate a fine grid problem on a coarser grid. In addition, the smoother must be effective on the coarser grids, since these grids are responsible for removing the low-frequency error modes that cause slow asymptotic convergence of iterative schemes. The second element for accelerating convergence is the expulsion of errors on the coarse grids, which occur faster for time-like iterative methods due to the larger time steps permitted on coarser grids.

The most critical element of a successful multigrid approach is the development of an effective smoother to remove various types of stiffness in the spatial operator. For example, the stiffness due to low flow speed and grid anisotropy was known to degrade the performance of multigrid solvers. Many novel numerical techniques were developed to address these issues,
including low speed preconditioning [35, 173], line-implicit solvers [109], semi-coarsening [113], multigrid cycles with a preconditioned non-linear LU-SGS smoother [86] and multigrid cycles with a Runge-Kutta/Implicit scheme [149, 169].

In addition, $p$-multigrid, or multiorder, solution strategies can be used in an analogous manner to accelerate convergence of high-order methods to steady state by recursively iterating on solution approximations of different polynomial order. Therefore, lower-order spatial operators in a $p$-multigrid approach serve as the coarse grid operators in the $h$-multigrid counterpart. Lower-order operators have similar advantages: they have more numerical damping and allow larger time steps than high-order operators. The $p$ component of this algorithm was proposed by Rønquist and Patera [146] and analyzed by Maday and Munoz [107] for 1D Galerkin spectral element discretization of Laplace equation. Helenbrook [68] combined $p$-multigrid with standard low-order multigrid and applied it to an unstructured stream-wise-upwind-Petrov-Galerkin (SUPG) discretization of the incompressible Navier-Stokes equations. In the context of DG approximations, Hemker et al. [71] analyzed block Jacobi smoothing strategies with $h$-multigrid for 1D diffusion problems. In addition, $p$-multigrid, or multiorder, solution strategies have been studied for high-order DG by Helenbrook et al. [70], Bassi and Rebay [18] and Fidkowski et al. [48], showing several advantages such as ease of implementation and order-independent convergence rates. Nastase and Mavriplis [115] used a $hp$-multigrid approach for the DG method, showing convergence rates which are independent of both order of accuracy ($p$) of the discretization and level of mesh resolution ($h$). It was also applied to the SV method by Van den Abeele et al. [174], Parsani et al. [127, 128, 130] and Kannan et al. [90], and to the SD method by Premasuthan et al. [133] and May et al. [110]. In Liang et al. [101] a $p$-multigrid method was applied to the spectral difference method with explicit and implicit smoothers on unstructured triangular grids.
## Chapter 3

### Governing equations

In this chapter, the governing equations which mathematically describe all physical flow problems presented in this thesis are reviewed. Firstly, the complete set of convection-diffusion equations for the macroscopic motion of real fluid substances are presented. These equations correspond to the *compressible Navier-Stokes (N-S) equations*. Afterwards, the constitutive relation for *Newtonian fluids* is introduced. Air, which is the working fluid in the present work, belongs to this fundamental class of fluids. The system of equations for Newtonian fluids is then completed by modelling the thermodynamic properties of the fluid with the *ideal gas model*. This model compares well with the thermodynamic behavior of air, for the flow problems considered in this thesis.

Next, important non-dimensional parameters as *Reynolds number*, *Mach number* and the *Prandtl number*, which characterize the flow, are introduced. Successively, defining the concept of spatial filtering technique, the compressible N-S equations are presented in the framework of *large eddy simulation (LES)*. To close the LES system of equations, the *wall-adapting local eddy-viscosity (WALE) model* is introduced. For both N-S and LES equations the formulation in tridimensional Cartesian space is given.

Since the N-S equations describes an *initial-boundary value problem (IBVP)*, it must be equipped with appropriate initial conditions and boundary conditions. The boundary conditions used in this work are introduced after the presentation of the governing equations, whereas the initial conditions are specified only in the next chapters, where flow simulations are presented.
Dimensionless quantities like pressure coefficient, lift coefficient and the drag coefficient, which can be used to assess the accuracy of a numerical method to solve the fluid flow problems, are also discussed.

To conclude the chapter, the conservation law for the linear advection equation used in Chapter 6 to perform the Von Neumann analysis of the lower-upper symmetric Gauss-Seidel algorithm (LU-SGS) is defined.

### 3.1 Compressible Navier-Stokes equations

The compressible Navier-Stokes (N-S) equations are derived from the basic principles of conservation of mass, momentum, and energy with the assumption that the fluid, at the scale of interest, is a continuum (continuum hypothesis), i.e. the macroscopic scale of the fluid motion is large compared to the distance between the molecules of the fluid. It is important to notice that, once the continuum hypothesis is invoked, continuous fields are obtained, and all the notions of the discrete molecular nature of the fluid can be neglected. Consequently, the molecular scales cease to be relevant and the fluid properties can be described by continuous functions of the position vector\(^1\) \(\vec{r} \in \mathbb{R}^{\text{dim}}\), which belongs to a \(\text{dim}\) dimensions physical space, and the time \(t \in \mathbb{R}\) [120, 132]. Therefore, introducing the following notation: \(\rho(\vec{r}, t) : \mathbb{R}^{\text{dim}} \times \mathbb{R} \rightarrow \mathbb{R}\) for the mass density, \(\vec{u}(\vec{r}, t) : \mathbb{R}^{\text{dim}} \times \mathbb{R} \rightarrow \mathbb{R}^{\text{dim}}\) for the velocity vector and \(e^t(\vec{r}, t) : \mathbb{R}^{\text{dim}} \times \mathbb{R} \rightarrow \mathbb{R}\) for the specific total energy, the vector of the conserved variables can be defined as

\[
\mathbf{w}(\vec{r}, t) = \begin{pmatrix} \rho \\ \rho \vec{u} \\ \rho e^t \end{pmatrix} = \begin{pmatrix} \rho \\ \vec{m} \\ \rho e^t \end{pmatrix},
\]

where \(\mathbf{w}(\vec{r}, t) : \mathbb{R}^{\text{dim}} \times \mathbb{R} \rightarrow \mathbb{R}^{\text{vars}}\), with the superscript \(\text{vars}\) denoting the number of conservative variables, i.e. the number of scalar equations obtained from the the basic principles of conservation. Now, consider a fluid in a domain \(\Omega\) with boundary surface denoted by \(\partial \Omega\), without radiation and external volume forces. In this situation, the system of the compressible N-S equations are mathematically described by the following convection-

\(^1\)Vectors are assumed to be column vectors.
3.1. COMPRESSIBLE NAVIER-STOKES EQUATIONS

diffusion equations, written in conservative form

\[
\frac{\partial \rho}{\partial t} + \nabla \cdot \vec{f}_\rho (\mathbf{w}) = 0, \quad \text{with} \quad \vec{f}_\rho = \vec{m}, \quad (3.2a)
\]

\[
\frac{\partial \vec{m}}{\partial t} + \nabla \cdot \vec{f}_\vec{m} (\mathbf{w}) = 0, \quad \text{with} \quad \vec{f}_\vec{m} = \frac{\vec{m} \otimes \vec{m} - \vec{G}(\mathbf{w})}{\rho}, \quad (3.2b)
\]

\[
\frac{\partial \rho e_t}{\partial t} + \nabla \cdot \vec{f}_{\rho e_t} (\mathbf{w}) = 0, \quad \text{with} \quad \vec{f}_{\rho e_t} = \frac{\vec{m}}{\rho} \left[ \rho e_t - \vec{G}(\mathbf{w}) \right] + \vec{q}(\mathbf{w}), \quad (3.2c)
\]

where \( \vec{f}_\rho, \vec{f}_\vec{m} \) and \( \vec{f}_{\rho e_t} \) are the mass flux vector, the momentum flux vector\(^2\) and the total energy flux vector, respectively. Moreover, in this expression, \( \nabla, \vec{G} (\mathbf{w}) : \mathbb{R}^{\text{vars}} \rightarrow \mathbb{R}^{\text{dim}} \times \mathbb{R}^{\text{dim}} \) and \( \vec{q}(\mathbf{w}) : \mathbb{R}^{\text{vars}} \rightarrow \mathbb{R}^{\text{dim}} \) denote the nabla/del operator, the stress tensor and the heat flux vector, which are functions of the conserved variables. System (3.2) can also be rewritten in a compact form as

\[
\frac{\partial \mathbf{w}}{\partial t} + \nabla \cdot \vec{f}(\mathbf{w}) = 0, \quad (3.3)
\]

where symbol \( \vec{f}(\mathbf{w}) : \mathbb{R}^{\text{vars}} \rightarrow \mathbb{R}^{\text{vars}} \times \mathbb{R}^{\text{dim}} \) denotes the vector flux function

\[
\vec{f} = \left( \vec{f}_\rho (\mathbf{w}), \vec{f}_\vec{m} (\mathbf{w}), \vec{f}_{\rho e_t} (\mathbf{w}) \right)^T, \quad (3.4)
\]

which is only function of the conserved variables. This implies that, both stress tensor and heat flux vector are functions of the conserved variables. Therefore, Equation (3.3) represents the conservative form of the compressible Navier-Stokes equations. The explicit dependence of \( \vec{G} \) and \( \vec{q} \) on \( \mathbf{w} \) will be shown in the next sections.

Notice that, since the system (3.2) describes an initial-boundary value problems it must be equipped with appropriate initial and boundary conditions, i.e.

\[
\mathbf{w}(\vec{r}, 0) = \mathbf{w}^0(\vec{r}), \quad \forall \vec{r} \in \Omega, \quad (3.5a)
\]

\[
\mathbf{w}|_{\partial \Omega}(t) = \mathbf{w}_b(t), \quad \forall t \in [t^0, t^{\text{end}}], \quad (3.5b)
\]

where \( t^0 \) and \( t^{\text{end}} \) denote the lower and the upper limits of the time variable \( t \).

\(^2\)The symbol \( \otimes \) denotes the tensor product operator.
CHAPTER 3. GOVERNING EQUATIONS

3.1.1 Newtonian fluid

Generally speaking, a symmetric tensor can be decomposed in an isotropic part and in a symmetric one. The latter represents the deviation from the isotropic part which is called the deviatoric stress tensor. Therefore, applying this decomposition to the stress tensor \( \vec{G} \), introduced in Equations (3.2b) and (3.2c), one obtains

\[
\vec{G} = -\zeta \vec{I} + \vec{\sigma},
\]

(3.6)

where \( \vec{I}, \zeta \) and \( \vec{\sigma} \) represent the isotropic stress tensor, the scalar number which characterizes \( \vec{I} \) and the deviatoric stress tensor, respectively. This decomposition is not unique because \( \zeta \) is a free parameter. However, if \( \zeta \) is set equal to the thermodynamic pressure \( P^3 \), then tensor \( \vec{I} \) becomes the identity tensor and the deviatoric stress tensor \( \vec{\sigma} \) becomes the viscous stress tensor \([36]\). The latter is related to the symmetric velocity gradient tensor \( \vec{S} \) defined by

\[
\vec{S}(w) = \frac{1}{2} \left\{ [\vec{\nabla} \otimes \left( \frac{\vec{m}}{\rho} \right)] + [\vec{\nabla} \otimes \left( \frac{\vec{m}}{\rho} \right)]^T \right\}.
\]

(3.7)

Now, in the present thesis the working fluid is air. Air belongs to the fundamental class of Newtonian fluid, for which a linear relation between \( \vec{\sigma} \) and \( \vec{S} \) exists. Therefore, the viscous stress tensor \( \vec{\sigma} \) can be expressed as

\[
\vec{\sigma}(w) = 2 \mu \vec{S}(w) + \lambda_d \left[ \vec{\nabla} \cdot \left( \frac{\vec{m}}{\rho} \right) \right] \vec{I},
\]

(3.8)

where \( \mu \) and \( \lambda_d \) are the dynamic viscosity and the dilatation viscosity coefficients. The latter is related to a viscous stress caused by a volume change. Both coefficients are in general functions of the temperature \( T \) and the pressure \( P \) of the fluid, i.e. \( \mu = \mu(T, P) \) and \( \lambda_d = \lambda_d(T, P) \). Moreover, they must satisfy the following conditions: \( \mu > 0, \forall (T, P) \) and \( \lambda_d + \frac{2}{3}\mu \geq 0, \forall (T, P) \) \([134]\).

Now, assuming the Stokes’ hypothesis to be valid\(^4\) \([120, 134]\), i.e. \( \lambda_d = -\frac{2}{3}\mu \),

\(^3\)The mathematical/thermodynamic definition of both temperature and pressure will be introduced in Section 3.1.3.

\(^4\)The Stokes’s hypothesis is valid for mono-atomic gases and for more complex gases (such as air) in a wide range of thermodynamic conditions.
Equation (3.8) becomes
\[
\vec{\sigma} (\vec{w}) = 2 \mu \left\{ \vec{S} (\vec{w}) - \frac{1}{3} \left[ \nabla \cdot \left( \frac{\vec{m}}{\rho} \right) \right] \vec{I} \right\}.
\] (3.9)

Therefore, inserting (3.9) in (3.6), with \( \zeta = P \), one obtains
\[
\vec{G} = -P \vec{I} + 2 \mu \left\{ \vec{S} (\vec{w}) - \frac{1}{3} \left[ \nabla \cdot \left( \frac{\vec{m}}{\rho} \right) \right] \vec{I} \right\}.
\] (3.10)

Notice that, the pressure, which appears explicitly in Equation (3.10), and the temperature, whose contribution is hidden inside the coefficient \( \mu \), must be expressed as functions of the conserved variables, in order to assert that the stress tensor \( \vec{G} \) is only function of the conserved variables, i.e. \( \vec{G} = \vec{G} (\vec{w}) \).

### 3.1.2 Thermal conductivity

Assume the Fourier’s law of conduction to be valid [108]. Therefore, the relation between the heat flux vector \( \vec{q} \) and the gradient of the temperature \( \nabla T \) is linear and can be expressed as
\[
\vec{q} = -\kappa \nabla T,
\] (3.11)

where \( \kappa \) is the fluid’s conductivity coefficient which is also function of the thermodynamic conditions of the fluid, i.e. \( \kappa = \kappa (T, P) \). The coefficient \( \kappa \) must satisfy the following condition: \( \kappa > 0 \), \( \forall (T, P) \). Notice once more that, in order to assert that \( \vec{q} = \vec{q} (\vec{w}) \), both temperature and pressure must be expressed as functions of the conserved variable \( \vec{w} \).

### 3.1.3 Thermodynamic properties: ideal gas model

The compressible N-S equations defined by system (3.2) consist of three equations (two scalar equations and one vectorial equation) with seven unknowns: \( \rho (\vec{r}, t), \vec{u} (\vec{r}, t), e^t (\vec{r}, t), P (\vec{r}, t) \) and \( T (\vec{r}, t) \). Therefore, additional equations (together with suitable initial and boundary conditions) must be prescribed to complete (3.2). The missing equations are derived from the fundamental equation of the thermodynamics which, for single-component fluid at the chemical equilibrium, is the entropy function \( S \). The entropy is a function of the extensive parameters of the thermodynamic system such as the internal energy \( E \), the volume \( V \) and number of moles \( N_{mol}^5 \), i.e.

---

\(^5\)In the present thesis only single-component fluid is considered.
CHAPTER 3. GOVERNING EQUATIONS

\[ S = S(E, V, N_{mol}) \] [29]. The entropy function must satisfy three essential properties [29]:

- Homogeneous first-order function of the extensive parameters.
- Continuous, differentiable and monotonically increasing function of the internal energy. This implies that \( S \) can be inverted with respect to the internal energy and that the energy is a single-valued, continuous, and differentiable function of \( S, V \) and \( N_{mol} \), i.e. \( E = (S, V, N_{mol}) \).
- Super-additive function in the sense specified in [51].

The various partial derivatives of the fundamental equation in the form \( E = (S, V, N_{mol}) \) are the intensive parameters and they are defined as

\[
\begin{align*}
\frac{\partial E}{\partial S} \bigg|_{V,N_{mol}} &\equiv T(S, V, N_{mol}), \quad \text{the temperature,} \\
\frac{\partial E}{\partial V} \bigg|_{S,N_{mol}} &\equiv P(S, V, N_{mol}), \quad \text{the pressure,} \\
\frac{\partial E}{\partial N} \bigg|_{S,V} &\equiv \iota(S, V, N_{mol}), \quad \text{the electro-chemical potential.}
\end{align*}
\]

Equations (3.12) are called equations of state. They are not independent because of the Schwarz theorem which states that the partial derivatives commute [2]. Therefore, the knowledge of two equations of state is the necessary and sufficient condition to define the thermodynamic state of the fluid. In fact, the Gibbs-Duhem relations [29] combined with two equations of state allows the reconstruction of the fundamental equation of the thermodynamics.

Now, consider the ideal gas model, which approximates well with the thermodynamic behavior of air in a wide range of thermodynamic conditions. The fundamental equation of thermodynamic in specific variables, written both in entropic and energetic representations, is then

\[
\begin{align*}
s = s(e, v) &= s_0 + R \ln \left[ \left( \frac{e}{e_0} \right)^{\frac{1}{\gamma-1}} \frac{v}{v_0} \right], \quad (3.13a) \\
\end{align*}
\]

\[
\begin{align*}
e = e(s, v) &= e_0 \frac{e^{(\gamma-1)(s-s_0)/R}}{(v/v_0)^{\gamma-1}}, \quad (3.13b)
\end{align*}
\]

where \( e_0, v_0 \) and \( s_0 \) are the extensive specific value at an initial reference thermodynamic state, with \( s_0 \) defined by

\[
s_0 = \frac{\gamma R}{\gamma - 1} - \frac{\iota}{T_0}. \quad (3.14)
\]
3.1. COMPRESSIBLE NAVIER-STOKES EQUATIONS

The parameter $R$ is the specific gas constant, which is about $287.06 \, J \, kg^{-1} \, K^{-1}$ for air, and $\gamma$ is a second characteristic gas constant which will be defined later on. Therefore, temperature $T$ and pressure $P$ obey to the following equations:

$$T = \frac{\partial e}{\partial s} = \frac{\gamma - 1}{R} e,$$

(3.15a)

$$P = \frac{\partial e}{\partial v} = (\gamma - 1) \frac{e}{v} = (\gamma - 1) \rho e.$$

(3.15b)

Equation (3.15b) shows that $e = e(T) = (\gamma - 1) / RT$. Consequently, recalling that the total energy per unit volume is defined as

$$\rho e^t = \rho \left( e + \frac{1}{2} \frac{|\vec{m}|^2}{\rho} \right),$$

(3.16)

the specific internal energy can be expressed as a function of the conserved variables, i.e.

$$e = \frac{\rho e^t}{\rho} - \frac{1}{2} \frac{|\vec{m}|^2}{\rho} = e(\mathbf{w}).$$

(3.17)

This equation allows to express both pressure and temperature as a function of the conservative variables:

$$T(\mathbf{w}) = \frac{\gamma - 1}{R} \left( \frac{\rho e^t}{\rho} - \frac{1}{2} \frac{|\vec{m}|^2}{\rho} \right),$$

(3.18a)

$$P(\mathbf{w}) = (\gamma - 1) \left( \frac{\rho e^t}{\rho} - \frac{1}{2} \frac{|\vec{m}|^2}{\rho} \right).$$

(3.18b)

Therefore, system (3.2) together with Equations (3.18a) and (3.18b) results in a closed system of five non-linear partial differential equations (PDEs), which defines the complete system of compressible N-S equations for a Newtonian fluid modelled by the thermodynamic ideal gas model.

To conclude this section, the definition of specific heat capacities and speed of sound are introduced, for the ideal gas model. Using the law of conservation of energy [134], the specific heat capacities at constant volume and constant pressure result in

$$c_v = \left. \frac{\partial e}{\partial T} \right|_{v=\text{const}} = \frac{R}{\gamma - 1},$$

(3.19a)

$$c_P = \left. \frac{\partial e}{\partial T} \right|_{P=\text{const}} + P \left. \frac{\partial v}{\partial T} \right|_{P=\text{const}} = \frac{R}{\gamma - 1} + R.$$  

(3.19b)
CHAPTER 3. GOVERNING EQUATIONS

Equation (3.19a) shows that the heat capacity at constant volume $c_v$ is constant. A gas with constant $c_v$ is named *polytropic gas*. Combining Equation (3.19a) with Equation (3.19b), the Meyer’s relation is obtained:

$$c_P = c_v + R. \quad (3.20)$$

This expression demonstrates that, also the heat capacity at constant pressure $c_p$ is constant. Now, inverting Equation (3.19a) with respect to the parameters $\gamma$ and using Meyer’s relation, one obtains

$$\gamma = \frac{c_p}{c_v}, \quad (3.21)$$

which shows that the second characteristic gas constant introduced in (3.14) represents the ratio of the specific heat capacities.

Another thermodynamic quantity, which will be useful to define a dimensionless parameter, is the speed of sound $c$. For a polytropic ideal gas, $c$ is defined by [29, 134, 162]

$$c = -v^2 \frac{\partial P(s,v)}{\partial v} = \sqrt{\frac{\gamma P}{\rho}} = \sqrt{\gamma RT}, \quad (3.22)$$

which shows that, for the polytropic ideal gas model, the speed of sound depends only on the temperature $T$.

### 3.1.4 Formulation in Cartesian space

In order to show the contributions of the convective and the diffusive fluxes, separately, Equations (3.2) can be rewritten as

$$\frac{\partial \mathbf{w}}{\partial t} + \nabla \cdot \bar{f}_C(\mathbf{w}) = \nabla \cdot \bar{f}_D(\mathbf{w}, \nabla \mathbf{w}), \quad (3.23)$$

where $\bar{f}_C(\mathbf{w}(\vec{r}, t)) : \mathbb{R}^{vars} \rightarrow \mathbb{R}^{vars \times dim}$ and $\bar{f}_D(\mathbf{w}(\vec{r}, t), \nabla \mathbf{w}(\vec{r}, t)) : \mathbb{R}^{vars} \rightarrow \mathbb{R}^{vars \times dim}$ are the convective and the diffusive flux vectors defined as

$$\bar{f}_C = \left( \vec{m}, \frac{\vec{m} \otimes \vec{m}}{\rho} + P \vec{I}, \frac{\vec{m}}{\rho} \left[ \rho e^t + P \vec{I} \right] \right)^T, \quad (3.24)$$

$$\bar{f}_D = \left( \mathbf{0}, \bar{\sigma}(\mathbf{w}), \frac{\vec{m}}{\rho} \bar{\sigma}(\mathbf{w}) - \bar{q}(\mathbf{w}) \right)^T. \quad (3.25)$$
3.1. COMPRESSIBLE NAVIER-STOKES EQUATIONS

In (3.23), the divergence of the convective flux $\vec{\nabla} \cdot \vec{f}_C (w)$ and the divergence of the diffusive flux $\vec{\nabla} \cdot \vec{f}_D (w, \vec{\nabla} w)$ describe the transport (convection process) of the conserved quantities and the mechanisms that dissipate the conserved quantities (diffusion process), respectively.

Now, consider a general tridimensional ($3D \rightarrow \text{dim} = 3$) Cartesian physical space with position vector $\vec{\chi}$ defined by $\vec{\chi} = [\chi_1, \chi_2, \chi_3]^T$. The convective and the diffusive flux vectors are then defined as $\vec{f}_C (w) = [f_C, g_C, h_C]^T$ and $\vec{f}_D (w, \vec{\nabla} w) = [f_D, g_D, h_D]^T$, with

$$f_C = \begin{pmatrix} \rho u_1 \\ \rho u_1^2 + P \\ \rho u_1 u_2 \\ \rho u_1 u_3 \\ u_1 (\rho e^t + P) \end{pmatrix}, \quad (3.26a)$$

$$g_C = \begin{pmatrix} \rho u_2 \\ \rho u_1 u_2 \\ \rho u_2^2 + P \\ \rho u_2 u_3 \\ u_2 (\rho e^t + P) \end{pmatrix}, \quad (3.26b)$$

$$h_C = \begin{pmatrix} \rho u_3 \\ \rho u_1 u_3 \\ \rho u_2 u_3 \\ \rho u_3^2 + P \\ u_3 (\rho e^t + P) \end{pmatrix}, \quad (3.26c)$$

---

6In Section 3.1, where the Navier-Stokes equations were introduced using a general vectorial notation, the position vector was denoted by symbol $\vec{r}$. 

---

21
In Equations (3.27), $\sigma_{ij}$ represents the $ij$-component of the viscous stress tensor $\sigma$ which, assuming both Newtonian fluid and the Stokes' hypothesis to be valid (see Equation (3.9)), can be rewritten in tensorial notation as

$$
\sigma_{ij} = 2\mu \left( S_{ij} - \frac{\delta_{ij}}{3} S_{kk} \right) \quad i, j = 1, \ldots, \text{dim.} \tag{3.28}
$$

In this expression $\delta_{ij}$ is the Kronecker delta function\textsuperscript{7} and $S_{ij}$ represents the $ij$-component of the symmetric velocity gradient tensor $\mathbf{S}$, which, according to Equation (3.7), can also be rewritten in tensorial notation as

$$
S_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial \chi_j} + \frac{\partial u_j}{\partial \chi_i} \right) \quad i, j = 1, \ldots, \text{dim.} \tag{3.29}
$$

Moreover, in Equations (3.27), $q_i$ represents the $i-$ component of the heat flux vector, which, assuming the Fourier's law of conduction to be valid, can be expressed as

$$
q_i = -\kappa \frac{\partial T}{\partial \chi_i} \quad i = 1, \ldots, \text{dim.} \tag{3.30}
$$

\textsuperscript{7}In Section 3.1.1, the parameter $\zeta$ was set equal to the thermodynamic pressure $P$ and consequently, tensor $I$ became the identity tensor.
3.1. COMPRESSIBLE NAVIER-STOKES EQUATIONS

3.1.5 Dimensionless numbers

Consider a flow modelled by the compressible N-S equations, valid in a domain \( \Omega \) with boundary \( \partial \Omega \). Dimensional analysis shows that this flow is completely characterized by three dimensionless numbers: Mach number, Reynolds number and Prandtl number. The Mach Number is a value useful for analyzing fluid flow dynamics problems where compressibility is a significant factor. It is defined by

\[
M = \frac{|\vec{u}_c|}{c},
\]

where the quantity at the denominator \( c \) is the speed of sound, defined by Equation (3.22) for the ideal gas model, and \( \vec{u}_c \) is a characteristic flow velocity. Notice that, the subscript \( c \) denotes a characteristic value and should not be confused with the speed of sound. If the Mach number is less than one, then the local flow is subsonic, and if it is greater than one, then the local flow is supersonic. Typically, a flow can be treated as incompressible if \( M \ll 0.3 \). For higher Mach numbers, compressibility must be taken into account.

The Reynolds number is defined by

\[
Re = \frac{\rho_c |\vec{u}_c| L_c}{\mu_c} = \frac{|\vec{u}_c| L_c}{\nu_c},
\]

where \( \nu \) is the kinematic viscosity coefficient given by

\[
\nu = \frac{\mu}{\rho}.
\]

The quantity \( L_c \) represents a characteristic length scale of the flow problem. The Reynolds number can be interpreted as a ratio of typical inertial stresses and typical viscous stresses. In fact, its definition can be rewritten as

\[
Re = \frac{\rho_c |\vec{u}_c|^2}{\mu_c |\vec{u}_c| L_c} \approx \frac{\text{inertial forces}}{\text{viscous forces}}
\]

If the Reynolds number is low, then the flow is dominated by the viscous stresses, which results in a laminar flow. If the Reynolds number is high, then the inertial stresses dominate and the flow is turbulent, characterized by random vortices and stochastic processes [132].

The Prandtl number \( Pr \) is defined by

\[
Pr = \frac{\mu_c c_P c}{\kappa_c}.
\]
CHAPTER 3. GOVERNING EQUATIONS

It is a measure of the ratio between momentum diffusivity and thermal diffusivity, or the ratio of the rate by which momentum is transferred by viscosity and the rate by which heat is transferred by conduction. The latter interpretation can be understood from the following relation:

\[
Pr = \frac{\nu_c}{\kappa_c} \approx \frac{\text{momentum diffusivity}}{\text{thermal diffusivity}}.
\] (3.36)

According to Equation (3.35), the Prandtl number depends only on the fluid’s properties, which for air, in a wide range of thermodynamic conditions around the standard values, can be assumed constant and equal to 0.72 [162].

3.2 Large eddy simulation

The four main numerical procedures for solving the N-S equations are Direct Numerical Simulation (DNS), Large Eddy Simulation (LES), Detached Eddy Simulation (DES) and Reynolds Averaged N-S (RANS) approach. An elaborate overview of these techniques can be found in several books [105, 132, 193]. The most accurate approach is DNS. DNS consists in solving the N-S equations (system (3.2) together with Equations (3.18a) and (3.18b)), resolving all the scales of motion, with initial and boundary conditions appropriate to the flow considered. Conceptually it is the simplest approach and, when it can be applied, it is unrivalled in accuracy and in the level of description provided. However, it is important to appreciate that the cost is extremely high. In fact, since all the spatial scales, from the smallest dissipative Kolmogorov scale \( l_\eta \) up to the energy containing integral scale \( l \), are needed to be resolved by the computational mesh, the number of modes (points) required for a resolved DNS in 3D can be estimated as [132],

\[
N_{\text{mod},3D} = \left( \frac{l}{l_\eta} \right)^3 \sim \left( \frac{u_l l_c}{\nu_c} \right)^3 = Re^{9/4}.
\] (3.37)

The number of modes, for fully resolved DNS is enormous large, especially for high Reynolds number flows encountered in most industrial applications, and consequently DNS is restricted to relatively low Reynolds number flows\(^8\). To overcome this limitation and solve high-Reynolds number flows, other numerical techniques like for instance LES, DES and RANS

---

\(^8\)DNS is generally used as a research tool for analyzing the mechanics of turbulence, such as turbulence production, energy cascade, energy dissipation, drag reduction etc.
3.2. LARGE EDDY SIMULATION

have been developed. In this thesis, the LES approach is used as a cheaper alternative to DNS.

In LES, the larger 3D unsteady turbulent motions are directly represented, whereas the effect of the smaller-scale motions are modelled. In computational expense, LES lies between RANS and DNS, and it is motivated by the limitations of each of these approaches [132]. Because the large-scale unsteady motions are represented explicitly, LES can be expected to be more accurate and reliable than RANS for flows in which large-scale unsteadiness is significant - such as the flow over bluff bodies, which involves unsteady separation and vortex shedding. As already discussed, the computational cost of DNS is high, and it increases as the cube of the Reynolds number. Nearly all of the computational effort in DNS is expended on the smallest, dissipative motions, whereas the energy and anisotropy are contained predominantly in the larger scales of motion. In LES, the dynamic of the larger-scale motions (which are affected by the flow geometry and are not universal) are computed explicitly, while the influence of the smaller scales (which have a universal character) are being represented by simple models. Thus, compared with DNS, the large computational cost of explicitly representing the small-scale motions is avoided.

There are four conceptual steps in LES.

- Decompose the velocity \( \tilde{u}(\chi, t) \) into the sum of a filtered (or resolved) component \( \tilde{u}(\chi, t) \) and a residual (or subgrid-scale, ”sgs”) component \( \tilde{u}^{sgs}(\chi, t) \equiv \tilde{u}'(\chi, t) \). The filtered velocity field \( \tilde{u} \) represents the motion of the large eddies.

- The equations for the evolution of the filtered fields are derived from the compressible N-S equations (system (3.2) together with Equations (3.18a) and (3.18b)). Thus, the momentum equation contains the residual stress tensor (or ”sgs” stress tensor) \( \tilde{\tau}^{sgs} \) and the energy equation contains both \( \tilde{\tau}^{sgs} \) and the residual heat flux vector (or ”sgs” heat flux vector) \( \tilde{q}^{sgs} \).

- Closure is obtained by modelling \( \tilde{\tau}^{sgs} \) and \( \tilde{q}^{sgs} \), most simply by eddy-viscosity [132] and eddy-diffusivity [50] models, respectively.

- The resulting filtered equations, i.e. the filtered compressible Navier-Stokes equations, are solved numerically, providing one approximated realization of the turbulent flow.
CHAPTER 3. GOVERNING EQUATIONS

3.2.1 Formulation in Cartesian space

Proceeding with the first two steps described in Section 3.2, one can define the vector of the conserved variables for LES as

\[ \mathbf{w}^{LES} = \begin{pmatrix} \frac{\rho}{\rho u} \\ \frac{\rho u}{\rho e_t} \\ \frac{\rho e_t}{\rho e_t} \end{pmatrix} = \begin{pmatrix} \frac{\rho}{\rho \tilde{u}} \\ \frac{\rho}{\rho \tilde{u}} \\ \frac{\rho}{\rho \tilde{u}} \end{pmatrix}, \tag{3.38} \]

where the symbols (·) and (·) represent the spatially filtered and the Favre filtered fields, respectively. The Favre filtered is defined as \( \tilde{g} = \frac{\rho g}{\rho} \), where \( g \) is any field [97]. Consequently, the components of the convective and the diffusive flux vector of the filtered compressible N-S equations are

\[ f_C^{LES} = \begin{pmatrix} \frac{\rho \tilde{u}_1}{\rho \tilde{u}_2} \\ \frac{\rho \tilde{u}_1 \tilde{u}_2}{\rho \tilde{u}_2} \\ \frac{\rho \tilde{u}_1 \tilde{u}_3}{\rho \tilde{u}_3} \\ \frac{\rho \tilde{u}_2}{\rho \tilde{u}_3} \end{pmatrix}, \tag{3.39a} \]

\[ g_C^{LES} = \begin{pmatrix} \frac{\rho \tilde{u}_2}{\rho \tilde{u}_2} \\ \frac{\rho \tilde{u}_1 \tilde{u}_2}{\rho \tilde{u}_2} \\ \frac{\rho \tilde{u}_2 \tilde{u}_3}{\rho \tilde{u}_3} \\ \frac{\rho \tilde{u}_3}{\rho \tilde{u}_3} \end{pmatrix}, \tag{3.39b} \]

\[ h_C^{LES} = \begin{pmatrix} \frac{\rho \tilde{u}_3}{\rho \tilde{u}_3} \\ \frac{\rho \tilde{u}_1 \tilde{u}_3}{\rho \tilde{u}_3} \\ \frac{\rho \tilde{u}_2 \tilde{u}_3}{\rho \tilde{u}_3} \\ \frac{\rho \tilde{u}_3}{\rho \tilde{u}_3} \end{pmatrix}. \tag{3.39c} \]
and

\[ f_{LES}^{D} = \begin{pmatrix} 0 \\ \tilde{u}_1 (\tilde{\sigma}_{11} - \tau_{11}^{sgs}) + \tilde{u}_2 (\tilde{\sigma}_{21} - \tau_{21}^{sgs}) + \tilde{u}_3 (\tilde{\sigma}_{31} - \tau_{31}^{sgs}) - \tilde{q}_1 - q_1^{sgs} \end{pmatrix} \]

(3.40a)

\[ g_{LES}^{D} = \begin{pmatrix} 0 \\ \tilde{u}_1 (\tilde{\sigma}_{12} - \tau_{12}^{sgs}) + \tilde{u}_2 (\tilde{\sigma}_{22} - \tau_{22}^{sgs}) + \tilde{u}_3 (\tilde{\sigma}_{32} - \tau_{32}^{sgs}) - \tilde{q}_2 - q_2^{sgs} \end{pmatrix} \]

(3.40b)

\[ h_{LES}^{D} = \begin{pmatrix} 0 \\ \tilde{u}_1 (\tilde{\sigma}_{13} - \tau_{13}^{sgs}) + \tilde{u}_2 (\tilde{\sigma}_{23} - \tau_{23}^{sgs}) + \tilde{u}_3 (\tilde{\sigma}_{33} - \tau_{33}^{sgs}) - \tilde{q}_3 - q_3^{sgs} \end{pmatrix} \]

(3.40c)

where, according to Equations (3.28), (3.29) and (3.30),

\[ \tilde{\sigma}_{ij} = 2\mu \left( \tilde{S}_{ij} - \delta_{ij} \tilde{S}_{kk} \right) \quad i, j = 1, \ldots, \text{dim}, \]

(3.41)

\[ \tilde{S}_{ij} = \frac{1}{2} \left( \frac{\partial \tilde{u}_i}{\partial \chi_j} + \frac{\partial \tilde{u}_j}{\partial \chi_i} \right) \quad i, j = 1, \ldots, \text{dim}, \]

(3.42)

\[ \tilde{q}_i = -\kappa \frac{\partial \tilde{T}}{\partial \chi_i} \quad i = 1, \ldots, \text{dim}. \]

(3.43)

Notice that, according to the LES approach, the total energy \( \tilde{e}^t \) should be calculated as follows

\[ \tilde{e}^t = \tilde{e} + \frac{1}{2} \left| \frac{m}{\rho} \right|^2 + k^{sgs}, \]

(3.44)

where \( k^{sgs} \) is the subgrid-scale kinetic energy. For air, \( k^{sgs} \) is very small and can be neglected as shown by Lesieur and Comte [97].
From the definitions of the fluxes components it is seen that both momentum and energy equations differ from the classical fluid dynamic equations only for two terms which take into account the contributions from the unresolved scales. These contributions, represented by the specific subgrid-scale stress tensor $\tau_{ij}^{sgs}$ and by the subgrid heat-flux vector defined $q_i^{sgs}$, appear when the spatial filter is applied to the convective terms and they are defined as follows [56, 97]

$$
\tau_{ij}^{sgs} = \tilde{\rho} (\tilde{u}_i \tilde{u}_j - \tilde{u}_i \tilde{u}_j) \quad i, j = 1, \ldots, \text{dim}, \tag{3.45}
$$

$$
q_i^{sgs} = c_P \tilde{\rho} (\tilde{T} \tilde{u}_i - \tilde{T} \tilde{u}_i) \quad i = 1, \ldots, \text{dim}. \tag{3.46}
$$

The interactions of $\tau_{ij}^{sgs}$ and $q_i^{sgs}$ with the resolved scales have to be modeled through a subgrid-scale model because they cannot be determined using only the resolved flow field $\mathbf{w}^{LES}$.

Notice that, according to Vreman et al. [182], terms $\tilde{u}_i \tau_{ij}^{sgs}$ in Equations (3.40) improve the results only at moderate or high Mach number. Consequently, if the Mach number is low $\tilde{u}_i \tau_{ij}^{sgs}$ can be neglected.

### 3.2.2 The wall-adapted local eddy-viscosity model

In the previous section, it has been shown that the smaller scales and their interaction with the resolved scales have to be modeled through the subgrid-scale stress tensor $\tilde{\tau}_{ij}^{sgs}$. Tensor $\tilde{\tau}_{ij}^{sgs}$ can be modeled at different levels of complexity. The most common approach is based on the eddy-viscosity concept in which one assumes that the residual stress is proportional to the filtered rate of strain [132]:

$$
\tau_{ij}^{sgs} - \tau_{kk}^{sgs} \delta_{ij} = -2 \tilde{\rho} \nu_t \left( \tilde{S}_{ij} - \frac{\delta_{ij}}{3} \tilde{S}_{kk} \right) = -2 \tilde{\rho} \nu_t \tilde{S}_{ij}^D, \tag{3.47}
$$

where $\nu_t$ is the turbulent kinematic viscosity or eddy-viscosity coefficient. In the wall-adapted local eddy-viscosity (WALE) proposed by Nicoud and Ducros [116], it is assumed that the eddy-viscosity $\nu_t$ is proportional to the square of the length scale of the filter and the filtered local rate of strain. Although the model was originally developed for incompressible flows, it can also be used for variable mass density flows by giving the formulation as follows [56, 126]

$$
\nu_t = (C \Delta)^2 \left| \frac{\tilde{S}_i}{\tilde{S}_j} \right|. \tag{3.48}
$$
3.2. LARGE EDDY SIMULATION

Here $|\tilde{S}|$ is defined as

$$
|\tilde{S}| = \left[ \tilde{S}_{ij} \tilde{S}_{ij} \right]^{5/4},
$$

(3.49)

where $\tilde{S}_{ij}^d$ is given by

$$
\tilde{S}_{ij}^d = \frac{1}{2} (\tilde{g}_{ij}^2 + \tilde{g}_{ji}^2) - \frac{\delta_{ij}}{3} \tilde{g}_{kk}^2,
$$

(3.50)

with

$$
\tilde{g}_{ij}^2 = \frac{\partial \tilde{u}_i}{\partial x_k} \frac{\partial \tilde{u}_k}{\partial x_j},
$$

(3.51)

Note that in Equation (3.48) $\Delta$, i.e. the grid filter width, is an unknown function and is often taken to be proportional to the smallest resolvable length scale of the spatial discretization. In this thesis, the definition of the grid filter function will be discussed in Section 4.2.6, where the high-order spectral difference method is presented.

The WALE model is specifically designed to return the correct wall-asymptotic $y^+3$ variation of the subgrid-scale viscosity $\nu_t$ [116] and the constant model coefficient $C$ can be adjusted so that the correct amount of subgrid dissipation is obtained. This model has the following properties:

- It is invariant to any coordinate translation or rotation.
- It is easily computed on any kind of computational grid.
- It is a function of both the strain and the rotation rates.
- It naturally goes to zero at the wall: neither damping function nor dynamic procedure is needed to reproduce the effect of the no-slip condition.

For the subgrid heat-flux vector $q_{i}^{sgs}$, if an eddy diffusivity model [50] is used, the following expression is obtained

$$
q_{i}^{sgs} = c_P \rho \left( \tilde{T}_{ui} - \tilde{T}_{u_i} \right) = -\nu_t c_P \rho \frac{\partial T}{\partial x_i},
$$

(3.52)

where the value of the turbulent Prandtl number $Pr_t$ for air is usually set to 0.72 [198] and the eddy-viscosity is computed by Equation (3.48).
CHAPTER 3. GOVERNING EQUATIONS

3.3 Boundary conditions

Most partial differential equations encountered in physical applications are of three basic types: elliptic, parabolic and hyperbolic systems. Gustafsson and Sundström [59] classified the unsteady compressible N-S equations as being an incompletely parabolic system. They demonstrated that the number of boundary conditions shown in Table 3.3 is necessary and sufficient to define a well-posed problem. Notice that in Table 3.3, the number of boundary conditions for the compressible Euler equations is also given. These equations are used to model the flow of a compressible inviscid fluid, or the flow of a compressible viscous fluid in flow regions where the effect of viscosity and heat conduction are negligible. The compressible Euler equations may be derived from the compressible N-S equations neglecting the contribution of the diffusive fluxes, i.e. $\vec{f}_D(\vec{w}, \vec{\nabla} \vec{w}) = 0$.

In this work, the boundary conditions are specified by a weak imposition approach, i.e imposing a boundary value through the boundary face flux. This is achieved by introducing ghost solutions, which are defined such that the average of the ghost solution, indicated by the subscript $\text{gho}$, and the internal numerical solution, indicated by the subscript $\text{int}$, is equal to the value prescribed by the boundary condition.

In this work the following boundary conditions are used: far field, mass density-velocity inlet, pressure outlet and solid surface. In the next sections, these boundary conditions are introduced for the compressible N-S equations completed with the ideal gas model. However, for obvious reasons, they are also valid for the LES equations presented in Section 3.2.

### Table 3.1: Number of boundary conditions to be specified for both compressible Euler and Navier-Stokes equations.

<table>
<thead>
<tr>
<th>Comp. Euler Eqs.</th>
<th>Inlet</th>
<th>Outlet</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Subsonic</td>
<td>Supersonic</td>
</tr>
<tr>
<td>Comp. N-S Eqs.</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>

3.3.1 Far field

In real life, with external flow problems, the domain extends to infinity - or at least far enough from the body to justify this assumption. However,
3.3. BOUNDARY CONDITIONS

when the governing equations of fluids motion are solved numerically, it is not possible to represent an infinite domain and an artificial far field boundary, as far away as possible from the region of interest, must be introduced. In addition, at this boundary, a suitable far field boundary condition, which minimizes the reflections of out-going waves and is accurate, must be prescribed. Currently, there is no method available that is capable of completely avoiding such spurious reflections.

The simplest choice to impose a far field boundary condition is to assume that the flow far from the region of interest is undisturbed. That means treating the boundary with a Dirichlet boundary condition. The ghost solution is then computed as

\[ w_{gho} = 2w_\infty - w_{int}, \] (3.53)

where \( w_\infty \) is the undisturbed flow solution. The ghost values of the gradient is set to the internal value,

\[ (\vec{\nabla}w)_{gho} = (\vec{\nabla}w)_{int}, \] (3.54)

such that the average of the ghost gradient and the internal gradient is equal to the internal gradient. This approach does not take into account the flow physics, but it generally works well if the boundary is sufficiently far away from the body. It can be combined with a 'buffer layer' to avoid strong spurious wave reflections. Such a buffer layer is a zone between the domain of interest and the actual far field boundary, where outgoing and spuriously reflected waves are damped. The damping can be accomplished for instance by progressively increasing the size of the cells towards the outflow, which increases the numerical damping introduced by the solution method, or by introducing additional damping terms in the buffer layer (see for instance Zhou and Wang [200]). The latter approach belongs to a class of techniques called non-reflecting/absorbing boundary conditions.

Among the non-reflecting boundary condition approaches, the most used technique is based on the characteristics of the compressible Euler equations [57, 74, 75, 98, 131, 170]. This approach was developed as one of the first attempts to minimize the reflection of outgoing waves. It is based on a one-dimensional (1D) local approximation of the flow for the normal direction to the boundary. A number of physical variables equal to the number of outgoing Riemann invariants is extrapolated from the internal domain. The remaining physical variables at the boundary are computed using the expressions that state that the incoming Riemann boundary invariants corresponding to the 1D approximation are zero. For instance, in
a general tridimensional case, the number of conserved variables, i.e. the number of the unknown in the compressible N-S equations, is five. According to the 1D characteristic boundary condition approach, at subsonic inflow, four variables must be specified with free stream values and one is computed from the interior solution. At subsonic outflow, one variable must be specified and four others are computed from inside the computational domain. If the flow is supersonic, all variables must be specified at an inflow boundary and all must be computed from the interior solution at a supersonic outflow boundary. This approach causes less reflections than a simple Dirichlet boundary condition and it works well for steady problems. However, the boundary should still be placed as far away as possible from the region of interest. For multi-dimensional problems, the performance of the 1D characteristic boundary condition approach degrades if the wave propagation direction is not aligned with the boundary face normal direction. It can also be combined with a buffer layer to better damp outgoing and spuriously reflected waves. A family of boundary conditions which simulate outgoing radiation are derived by Bayliss and Turkel \[19\]. Those boundary conditions were applied to the computation of steady state flows and are shown to significantly accelerate the convergence to steady state.

### 3.3.2 Inlet mass density and velocity

According to the 1D characteristic boundary condition approach, for a subsonic inflow, four variables must be specified and one must be computed from the interior solution. This idea is used to impose inlet boundary conditions for flow problems where, for obvious reasons, the boundary \(\partial \Omega\) or a part of it is always associated to a subsonic inflow\(^9\). In this case, both mass density \(\rho_{\text{inlet}}\) and velocity profiles \(\vec{u}_{\text{inlet}}\) are prescribed and the ghost solution is computed as

\[
\rho_{\text{gho}} = 2 \rho_{\text{inlet}} - \rho_{\text{int}}, \quad \text{(3.55a)}
\]

\[
\vec{m}_{\text{gho}} = \rho_{\text{gho}} \vec{u}_{\text{gho}}, \quad \text{with} \quad \vec{u}_{\text{gho}} = 2 \vec{u}_{\text{inlet}} - \vec{u}_{\text{int}}, \quad \text{(3.55b)}
\]

\[
\left( \rho e^t \right)_{\text{gho}} = \frac{P_{\text{int}}}{\gamma - 1} + \frac{1}{2} \rho_{\text{gho}} \left| \frac{\vec{m}_{\text{gho}}}{\rho_{\text{gho}}} \right|^2, \quad \text{(3.55c)}
\]

where \(P_{\text{int}}\) denotes the internal numerical pressure at the boundary. In the latter expression, the internal energy per unit volume has been expressed using Equation (3.15b), which is valid for the ideal gas model.

\(^9\)For instance, the inlet section of a pipe.
3.3.3 Pressure outlet

This boundary condition is used for outlet boundary where, according to the 1D characteristic boundary condition approach, one variable must be specified. It is assumed that the pressure distribution $P_{\Omega}$ is known and equal to $P_{\text{outlet}}$. Consequently, the ghost solution is computed as

$$\rho_{\text{gho}} = \rho_{\text{int}},$$

$$\vec{m}_{\text{gho}} = \rho_{\text{gho}} \vec{u}_{\text{gho}}, \quad \text{with} \quad \vec{u}_{\text{gho}} = \vec{u}_{\text{int}},$$

$$\left(\rho e^t\right)_{\text{gho}} = \frac{P_{\text{gho}}}{\gamma - 1} + \frac{1}{2} \rho_{\text{gho}} \left| \frac{\vec{m}_{\text{gho}}}{\rho_{\text{gho}}} \right|^2, \quad \text{with} \quad P_{\text{gho}} = 2P_{\text{outlet}} - P_{\text{int}}. \quad (3.56c)$$

Notice that, also in (3.56c) the internal energy per unit volume has been expressed using Equation (3.15b). This type of outlet boundary condition works well if the outlet boundary is sufficiently far away from the region of interest. To avoid strong spurious wave reflections, it can be combined with a buffer layer.

3.3.4 Solid wall

At solid walls, the only physical boundary condition required by a viscous fluid is the no-slip boundary condition, which states that the fluid velocity relative to the wall has to vanish. This translates into the following mathematical expression:

$$\vec{u} |_{\partial \Omega_{\text{wall}}} - \vec{u}_{\text{wall}} = \vec{0}. \quad (3.57)$$

Recalling the significance of the Reynolds number as the ratio of convective to viscous effects, a high Reynolds number means that the flow system will be dominated by its inviscid properties. This can be clearly seen when analyzing viscous flows, with the important exception of the near-wall regions, where the viscous effects, leading to a boundary layer or frictional layer configuration (see Figure 3.1), dominate the flow behavior. The higher the Reynolds number, the thinner the boundary layer becomes.

The concept of boundary layer, therefore, implies that flows at high Reynolds numbers can be divided up into two unequally large regions. In the bulk of the flow region the viscosity can be neglected, and the flow corresponds to the inviscid limiting solution. This is called the inviscid outer flow. The second region is the very thin boundary layer at the wall where the viscosity plays a very important role. Within the boundary layer both laminar and turbulent flow can occur. One then speaks of laminar boundary layer flows,
and equivalently of turbulent boundary layer flows. High Reynolds number flow simulations exhibit strong gradients normal to walls and across shear layers requiring much finer resolution of the solution in some direction compared to others. To keep mesh sizes manageable for such problems, meshes with highly anisotropic elements are necessary. This introduces a geometrical stiffness which the solver must also be able to deal with.

It must be made absolutely clear that the concept of boundary layer thickness has been artificially introduced. The transition from boundary layer flow to outer flow, at least in the case of laminar flows, takes place continuously, so that a precise boundary layer thickness cannot, in principle, be given. Since this concept is so vivid, it is very often used in practice though. Frequently the boundary is arbitrarily given as being at the point where the velocity reaches a certain percentages of the outer velocity, e.g. 99% as shown in Figure 3.1.

A boundary condition for the temperature status of the solid wall should also be specified. Constant temperature wall leads to a Dirichlet type boundary condition

\[ T|_{\partial \Omega_{wall}} - T_{wall} = 0. \]  

(3.58)
The ghost solution is then computed as

\[ \rho_{gho} = \rho_{int}, \quad (3.59a) \]

\[ \vec{m}_{gho} = \rho_{gho} \vec{u}_{gho}, \quad \text{with} \quad \vec{u}_{gho} = -\vec{u}_{int}, \quad (3.59b) \]

\[ (\rho e^t)_{gho} = \rho_{gho} c_v T_{gho} + \frac{1}{2} \rho_{gho} \left| \frac{\vec{m}_{gho}}{\rho_{gho}} \right|^2, \quad \text{with} \quad T_{gho} = 2T_{wall} - T_{int}, \quad (3.59c) \]

where Equation (3.19a) has been used to express the internal energy \( e \) as a function of the temperature \( T \).

In case of imposed heat flux, the solid wall is the source of a fixed heat flux normal to the wall to or from the fluid flow, for instance when the solid surface is part of a heat exchanger system. The flux will be positive for a heated wall or negative for a cooled wall. This boundary condition, assuming the Fourier’s law of conduction to be valid, is described by the following mathematical expression:

\[ -\kappa \left( \vec{1}_n \cdot \vec{\nabla}T \right)_{\partial\Omega_{wall}} - q_{n,wall} = 0, \quad (3.60) \]

where \( \vec{1}_n \) represent the normal unit vector to the wall and \( q_{n,wall} \) is heat flux normal to the wall defined as \( q_{n,wall} = \vec{q}_{wall} \cdot \vec{1}_n \). In this case, within a thermal boundary layer the temperature rapidly changes from the value imposed by the wall to the one of the flow away from the wall.

The Prandtl number is a measure of the relative thickness of the momentum\(^{10} \) to the thermal boundary layers. In heat transfer problems, the Prandtl number controls the relative thickness of the momentum and thermal boundary layers. When \( Pr < 1 \), it means that the heat diffuses very quickly compared to the velocity (momentum).

In case of imposed heat flux, the ghost value of the temperature gradient is specified as

\[ (\vec{\nabla}T)_{gho} = (\vec{\nabla}T)_{int} - 2 \left[ \frac{q_{n,wall}}{\kappa} + \vec{1}_n \cdot (\vec{\nabla}T)_{int} \right] \vec{1}_n. \quad (3.61) \]

\(^{10}\)The boundary-layer thickness is used for a thickness beyond which the velocity is essentially the free-stream velocity (see Fig. 3.1).
CHAPTER 3. GOVERNING EQUATIONS

3.4 Aerodynamic coefficients

The exact solution of the N-S equations is generally not available. Therefore, numerical solutions are mainly validated by comparison with experimental results and/or accurate and usually very expensive simulations. Dimensionless quantities like pressure, lift and drag coefficients can also be used to assess the accuracy of a numerical method to solve fluid flow problems. These coefficients contain complex dependencies of the geometry of the flow problem. To correctly use them, the viscosity and compressibility effects of predicted case must be equal to those of the measured and/or computed case. Otherwise, the prediction will be inaccurate.

Let the quantity dynamic pressure be defined as $\frac{1}{2}\rho_{c} |\vec{u}_{c}|^2$. The pressure coefficient $C_{P}$ is then defined as the difference of the instantaneous pressure in one point and a reference pressure, divided by the dynamic pressure, i.e.

$$C_{P} = \frac{P - P_{c}}{\frac{1}{2}\rho_{c} |\vec{u}_{c}|^2}.$$  \hspace{1cm} (3.62)

The pressure coefficient is a dimensionless number which describes the relative pressures throughout a flow field in fluid dynamics. Every point in a fluid flow field has its own unique pressure coefficient.

The lift coefficient $C_{L}$ expresses the ratio between the component $F_{L}$ of the global aerodynamic force in the direction perpendicular to the reference velocity $\vec{u}_{c}$ and the force produced by the dynamic pressure times a reference area of the object $A_{c}$, i.e.

$$C_{L} = \frac{F_{L}}{\frac{1}{2}\rho_{c} |\vec{u}_{c}|^2 A_{c}}.$$  \hspace{1cm} (3.63)

The drag coefficient $C_{D}$ expresses the ratio between the component $F_{D}$ of the global aerodynamic force in the direction parallel to the reference velocity $\vec{u}_{c}$ and the force produced by the dynamic pressure times a reference area of the object $A_{c}$. Its definition is

$$C_{D} = \frac{F_{D}}{\frac{1}{2}\rho_{c} |\vec{u}_{c}|^2 A_{c}}.$$  \hspace{1cm} (3.64)
3.5 Linear convection equation

The linear convection equation (also known as linear advection equation) describes mathematically the conservation law of a scalar conserved variable $w$ convected by a constant vectorial velocity field $\vec{\alpha}$. Using the notation introduced for the N-S equations in Section 3.2.1, the linear convection equation is defined by

$$\frac{\partial w}{\partial t} + \vec{\nabla} \cdot \vec{f}_C(w) = 0,$$

(3.65)

where $\vec{f}_C(w) = \vec{\alpha} w$. Although this model is very simple, it describes important physical phenomena, like the propagation of waves or propagation of a chemical substance where the diffusion in the flow is negligible. Moreover, since this simple model has a general analytical solution [98, 100], it is often used to assess the stability properties of a spatial discretization scheme and, as it will be shown in Chapter 6, to study the stability and the damping properties of time solvers.

For an initial condition $w^0(\vec{r})$, Equation (3.65) has the general solution

$$w(\vec{r}, t) = w^0(\vec{r} - \vec{\alpha} t).$$

(3.66)
CHAPTER 3. GOVERNING EQUATIONS
Chapter 4

Spatial discretization

In the previous chapter, the Navier-Stokes and the filtered Navier-Stokes equations, which govern all the physical problem considered in this thesis, have been discussed. They form two systems of partial differential equations (PDEs) which describe initial-boundary value problems. In a very limited number of test cases, the exact solution of the Navier-Stokes equations is known. Therefore, in the majority of the flow problems, the solution must be computed numerically and the PDEs must be discretized both in space and time. In order to achieve this, the method of lines (MOL) is used in this work [75, 157]. The basic idea of the MOL is to replace the spatial derivatives in the PDEs with algebraic approximations. Once this is done, the spatial derivatives are no longer stated explicitly in terms of the spatial independent variables. Thus, only the initial value variable, typically time in a physical problem, remains. With only one remaining independent variable, we have then a system of ordinary differential equations (ODEs) that approximate the original PDEs. Therefore, the space discretization is separate from the time integration. A significant advantage of the method is that it allows the solution to take advantage of the sophisticated general purpose methods and software that have been developed for numerically integrating ODEs. For the PDEs to which the method of lines is applicable, the method typically proves to be quite efficient.

In this chapter, two compact methods that are specifically designed for high-order accuracy on unstructured grids, namely the spectral volume (SV) and the spectral difference (SD) methods, are presented for the spatial discretization of system of PDEs. The mathematical formulations of the SV and SD methods are described in the following sections.
4.1 Spectral volume method

The SV method, with applications to 1D scalar conservation laws, was proposed in 2002 by Wang [183] as an alternative for the discontinuous Galerkin (DG) method [37, 39–41, 43, 140, 180]. Further development of the SV method for 2D and for non-linear hyperbolic systems, such as the Euler equations, was then reported in subsequent papers by Wang et al. [186, 187, 191]. The extension to 3D for linear systems was described in Liu et al. [104]. The method was applied to the 2D N-S equations in Sun et al. [164] and was extended to curved boundary representation by Wang et al. [188]. The first application of the SV method to the 3D N-S equations was reported by Haga et al. [62]. A positive step towards addressing the issue of stability was given by Van den Abeele et al. [174–176], for 1D, 2D and 3D SV schemes.

4.1.1 Discretization of convective term

The spectral volume method can be interpreted as a finite volume (FV) method, where within each grid cells a miniature structured mesh of control volumes (CVs) is formed. In this way a unique stencil for the flux is defined for each face, eliminating the need for searching operations, which are needed in traditional high-order (> 2) FV methods on unstructured grids. Volume-averaged conserved variables on the CVs, which are the solution variables for the SV method, are then used to reconstruct a high-order polynomial inside the cell. To represent the solution as a polynomial of degree $p$, i.e. a $(p+1)$-th-order accurate scheme, we need $N^s$ pieces of independent information, or degrees of freedom (DOFs). Simplex cells (lines, triangles and tetrahedrons) are then subdivided into

$$N^s = \frac{(p + \text{dim})!}{p! \text{dim}!}$$  \hspace{1cm} (4.1)

CVs. Tensor product cells (quadrilaterals and hexahedrons) are subdivided into

$$N^s = (p + 1)^{\text{dim}}$$  \hspace{1cm} (4.2)

CVs. For example, 1D, 2D and 3D spectral volumes supporting a quadratic data reconstruction ($p = 1$) are shown in Figures 4.1(a), 4.1(b) and 4.1(c) respectively.

Consider the following form of an hyperbolic system of conservation laws
4.1. SPECTRAL VOLUME METHOD

(a) 1D SV.

(b) 2D SV triangular cell.

(c) 3D SV tetrahedral cell.

Figure 4.1: Partitions for second-order ($p = 1$) SV method for 1D cell, 2D triangular cell and 3D tetrahedral cell.
with only convective fluxes:
\[
\frac{\partial w}{\partial t} + \vec{v} \cdot \vec{f}_C(w) = 0, \tag{4.3}
\]
valid on a domain \( \Omega \) with boundary \( \partial \Omega \). Define the CV-averaged conserved variable for the CV \( j \) which belongs to cell \( i \) as
\[
\bar{W}_{i,j} = \frac{1}{\Delta \Omega_{i,j}} \int_{\Delta \Omega_{i,j}} w \, d\Omega, \tag{4.4}
\]
where \( \Delta \Omega_{i,j} \) is the volume of the CV. Now, integrating the system (4.3) over the CV \( j \) and applying Gauss’s theorem, one obtains
\[
\frac{d\bar{W}_{i,j}}{dt} = -\frac{1}{\Delta \Omega_{i,j}} \oint_{\partial \Omega_{i,j}} \vec{f}_C \cdot \vec{1}_n \, dS = R_{i,j}, \tag{4.5}
\]
where \( R_{i,j} \) is the residuals which governs the evolution of the solution variables in time. Given the \( N^* \) CV-averaged conserved variables for all CVs in cell with index \( i \), a polynomial of degree \( p \) can be constructed such that it is a \((p + 1)\)-th-order accurate approximation of \( w \) inside the cell \( i \).

The solution polynomial in each cell allows to evaluate the surface integrals over the internal faces that enclose the CVs with \((p + 1)\)-th-order of accuracy (i.e. the integral in Equation 4.5). Here, a quadrature-free approach, which does not need the Gaussian quadrature rules [67], is used to perform the integration. At the external faces, which lie on a grid face between two cells, two approximate values for the conserved variables are available, from the solution polynomials corresponding to the two neighboring cells. In order to ensure numerical conservation, the contributions of a face to its two neighboring cells should be equal in magnitude and opposite in sign. Thus, a unique flux \( \vec{f}_C \cdot \vec{1}_n \) should be computed from the two available solutions. In Godunov’s original method, this was achieved by computing an exact solution to the resulting Riemann problem [100]. Nowadays, approximate Riemann solvers replace the exact Riemann solver and all of them are of the following form:
\[
\vec{F}^{AR}(W_L, W_R) \cdot \vec{1}_n = \frac{\vec{f}_C(W_L) + \vec{f}_C(W_R)}{2} \cdot \vec{1}_n - |A| \frac{W_R - W_L}{2}, \tag{4.6}
\]
where \( W_L \) and \( W_R \) are the reconstructed conserved variables from the left and the right side of a face and, by convention, the unit normal \( \vec{1}_n \) is oriented from the left to the right side. The first term in the right-hand-side
of expression (4.6) is just the average of the convective fluxes on the left and the right side of the face. The second term ensures that the data used for the computation of the flux comes from the side from which the physical quantities are propagating. This is called upwinding and it introduces a necessary amount of damping to stabilize the computations. The matrix $|A|$ should be defined such that a correct upwinding is ensured. The different approximate Riemann solvers are distinguished by the corresponding definitions of this matrix; see for instance Rusanov [150] and Roe [143]. The flux values obtained from the approximate Riemann solver at each point are then used to evaluated the surface integrals over the external faces with $(p+1)$-th-order of accuracy by means of an appropriate quadrature-free approach.

The final expression of the spatial discretization with the SV method is then

$$\frac{d\mathbf{W}_{i,j}}{dt} = -\frac{1}{\Delta \Omega_{i,j}} \sum_{m=1}^{N_{\text{intfac}}^i} \int_{\partial \Omega_{i,j} \cap \Delta S_m} \mathbf{f}_C \cdot \mathbf{n} \, dS - \frac{1}{\Delta \Omega_{i,j}} \sum_{m=1}^{N_{\text{extfac}}^i} \int_{\partial \Omega_{i,j} \cap \Delta S_m} \mathbf{F}^{AR} \cdot \mathbf{n} \, dS = \mathbf{R}_{i,j}, \quad (4.7)$$

where $\Delta S_m$ is the surface of the $m$-th face. Expression (4.7) is a system of ODEs in time that approximates the original set of PDEs. It can be solved with any time marching scheme.

### 4.1.2 SV basis polynomials

In order to define the basis functions for the construction of the solution polynomial, a mapped coordinate system $\tilde{\xi} = [\xi_1, \xi_2, \xi_3]^T$ is introduced on each cell. The transformation from the standard to the physical element in the global Cartesian coordinates for the cell $i$ is given by

$$\mathbf{\chi}_i = \begin{bmatrix} \chi_{1,i} \\ \chi_{2,i} \\ \chi_{3,i} \end{bmatrix} = \begin{bmatrix} \chi_{1,i}(\xi_1, \xi_2, \xi_3) \\ \chi_{2,i}(\xi_1, \xi_2, \xi_3) \\ \chi_{3,i}(\xi_1, \xi_2, \xi_3) \end{bmatrix} = \tilde{\mathbf{\chi}}_i(\tilde{\xi}), \quad (4.8)$$

where the corresponding Jacobian matrix and the Jacobian determinant are denoted respectively by $\tilde{J}_i$ and $J_i$. The solution polynomial $\mathbf{W}_i(\tilde{\xi})$ in
cell $i$ has the form

$$W_i(\tilde{\xi}) = \sum_{j=1}^{N_s} W_{i,j} \tilde{L}_{i,j}(\tilde{\xi}),$$  \hspace{1cm} (4.9)

where the $\tilde{L}_{i,j}(\tilde{\xi})$ are a complete polynomial basis (SV basis polynomials) for the polynomials space of degree $p$. In order to solve the reconstruction problem, one should recall that the solution polynomial over a CV must be equal to the corresponding CV-averaged solution, i.e.

$$\frac{1}{\Delta \Omega_{i,j}} \int_{\Delta \Omega_j^\xi} W_i(\tilde{\xi}) J_i d\Omega^\xi = \tilde{W}_{i,j}, \hspace{1cm} j = 1, \ldots, N^s.$$  \hspace{1cm} (4.10)

Therefore, inserting expression (4.9) into identity (4.10) the following set of linear algebraic systems that define the SV basis polynomials are obtained:

$$\frac{1}{\Delta \Omega_{i,j}} \int_{\Delta \Omega_j^\xi} \tilde{L}_{i,m}(\tilde{\xi}) J_i d\Omega^\xi = \delta_{jm}, \hspace{1cm} j, m = 1, \ldots, N^s,$$  \hspace{1cm} (4.11)

where $\delta_{jm}$ is the Kronecker delta function. This shows that in general, the SV basis polynomials depend on the local geometry of the corresponding cell, through the Jacobian determinant $J_i$ \cite{46,183,186}. Consequently, the coefficients of these polynomials should either be stored for each cell, or recomputed on the fly when needed. The first approach requires a very large amount of memory, while the second uses too much computational time. However, if grids with simplex cells are considered, the transformation (4.8) is linear for all the cells, except for those with curved boundary representation, and the Jacobian determinant is a constant proportional to the volume of the cell \cite{46}, i.e.

$$J_i = d! \Delta \Omega_i.$$  \hspace{1cm} (4.12)

In addition, for simplex cells with a linear mapping, one has

$$\frac{\Delta \Omega_{i,j}}{\Delta \Omega_i} = \frac{\Delta \Omega_j^\xi}{\Delta \Omega_i^\xi}, \hspace{1cm} \text{with} \hspace{1cm} \Delta \Omega_i^\xi = \frac{1}{d!}.$$  \hspace{1cm} (4.13)

Therefore, identity (4.11) reduces to

$$\frac{1}{\Delta \Omega_j^\xi} \int_{\Delta \Omega_j^\xi} L_{i,m}(\tilde{\xi}) d\Omega^\xi = \delta_{jm}, \hspace{1cm} j, m = 1, \ldots, N^s.$$  \hspace{1cm} (4.14)
Expression (4.14) shows that for simplex cells with a linear mapping, the SV basis polynomials are the same for each cell. Consequently, only one set of polynomial coefficients must be precomputed and stored before the simulation. Only for boundary cells, where a higher-degree mapping is sometimes needed to discretize properly curved geometries, might have a non-linear mapping. These cells usually form only a small fraction of the total number of cells, and consequently, the basis polynomial coefficients can be precomputed and stored for each individual element before the actual simulation.

For grids with quadrilateral or hexahedral cells, the transformation to the mapped coordinates is linear only for parallelograms and parallelepipeds. However, these type of cells are not flexible enough to mesh a general geometry, and consequently, almost every known implementation of the SV method is limited to simplex cells.

### 4.1.3 Discretization of diffusive terms

The proper discretization of viscous fluxes has been studied extensively in the DG methods and shown to be very important for accuracy and stability. In classical second-order finite volume methods, the solution gradients at an interface are usually computed by averaging the gradients of the neighbour cells sharing the face in consideration. However, for higher-order elements, special care has to be taken in computing the solution gradients. In the late 1970s and early 1980s, Arnold [5] introduced the discontinuous finite element methods known as penalty methods for elliptic operators. More recently, many researchers [15, 16, 27, 42, 44] have applied DG methods to diffusive operators. One procedure was the local discontinuous Galerkin (LDG) method, developed by Cockburn and Shu [42, 44]. Its simplicity and effectiveness have made it the main choice for discretizing the viscous fluxes. In 1997, Bassi and Rebay [15] came up with a symmetrical scheme to discretize the viscous fluxes. In 2000, again Bassi and Rebay came up with a second concept (also referred to as BR2 [17]). A nice unifying framework of these approaches for the DG method can be found in Arnold et al. [6], where more details on the aforementioned methods can be found. In this thesis, the local SV (LSV) approach is considered. It is based on the local DG (LDG) approach which was proposed by Cockburn and Shu [42, 44].

Consider the diffusive flux vector \( \vec{f}_D (\vec{w}, \vec{\nabla} w) \). To evaluate it, an approximation of the gradients of the conserved variables is needed. An approx-
CHAPTER 4. SPATIAL DISCRETIZATION

imation \( \bar{\Phi}_i(\xi) \) can be obtained by computing CV-averaged values as follows:

\[
\overline{\nabla w}_{|i,j} = \frac{1}{\Delta \Omega_{i,j}} \int_{\partial \Omega_{i,j}} w \overline{1}_n dS \\
\approx \frac{1}{\Delta \Omega_{i,j}} \sum_{m=1}^{N_{intfac}} \int_{\partial \Omega_{i,j} \cap \Delta S_m} W_i \overline{1}_n dS \\
+ \frac{1}{\Delta \Omega_{i,j}} \sum_{m=1}^{N_{extfac}} \int_{\partial \Omega_{i,j} \cap \Delta S_m} \hat{W} \overline{1}_n dS \\
= \overline{\Phi}_{i,j},
\]

(4.15)

where \( \hat{W} = \bar{W}(W_L, W_R) \) is an averaged value given by

\[
\bar{W} = \frac{W_L + W_R}{2} - \beta (W_R - W_L).
\]

(4.16)

In the latter expression \( \beta \) defines the bias in the averaging operator. The gradients in cell \( i \) can thus be approximated by the polynomial

\[
\nabla w \big|_i \approx \overline{\Phi}_i(\xi) = \sum_{j=1}^{N^s} \overline{\Phi}_{i,j} \bar{L}_{i,j}(\xi).
\]

(4.17)

The diffusive flux vector is then approximated as \( \hat{\mathbf{f}}_D \left( W_i, \overline{\Phi}_i \right) \) at an internal face. At a cell-face, two values \( \overline{\Phi}_L \) and \( \overline{\Phi}_R \) of \( \overline{\Phi} \) are available, and again an averaged value \( \hat{\overline{\Phi}} = \hat{\overline{\Phi}}(W_L, W_R, \overline{\Phi}_L, \overline{\Phi}_R) \) must be used. With the LSV approach \( \hat{\overline{\Phi}} \) is defined by

\[
\hat{\overline{\Phi}} = \frac{\overline{\Phi}_L + \overline{\Phi}_R}{2} + \beta (\overline{\Phi}_R - \overline{\Phi}_L) + \frac{\alpha}{h_f} (W_R - W_L) \overline{1}_n,
\]

(4.18)

where \( h_f \) is a local length scale associated to a face, defined as

\[
h_f = C \frac{(J_L + J_R)}{\left| \left( J \right) \#L^{-1} \right|^T \overline{1}_n \overline{1}_n^L} = C \frac{(J_L + J_R)}{\left| \left( J \right) \#R^{-1} \right|^T \overline{1}_n \overline{1}_n^R},
\]

(4.19)
with \( \vec{\xi}_n \) the unit normal to the face in the mapped coordinate system associated to a cell. In this work, the constant \( C \) is chosen so that for simplex cells with a linear transformation to the mapped coordinate system, \( h_f \) reduces to

\[
h_f = \frac{\text{dim}}{\text{dim} + 1} \frac{\Delta \Omega_L + \Delta \Omega_R}{\Delta S},
\]

(4.20)

which is the sum of the perpendicular distances between the neighboring cell centers and the face. For tensor product cells, it should be chosen analogously such that

\[
h_f = \frac{\Delta \Omega_L + \Delta \Omega_R}{2\Delta S},
\]

(4.21)

The parameter \( \alpha \) sets the amount of damping added to the gradient, while \( \beta \) defines again the bias in the averaging operator. In Arnold et al. [6], it was shown that \( \alpha > 0 \) results in stable LDG schemes, and no stability problems have been observed for the LSV approach either [179]. Notice that the LSV approach is not compact, since the neighbors of the neighboring cells are needed for the contribution of the external faces to the residuals in a cell. In this thesis, the ‘penalty method’ described in Kannan et al. [91] for which \( \alpha = 1 \) and \( \beta = 0 \) is considered.

### 4.1.4 Spectral volume partition

In order to define the SV method, the partition of the cell into CVs has to be chosen. Depending on the order of accuracy, the CV partition has a certain number of identifying parameters which strongly affect the stability and accuracy properties of the method. Consequently, a suitable choice for them is of paramount importance. The proper definition of CV partitions for the SV method has been subject of several researches [33, 46, 65, 66, 174, 176, 187].

**Second-order accurate scheme**

The partitions into CVs for second-order \((p = 1)\) accurate SV schemes are uniquely defined by a set of rules of thumb [46, 186, 187]. They can be seen for linear, triangular and tetrahedral cells in Figure 4.1. Their stability was confirmed by the wave propagation analysis reported in a series of paper by Van den Abeele et al. [174–176]. The higher-order partitions are defined as follows.
Partitions for 1D cells

A general third-order \((p = 2)\) 1D partition is shown in Figure 4.2(a). It has one DOF, \(\alpha_3 \in [0, 1]\), which can be easily understood from the figure. Figure 4.2(b) shows a general fourth-order \((p = 3)\) 1D partition. There is again one DOF, labeled \(\alpha_4 \in [0, 1]\).

\[
\begin{align*}
\text{(a) } p &= 2. \\
\text{(b) } p &= 3. 
\end{align*}
\]

Figure 4.2: General SV partitions for third- \((p = 2)\) and fourth-order \((p = 3)\) 1D schemes.

The family of third- and fourth-order SV schemes based on the Gauss-Lobatto points, i.e. \(\alpha_3 = 0.50\) and \(\alpha_4 = \cos \left(\frac{\pi}{4}\right)\), was found to be weakly unstable [174]. Van den Abeele and his co-workers carried out a detailed analysis of the 1D SV partition [46, 174] for third-, fourth-, fifth- and sixth-order schemes. In the above works, stable third- \((\alpha_3 = 0.58\) and \(\alpha_3 = 0.65)\) and fourth-order \((\alpha_4 = 0.78\) and \(\alpha_4 = 0.86)\) schemes that are optimal for wave propagation were designed and tested. In the same study, the stability of a recently proposed family of schemes based on the Gauss-Legendre quadrature points [79] was confirmed too.

Partitions for triangles

A general third-order partition of a triangular cell is plotted in Figure 4.3(a). It has two DOFs, which are defined as

\[
\alpha_3 = \frac{|AC|}{|AB|} \in 0, \frac{1}{2} \quad \text{and} \quad \beta_3 = \frac{|AE|}{|AD|} \in 0, \frac{2}{3}, \quad (4.22)
\]
where points $A$, $B$, $C$, $D$ and $E$ are shown in the figure. Figure 4.3(b) shows a general fourth-order partition of a triangular cell. In this case, there are four DOFs:

$$
\alpha_4 = \frac{|AC|}{|AB|} \in \left[0, \frac{1}{2}\right], \quad \beta_4 = \frac{|AE|}{|AD|} \in \left[0, \frac{2}{3}\right], \\
\gamma_4 = \frac{|GD|}{|AD|} \in \left[0, \frac{1}{3}\right] \quad \text{and} \quad \delta_4 = \frac{|AF|}{|AD|} \in \left[\beta_4, \frac{2}{3}\right].
$$

(4.23)

Weak instabilities in three third-order SV schemes found in the literature [104, 186, 188] were identified analytically and verified numerically in Van den Abeele et al. [176]. They correspond to $\alpha_3 = \frac{1}{4}$ and $\beta_3 = \frac{2}{3}$ [186], $\alpha_3 = \frac{1}{4}$ and $\beta_3 = \frac{1}{3}$ [104] and $\alpha_3 = \frac{1}{4}$ and $\beta_3 = \frac{1}{3}$ [188] respectively.

A stable and accurate third-order scheme was proposed by Vanden Abeele and his co-workers [176]. It is defined by the following choice of parameters: $\alpha_3 = 0.1093621117$ and $\beta_3 = 0.1730022492$ (see Figure 4.4(a)). It was found to be more accurate than the SV schemes proposed in Chen [33] and for this reason is used in the present work.

Weak instabilities in two previously proposed fourth-order SV schemes were also identified in Van den Abeele et al. [176]. Those partitions were first presented in Wang and Liu [186] and Harris and Wang [65] and are defined by $\alpha_4 = \frac{1}{15}$, $\beta_4 = \frac{2}{15}$, $\gamma_4 = \frac{1}{15}$, $\delta_4 = \frac{2}{15}$ and $\alpha_4 = 0.12061033$, $\beta_4 = 0.1730022492$.


\[ \beta_4 = 0.12129456, \gamma_4 = 0.066666667, \delta_4 = 0.312260947 \text{ respectively. Again in Van den Abeele et al. [176], a stable and accurate fourth-order scheme was proposed. This scheme shows better wave propagation properties than the previously proposed SV schemes [33]. It is defined by } \alpha_4 = 0.078, \beta_4 = 0.104, \gamma_4 = 0.052 \text{ and } \delta_4 = 0.351 \text{ (see Figure 4.4(b)). This SV partition is used in the present work.} \]

**Partitions for tetrahedra**

In the classical SV framework, the partition for a tetrahedral cell is fully defined by the following three parameters:

\[ \alpha_3 = \frac{|AC|}{|AB|} \in \left[ 0, \frac{1}{2} \right] , \beta_3 = \frac{|AE|}{|AD|} \in \left[ 0, \frac{2}{3} \right] \text{ and } \gamma_3 = \frac{|AG|}{|AF|} \in \left[ 0, \frac{3}{4} \right] . \] (4.24)

Depending on the criteria for the selection of the CVs, third-order SV scheme for tetrahedral cells allows four possible families of partitions which are included in Figures 4.5(a), 4.5(b), 4.5(c) and 4.5(d). The first family of partitions, Figure 4.5(a), was considered in Liu et al. [104] and Chen [32]. It is a simplified case, where it is imposed that the internal faces of the corner CVs are planar. This way, two free parameters remain, namely \( \alpha_3 \) and \( \beta_3 \), with the third parameter \( \gamma_3 \) defined by

\[ \gamma_3 = \frac{3\alpha_3\beta_3}{4\alpha_3 - \beta_3} . \] (4.25)
The other two parameters, namely $\alpha_3$ and $\beta_3$, should satisfy the following constrains: $\beta_3 < 4\alpha_3$ for $\gamma_3$ to be greater than zero, and $\beta_3 \leq \frac{4\alpha_3}{4\alpha_3 + 1}$ to satisfy $\gamma_3 \leq \frac{3}{4}$. For the second and the third family of partitions, shown in Figures 4.5(b) and 4.5(c) respectively, all parameters can be chosen freely. The internal faces of the corner CVs are subdivided into two triangles. The two families differ in the way these faces are subdivided. For the last family of partitions, shown in Figure 4.5(d), the internal faces of the corner CVs are treated as a single bilinear quadrilateral face, and no additional edge is introduced.

Figure 4.5: SV partition families of a tetrahedral cell for third-order ($p = 2$) scheme [175].

In Van den Abeele et al. [175], an extensive study of the variation of
the stability properties in the three-dimensional parameter space corresponding to these schemes was performed. The above study indicates that probably there is no partition that yields a stable third-order SV scheme for tetrahedral cells. In 2010, Harris and Wang [66] have presented a constrained minimization approach in the design of 3D third-order SV schemes. Several new partitions have been proposed which have a reduced maximum real part of the Fourier footprint by up to 20% over the original un-optimized partition proposed by Chen [32]. Numerical simulations have shown that the strength of the instability has been weakened by about an order of magnitude for some cases by employing the constrained minimization approach. However, also in this case, no fully stable third-order SV scheme for tetrahedral cells has been found to date.

4.2 Spectral difference method

An alternative to the SV method, is the spectral difference (SD) method. It is also designed for high-order accuracy on unstructured grids. The SD method originated in the staggered grid multi-domain spectral method [93, 94]. It was generalized to simplex elements by Liu et al. [103]. Extension of the SD method to the Euler equations was described in Wang et al. [189], and to the N-S equations in May and Jameson [111]. A 3D N-S implementation of the SD method for hexahedral grids was presented in Sun et al. [165]. In 2010, Parsani et al. [124, 126] coupled a high-order SD scheme on unstructured hexahedral grids with the Wall-Adapted Local Eddy-viscosity (WALE) model to perform LES.

The SD method has an important advantage over the DG and SV methods, in that no integrals have to be evaluated to compute the residuals, thus avoiding the need for costly high-order integral evaluations.

4.2.1 Discretization of convective term

Consider a problem governed by the system of conservation laws given by Equation (4.3) valid on a domain \( \Omega \) with boundary defined by \( \partial \Omega \). For each cell \( i \), take a mapped coordinate system \( \tilde{\xi} \), with the transformation to the global Cartesian coordinate system given by (4.8), with Jacobian matrix \( \tilde{J} \), and Jacobian determinant \( J \). The convective fluxes projected in the mapped coordinate system \( \tilde{f}_{C,i} \) are thus related to the flux components in
the global coordinate system by

\[
\vec{f}_{C,i} = J_i^{-1} \begin{bmatrix} \vec{f}_{C,i}^\xi \\ \vec{g}_{C,i}^\xi \\ \vec{h}_{C,i}^\xi \end{bmatrix} = J_i^{-1} \begin{bmatrix} \vec{f}_{C,i} \\ \vec{g}_{C,i} \\ \vec{h}_{C,i} \end{bmatrix}.
\]

Therefore, system (4.3) can be written in the mapped coordinate system as

\[
\frac{\partial (J_i \vec{w})}{\partial t} \equiv \frac{\partial \vec{w}^\xi}{\partial t} = -\frac{\partial \vec{f}^\xi}{\partial \xi_1} - \frac{\partial \vec{g}^\xi}{\partial \xi_2} - \frac{\partial \vec{h}^\xi}{\partial \xi_3} = -\vec{\nabla}^\xi \cdot \vec{f}^\xi,
\]

with \(\vec{w}^\xi_i \equiv J_i \vec{w}\) the conservative variables in the mapped coordinate system.

For a \((p+1)\)-th-order accurate \(dim\)-dimensional scheme, \(N^s\) solution points with index \(j\) are introduced at positions \(\vec{\xi}^s_j\) in each cell \(i\), with \(N^s\) given by (4.1) for simplex cells and by (4.2) for tensor product cells. Given the values at these points, a polynomial of degree \(p\) that approximates the solution in cell \(i\), can be constructed. This polynomial is called solution polynomial, and the conserved variables \(\vec{W}_{i,j}\) at the solution points are the solution variables of the SD method. The evolution of these variables is governed by (4.27), evaluated at the solution points.

To estimate the divergence of the mapped fluxes \(\vec{\nabla}^\xi \cdot \vec{f}^\xi_{C,i}\) at the solution points, a set of \(N^f\) flux points with index \(l\) and at positions \(\vec{\xi}^f_l\), supporting a polynomial of degree \(p+1\), is introduced. The evolution of the mapped flux vector \(\vec{f}^\xi_{C,i}\) in cell \(i\) is then approximated by a flux polynomial \(\vec{F}^\xi_{C,i}\), which is obtained by reconstructing the solution variables at the flux points and evaluating the fluxes \(\vec{F}^\xi_{C,i}\) at these points. To ensure a sufficient coupling between the cells, a number of flux points need to lie on the faces or the corners of the cell [103, 175, 189]. Examples of solution and flux point distributions are shown in Figure 4.6 for 1D, 2D triangular and 2D quadrilateral cells and for linear \((p = 1)\) reconstruction.

In order to maintain conservation at a cell level, the flux component normal to a face must be continuous between two neighboring cells. Since the solution at a face is in general not continuous, this requires the introduction of approximate Riemann solvers at those points. Two different approaches were discussed in Wang et al. [189]. The first approach
CHAPTER 4. SPATIAL DISCRETIZATION

Figure 4.6: General solution (∗) and flux points (▲) distribution for second-order ($p = 1$) 1D and 2D SD schemes.
4.2. SPECTRAL DIFFERENCE METHOD

involves the definition of multi-dimensional Riemann solvers, while the second one uses multiple 1D Riemann solvers. One can find more information in the reference mentioned above. Here, the first approach is used [126, 129, 178]. Taking the divergence of the flux polynomial \( \nabla \cdot \vec{F}_{C,i} \) in the solution points results in the following modified form of (4.27), describing the evolution of the conservative variables in the solution points:

\[
\frac{dW_{i,j}}{dt} = - \nabla \cdot \vec{F}_{C,i} \bigg|_j = - \frac{1}{J_{i,j}} \nabla \cdot \vec{F}_{C,i} \bigg|_j = R_{i,j},
\]  

(4.28)

where \( R_{i,j} \) is the SD residual associated to \( W_{i,j} \). This is a system of ordinary differential equations, in time, for the unknowns \( W_{i,j} \), which can be solved numerically using any method for such a system.

Notice that for \( p = 0 \), the SD method, like the DG and SV methods, reduces to the classical first-order accurate FV method. Furthermore, no special treatment is required for cells with a non-linear transformation to the mapped coordinate system, since the modified values of \( \vec{J}_i \) and \( J_i \) can just inserted into expression (4.28).

4.2.2 SD basis polynomials

For a \( \text{dim} \)-dimensional \( (p+1) \)-th order scheme, the \( N^s \) solution points support a set of solution basis polynomials \( L^s_j \left( \xi \right) \) of degree \( p \) (SD solution basis polynomials). Using these polynomials, the solution in cell \( i \) is approximated with a \( p \)-th order polynomial as follows

\[
W_i \left( \xi \right) = \sum_{j=1}^{N^s} W_{i,j} L^s_j \left( \xi \right).
\]  

(4.29)

In order to solve the reconstruction problem, one should recall that the solution polynomial at solution points must be equal to the solution variable, i.e.

\[
W_i \left( \xi^s_j \right) = W_{i,j}, \quad j = 1, \ldots, N^s.
\]  

(4.30)

Inserting expression (4.29) into identity (4.30) results in a set of linear algebraic systems that define the SD basis polynomials:

\[
L^s_j \left( \xi^s_m \right) = \delta_{jm}, \quad j, m = 1, \ldots, N^s.
\]  

(4.31)
The $N_f$ flux points support a set of flux basis polynomials $L_f^f(\xi)$ of degree $p + 1$ (SD flux basis polynomials). The degree $p + 1$ flux polynomial in cell $i$ is then given by

$$\mathbf{F}_{C,i} \left( \xi \right) = \sum_{l=1}^{N_f} \mathbf{F}_{C,i,l} L_f^f(\xi).$$  \hspace{1cm} (4.32)

In this case, the reconstruction problem is solved by imposing that the flux polynomial at flux points must be equal to the flux at the flux points. Consequently, the flux basis polynomials are defined by a set of linear algebraic systems of the form

$$L_f^f(\xi_m) = \delta_{lm}, \quad l, m = 1, \ldots, N_f.$$  \hspace{1cm} (4.33)

Unlike the SV basis polynomials, the SD solution and flux basis polynomials are always independent of local geometry of the cell, and thus of the cell index $i$. Consequently, independently of the type of cell, both SD solution and flux basis polynomials are the same for all cells and only two sets of polynomial coefficients have to be precomputed and stored before the actual simulation.

### 4.2.3 Discretization of diffusive terms

The treatment of the diffusive terms with the SD method is also derived from approaches that were developed for the DG method. In this thesis, the second approach of Bassi and Rebay (BR2) \cite{17} is considered. This approach was reported for the first time in combination with the SD method by Van den Abeele et al. \cite{178}.

Consider the diffusive flux vector $\mathbf{f}_D \left( w, \mathbf{\nabla} w \right)$. Defining the vectors $\mathbf{j}_i^{\xi_1}$, $\mathbf{j}_i^{\xi_2}$ and $\mathbf{j}_i^{\xi_3}$ as

$$J_i \mathbf{J}_i^{-1} = \begin{bmatrix} \mathbf{j}_i^{\xi_1} & \mathbf{j}_i^{\xi_2} & \mathbf{j}_i^{\xi_3} \end{bmatrix}^T,$$  \hspace{1cm} (4.34)

a polynomial approximation of the gradients $\mathbf{\nabla} w$ at the solution points is obtained by

$$\mathbf{\nabla} w \bigg|_{i,j} \approx \frac{1}{J_{i,j}} \begin{bmatrix} \frac{\partial \mathbf{W}_i \mathbf{j}_i^{\xi_1}}{\partial \xi_1} & \frac{\partial \mathbf{W}_i \mathbf{j}_i^{\xi_2}}{\partial \xi_2} & \frac{\partial \mathbf{W}_i \mathbf{j}_i^{\xi_3}}{\partial \xi_3} \end{bmatrix}_j = \mathbf{\Phi}_{i,j},$$  \hspace{1cm} (4.35)

56
4.2. SPECTRAL DIFFERENCE METHOD

where \( \hat{\mathbf{W}} \) is a polynomial of degree \( p + 1 \), defined by its values at the flux points. In an internal flux point, this is just the value of the polynomial \( \mathbf{W}_i \). In a face flux point, it is equal to an average \( \hat{\mathbf{W}} \), of the two available values \( \mathbf{W}_L \) and \( \mathbf{W}_R \):

\[
\hat{\mathbf{W}} = \frac{\mathbf{W}_L + \mathbf{W}_R}{2}.
\]  (4.36)

The gradients in cell \( i \) are then approximated by \( \hat{\Phi}_i \), given by

\[
\hat{\nabla} \mathbf{w} \bigg|_i \approx \hat{\Phi}_i (\hat{\xi}) = \sum_{j=1}^{N_s} \Phi_{i,j} L^*_j (\hat{\xi}).
\]  (4.37)

The diffusive flux vector is thus approximated as \( \hat{\Phi}_i \), in an internal flux point. At a face, two values, \( \Phi_L \) and \( \Phi_R \), of \( \Phi \) are available. An averaged value \( \hat{\Phi} \) defined as

\[
\hat{\Phi} = \frac{\hat{\nabla} \mathbf{W}_L + \hat{\nabla} \mathbf{W}_R}{2} + \frac{\tilde{\Lambda}_L + \tilde{\Lambda}_R}{2}
\]  (4.38)

is used. In (4.38) the lifting operators \( \tilde{\Lambda}_L \) and \( \tilde{\Lambda}_R \) associated to a cell-face are again polynomials in the neighbouring cells, defined by their values at the solution points:

\[
\tilde{\Lambda}_{L(R),j} = \frac{1}{J_{L(R),j}} \left[ \hat{\nabla} \xi \delta \mathbf{W}_{L(R)} \right]_j.
\]  (4.39)

In this expression, \( \delta \mathbf{W}_{L(R)} \) is a polynomial of degree \( p + 1 \), defined by its values at the flux points:

\[
\delta \mathbf{W}_{L(R),l} = \begin{cases} 
(W_{R,l} - W_{L,l}) \left( J \hat{\xi} J^{-1} \right)^T \bar{\mathbf{I}}_n \\ 0 \end{cases} \quad \bar{\mathbf{I}}_n \quad l \in \text{curr. face elsew}\text{ise}
\]  (4.40)

The BR2 approach is fully compact, as only the immediate neighbors are required for the computation of the residuals in a cell. Note that for \( p = 0 \), the gradients of the solution polynomials are always zero (\( \hat{\nabla} \mathbf{W}_L = \hat{\nabla} \mathbf{W}_R = 0 \), and the gradients at a face are approximated by the lifting operators
alone, which means that $\alpha$ must be equal to one for consistency [90].

The normal component of the diffusive flux vector is thus evaluated as $\vec{f}_D \left( \frac{\vec{W}_L + \vec{W}_R}{2}, \hat{\vec{\Phi}} \right) \cdot \vec{l}_n$ in a face flux point. The tangential component is computed using the internal or the averaged value of $\hat{\vec{\Phi}}$.

Notice that, if one is interested in the local SD approach (LSD), the definitions of $\hat{\vec{W}}$ and $\hat{\vec{\Phi}}$ are identical to those discussed for the LSV approach, namely Equation (4.16) and Equation (4.18).

### 4.2.4 Component-wise flux point distribution

The flux point distribution for quadrilateral cells shown in Figure 4.6(c) is not the one that is used in practical implementations of the SD method. Instead, cells with component-wise flux point distributions, like the ones shown in Figure 4.7, is used, [93, 94, 133, 164, 178].

![Figure 4.7: Quadrilateral SD cells with general component-wise flux point distribution for second-order ($p = 1$) scheme. Solution ($\circ$), $\xi_1$- ($\blacktriangledown$) and $\xi_2$-flux points ($\blacktriangle$).](image)

Different set of flux points are used for different components of the mapped flux vector. For the general $f^{\xi_i}_i$-component, a set of flux points that supports a polynomial $F^{\xi_i}_i$ of degree $(p + 1)$ in $\xi_1$ and of degree $p$ in $\xi_2$ (and in $\xi_3$, in the case of hexahedral cells) is defined. These flux points are labeled '$\xi_1$-
flux points' and are used for the approximation of the derivative of $f^\xi_i$ to $\xi_1$. For the $g^\xi_i$- and $h^\xi_i$-components, ‘$\xi_2$’- and ‘$\xi_3$-flux points’, supporting polynomials $G^\xi_i$ and $H^\xi_i$, are introduced analogously for the computation of the derivatives to $\xi_2$ and $\xi_3$ respectively.

The main advantage of using component-wise flux point distributions is that the reconstruction of the solution at the flux points and the computation of the flux derivatives at the solution points become one-dimensional operations if the solution and flux points are aligned correctly, as shown in Figure 4.7. This implies significant savings in computational time with respect to the multi-dimensional operations that are required for general quadrilateral cell SD schemes of the type shown in 4.6(c). Also notice that, since each set of flux points is responsible for a single component of the mapped flux, only the flux component normal to the face has to be computed at face flux points. Thus, a single one-dimensional Riemann flux for the convective terms and the normal flux component for the diffusive terms are used at face flux points. In this case, there are no corner flux points.

Obviously, the use of component-wise flux point distributions is only possible for quadrilateral and hexahedral cells.

4.2.5 Solution and flux points distribution

In 2008, in Van den Abeele et al. [177] showed an interesting property of the SD method, namely that it is independent of the positions of its solution points in most general circumstances, for both simplex and tensor-product cells. In the above work it has been shown that the distribution of the solution points has very little influence on the properties of the SD schemes, and in fact, for linear problems, different distributions lead to identical results. This property greatly simplifies the design of SD schemes, since only the flux point distributions have to be specified. It also implies an important improvement in efficiency, since the solution points can be placed at flux points and thus a significant number of solution reconstructions can be avoided.

1D cells

Respecting the symmetries of the cells and the minimum number of flux points at a face, the flux point distributions of second-order SD scheme is uniquely defined. Moreover, recalling that the SD method is independent
of the positions of its solution points, the practical implementation of the solution and flux points distribution is shown in Figure 4.8. Its stability was confirmed by the wave propagation analysis shown in Van den Abeele et al. [177].

![Figure 4.8](image)

Figure 4.8: Solution point distribution for second-order ($p = 1$) 1D SD cells, with all solution points at flux points. Solution (○) and flux points (▲).

Recently, Huynh [79] has proven that using the Legendre-Gauss quadrature points and the two end points of the cell as flux points results in stable 1D SD schemes of arbitrary orders of accuracy. Practical implementation of higher-order 1D SD schemes is carried out respecting the symmetries of the cell and placing most the solution points at flux points. For example, the distribution of the points for third-order SD scheme is shown in Figure 4.9. In this figure, the free parameter $\alpha_3 \in ]0, +1[$, which was already introduced for the 1D SV scheme, is also indicated. In order to obtain the Legendre-Gauss quadrature points, $\alpha_3$ is set to $\frac{1}{\sqrt{3}} \approx 0.577$.

![Figure 4.9](image)

Figure 4.9: Solution point distribution for third-order ($p = 2$) 1D SD cells, with symmetric and most solution points at flux points. Solution (○) and flux points (▲).

Stability and accuracy properties of the 1D SD schemes proposed by Huynh [79] was verified by Van den Abeele [46] with orders of accuracy up to thirty.

2D triangular cells

Third-order SD schemes for triangular cells with different Riemann flux approaches were examined using the wave propagation analysis in Van den Abeele et al. [177]. The analysis indicates that no stable flux point
distribution for such schemes exists, with neither the semi-upwind nor the full-upwind Riemann flux approach. The correctness of the analysis was verified with numerical tests.

2D quadrilateral and 3D hexahedral cells

In Section 4.2.4 the component-wise flux point distributions has been discussed. The main advantage of that approach is that the reconstruction of the solution at the flux points and the computation of the flux derivatives at the solution points become one-dimensional operations if the solution are aligned with the flux points. Following this idea, SD schemes for quadrilateral and hexahedral cells are entirely defined by a 1D SD cell, which is commonly labelled ‘1D source cell’. Moreover, in Huynh [79] it was proven that for quadrilateral and hexahedral cells, tensor product flux point distributions based on a 1D flux point distribution consisting of the end points and the Legendre-Gauss quadrature points, lead to stable schemes for arbitrary values of $p$.

In Figures 4.10 and 4.11, symmetric second- and third-order quadrilateral ‘locally 1D’ SD cells, with component-wise flux point distributions, are illustrated.

Figure 4.10: Symmetric second-order ($p = 1$) quadrilateral ‘locally 1D’ SD cells, with component-wise flux point distributions. All solution points at flux points. Solution (○), $\xi$- (▼) and $\eta$-flux points (▲).
CHAPTER 4. SPATIAL DISCRETIZATION

Figure 4.11: Symmetric third-order \((p = 2)\) quadrilateral ‘locally 1D’ SD cells, with component-wise flux point distributions. Most solution points at flux points. Solution (○), \(\xi\)- (▼) and \(\eta\)-flux points (▲).

Always stable higher-order SD schemes for quadrilateral and hexahedral cells are derived from their stable 1D counterparts [79].

4.2.6 Grid filter width for the subgrid-scale model

In Section 3.2.2 it has been shown that in the WALE model the grid filter width \(\Delta\) is used to compute the turbulent eddy-viscosity, i.e. Equation (3.48). In general \(\Delta\) is an unknown function and it has to be defined to have a closed subgrid-scale model. Often, the grid filter width is taken to be proportional to the smallest resolvable length scale of the discretization and for a general cell with index \(i\) is usually approximated by

\[
\Delta_i = \left( \prod_{k=1}^{\text{dim}} h_k \right)^{1/\text{dim}},
\]  

(4.41)

where \(h_k\) is the size of the cell in the \(k\)-direction. However, at the face flux points, two values of \(\Delta\) are available, labeled \(\Delta_L\) and \(\Delta_R\). Consequently, an averaged value for the filter width is generally used,

\[
\hat{\Delta}_i = \frac{\Delta_L + \Delta_R}{2}.
\]  

(4.42)
4.2. SPECTRAL DIFFERENCE METHOD

For classical FV methods, Equation (4.42) uniquely defines the grid filter width because for these schemes the flux points always lie at the cell face. However, the same reasoning applied to a general SD method implies a natural formulation of the grid filter width based on the Jacobian determinant of the transformation defined by Equation (4.8). In addition, since in the SD scheme each cell has interior solution points and a high-order polynomial approximation occurs in the cell, it is natural to choose the filter width depending on the order of the polynomial. For the SD method, the order of the polynomial reconstruction is taken into account through the division of the Jacobian determinant by the number of solution points, i.e. $N_s(p, dim)$. Consequently, for each cell with index $i$ and each flux points with index $l$ and positions $\xi^f_l$, the following definition of filter width is given:

$$
\Delta_{i,l} = \left[ \frac{1}{N_s} \det \left( \frac{\partial}{\partial \xi^f_l} J_i \right) \right]^{1/dim} = \left( \frac{J_{i,l}}{N_s} \right)^{1/dim}.
$$

(4.43)

Equation (4.43) uniquely defines the filter width for the internal flux points but for the face flux points two values of the Jacobians determinant are available, labeled $J_{L,l}$ and $J_{R,l}$. Consequently, an averaged value is again used, i.e.

$$
\hat{\Delta}_{i,l} = \left[ \frac{\det \left( \frac{\partial}{\partial \xi^f_l} J_{L,i} \right) + \det \left( \frac{\partial}{\partial \xi^f_l} J_{R,i} \right)}{2 N_s} \right]^{1/dim} = \left( \frac{J_{L,i} + J_{R,i}}{2 N_s} \right)^{1/dim}.
$$

(4.44)

Notice that with this approach, the cell filter width is not constant in one cell, but it varies because the Jacobian matrix is a function of the positions of the flux points. Moreover, for a given mesh, the number of solution points depends on the order of the SD scheme, so that the grid filter width varies by varying the order of the scheme. The proposed approach is valid for high-order approaches and it is consistent because the filter width is a function of the the polynomial order through the number of solution point. In fact, the grid filter width decreases by increasing the polynomial order of the approximation.

This new definition for the grid filter width with high-order SD scheme was proposed in Parsani et al. [126], for 2D large eddy simulations with
CHAPTER 4. SPATIAL DISCRETIZATION

SD method (SD-LES). Recently, Parsani et al. [122, 123] applied the SD-LES approach to 3D LES in a muffler.

4.3 Concluding remarks

In the previous sections, the mathematical formulations of the SV and SD methods for a general system of conservation laws have been discussed. With the SV method it has been shown that the basis polynomials depend on the local geometry of the corresponding cell, i.e. the CVs partition. Therefore, the polynomials coefficients should either be stored for each cell, or recomputed on the fly when needed. The first approach requires a prohibitively large amount of memory, while the second uses too much computational time. However, for grids with simplex cells, the transformation is linear for all the cells with straight edges and the SV basis polynomials are the same for each element. Thus, only one set of polynomial coefficients can be precomputed and stored before the simulation. For grids with quadrilateral or hexahedral cells, the transformation to the mapped coordinates is linear only for respectively parallelograms or parallelepipeds. However, these type of cells are not flexible enough to mesh a general geometry, and consequently, the implementation of the SV method is limited to simplex cells. It has been pointed out that the SV method is not uniquely defined for orders of accuracy higher than two. In fact, like the SD method, the SV method has a certain number of parameters that must be specified. These parameters define the partition of cells into CVs and they affect strongly the stability and accuracy properties of the method. For 1D and 2D SV schemes several stable and accurate partitions were proposed in literature. However, for the 3D case, no fully stable third-order SV scheme for tetrahedral cells has been yet found and weak instabilities have been discovered in all partitions for tetrahedra reported in literature. Consequently, the application of 3D high-order SV methods to general flow problems is still an issue. These schemes are not yet robust. Therefore, they can not be routinely employed in practical calculations, such as LES.

Unlike the SV basis polynomials, the SD solution and flux basis polynomials are always independent of the transformation to the mapped coordinate system, and consequently of the cell index. Therefore, independently of the type of cell, both SD solution and flux basis polynomials are the same for all cells and only two sets of polynomial coefficients have to be precomputed and stored before the simulation. The wave propagation analysis of
4.3. CONCLUDING REMARKS

the third-order SD schemes for triangular cells reported in literature indicates that no stable flux point distribution for such schemes exists. On the contrary, for quadrilateral and hexahedral cells, tensor product flux point distributions based on a 1D flux point distribution consisting of the end points and the Legendre-Gauss quadrature points, was shown to lead to stable schemes for arbitrary values of the order of accuracy. Consequently, always stable higher-order SD schemes for quadrilateral and hexahedral cells can be derived from their stable 1D counterparts.

Based on the important results summarized above, the coupling between the SV method and the non-linear LU-SGS algorithm, which will be presented in Chapter 5, has been done only for 2D SV scheme and steady-state computations. In fact, this PhD work has been accomplished in the framework of the IWT Project SBO 050163 ("Simulation and design tools towards the reduction of aerodynamic noise in confined flows"), and its final goal was the development of an efficient N-S/LES solver for high-order accurate schemes. In order to achieve the ultimate objective, since no stable 3D SV partition seems to exist, the author has mainly focused on the promising high-order SD method. In this context, the SD method has been coupled for the first time with the LES approach (SD-LES), through a new definition of a grid filter width, as already discussed in Section 4.2.6.
In the previous chapter, the spectral volume and spectral difference methods have been used for the spatial discretization of the equations of fluid dynamics. It has been shown that the spatial derivatives are no longer stated explicitly in terms of the spatial independent variables and a system of ODEs in time has been obtained. This system approximates the original set of PDEs and can theoretically be solved with any time marching scheme.

High-order explicit strong stability preserving (SSP) or total variation diminishing (TVD) Runge-Kutta (RK) schemes, developed by Shu [159], Shu and Osher [119] and Gottlieb and Shu [58], were introduced for DG space discretizations by Cockburn et al. [37, 39, 41, 43]. The same explicit methods were used for time marching SV and SD discretizations, respectively by Wang et al. [183, 186, 191] and Liu et al. [103]. May and Jameson [111] coupled their SD code with the popular four-stage fourth-order accurate Runge-Kutta scheme [87]. The motivations for using these matrix-free explicit methods are high-order of accuracy in time, simplicity in implementation, and low computational cost per time step. However, explicit methods are subjected to severe Courant-Friedrichs-Lewy (CFL) stability limitations. In fact, with spatially high-order methods, the systems of ODEs are generally stiff, and the stiffness increases with the order of the method.

Atkins and Shu [10] demonstrated that the increase of the order of accuracy of the DG method or the degree of the polynomials that are used for the expansion in the local finite element space rapidly lowers the maximum CFL number and causes increasingly more stringent time step lim-
CHAPTER 5. TIME DISCRETIZATION

...lations. Analogous limitations were encountered with the SV method by Kannan et al. [91], Haga et al. [63] and Parsani et al. [121, 128, 130] and with the SD method by Sun et al. [167] and Premasuthan et al. [133].

The CFL stability for spatially high-order discretizations is significantly lower than the CFL stability of equivalent-order finite volume or finite difference high-order methods. As a result, the computational effort of explicit methods for solutions of problems that require unstructured meshes with spatially high-order discretizations is so high that the overall numerical scheme often becomes inefficient. This limitation is more dramatic in case of viscous problems, where the solver must also be able to deal with the geometrical stiffness imposed by the Navier-Stokes grids where high-aspect ratios occur near walls. This is called grid clustering and it increases with the Reynolds number. In the case of compressible solvers there is an additional stiffness when solving for low speed flows caused by the disparate eigenvalues of the system.

Implicit time integration schemes can be used to deal with these problems. These schemes can advance the solution with significantly larger time steps compared to explicit methods. Two implicit time marching schemes, namely the backward Euler (BE) scheme and second-order backward difference formula (BDF2) with variable time step, are used here for the time discretization. The BE scheme, which is first-order accurate in time, is used to solve steady flow problems, whereas the BDF2 is used for unsteady flow simulations. Both schemes are A-stable, so that if they are used to solve the linear Cauchy problem

\[
\begin{align*}
\frac{dy}{dt} &= \Theta y(t), \quad t > 0 \\
y(0) &= y_0,
\end{align*}
\]

(5.1)

where \( \Theta \) is a complex number with negative real part \( \Theta \in \mathbb{C}, \Theta_R < 0 \), their numerical solutions at time level \( t^n \), i.e. \( Y^n \), satisfy the condition \( |Y^n| \rightarrow 0 \), for \( t^n \rightarrow \infty \) [135]. This is a very important property because the analytical solution of the Cauchy problem (5.1) is an exponential solution \( y(t) = e^{\Theta t} \) that goes to zero for \( t \rightarrow \infty \). The stability region of A-stable schemes is then the complete left part of the complex plane, namely

\[ A = \{ z = \Delta t^n \Theta \in \mathbb{C}^- \} \, . \]

The A-stability is an important property of the time marching scheme for solving systems of stiff ODEs, such as the systems arising from the spatial discretization with high-order accurate schemes. In fact, the Fourier foot-
print\(^1\) of a spatial scheme, which gives an indication of the stiffness of the problem, grows fast with increasing order of the method. Given a stable spatial discretization, the \(A\)-stability guarantees the stability for the combination of the spatial scheme and the time marching scheme for any order of accuracy of the space method. Notice that both BE and BDF2 schemes are also \(L\)-stable, i.e. they are \(A\)-stable and their amplification factors go to zero for \(z \rightarrow -\infty\) [135]. For more details about the Fourier footprint of a spatial scheme and its stability condition, the stability region of a temporal discretization and the global stability condition, the reader is referred to Appendix A.

Implicit temporal schemes imply the solution of non-linear algebraic systems. If the algorithm for such systems is not efficient then the schemes may be more expensive than explicit ones. Recently, there has been some research on RK schemes and backward difference formulae (BDF) [20, 30, 169]. In Bijl et al. [20], implicit Runge-Kutta (I-RK) solvers were investigated in combination with a standard cell-centered finite volume scheme with artificial dissipation added for stability. It was observed that significant potential improvements in the temporal efficiency of implicit schemes could be achieved from algebraic solver developments. In Swanson et al. [169] the convergence of an explicit Runge-Kutta (E-RK) scheme with \(h\)-multigrid was accelerated by preconditioning with a fully implicit operator and the resulting RK/Implicit Residual scheme was used as a smoother for an \(h\)-multigrid algorithm. It was demonstrated that the implicit preconditioner reduced the computational time of a well-tuned E-RK scheme by a factor between four and ten. Both studies [20] and [169], concluded that solver improvements can be more dramatic than improvements in integration techniques. For spatially high-order accurate methods, several algorithms have been proposed and used to solve efficiently the non-linear algebraic system. Element Jacobi methods were used by e.g. Helenbrook and Atkins [69] for the DG method. Newton-Raphson GMRES solvers with preconditioners were used in combination with DG schemes by Bassi and Rebay [13], and in combination with SV and SD schemes by Van den Abeele et al. [178, 179]. A matrix-free Krylov method was applied to DG schemes by Rasetarinera and Hussaini [139] and to SD schemes by May et al. [110]. Line-implicit solvers were combined with a DG method by Fidkowski et al. [48].

\(^1\)The Fourier footprint is the collection of all eigenvalues in the complex plane for all wave numbers of the modified dispersion relation which is obtained by wave propagation analysis of a spatial scheme. It grows faster than linear with \(p\).
CHAPTER 5. TIME DISCRETIZATION

In this work, an efficient algebraic algorithm, namely the non-linear lower-upper symmetric Gauss-Seidel algorithm (LU-SGS), is used to solve the non-linear algebraic systems arising from the implicit time discretization. The LU scheme was started by Jameson and Turkel [88] and later reformulated to use symmetric Gauss-Seidel by Jameson and Yoon [89] in the context of second-order central schemes. It was recently rediscovered by Sun et al. [167, 168] and adapted for use with SD schemes for steady state computations. The first application of the non-linear LU-SGS in combination with the SV scheme and a full $p$-multigrid algorithm was reported by Parsani et al. [128]. Applications of the non-linear LU-SGS in combination with $p$-multigrid to the SD method were reported by Premasuthan et al. [133] and May et al. [110]. In 2009, in Parsani et al. [129] the non-linear LU-SGS algorithm in combination with the BDF2 was coupled with the SD method to solve 2D unsteady Navier-Stokes equations for laminar/turbulent test cases. More recently, in Parsani et al. [122, 124, 126], the algorithm was used in combination with the SD method and an LES approach for the simulation of 2D and 3D turbulent flows.

The formulation of the non-linear LU-SGS algorithm coupled with the BE scheme and the BDF2 is described in the following sections. Extensions to other implicit time marching schemes is possible. The interested reader is referred to Appendix D, where the non-linear LU-SGS algorithm is used in combination with explicit-first-stage, single-diagonal-coefficient, diagonally-implicit Runge-Kutta (ESDIRK) schemes. The coupling of the non-linear LU-SGS solver with ESDIRK schemes was presented by Parsani et al. [128] for quasi-1D Euler equations.

5.1 Backward Euler scheme

Consider the semi-discrete form of a system of conservation laws associated to a general spatial discretization:

$$\frac{\partial W_{cc}}{\partial t} = R_{cc} (W_{cc}(t), \{W_{nb}(t)\}) = R_{cc} (W(t)),$$

(5.2)

where subscripts $cc$ and $nb$ denote the current cell and the neighbouring cells that contribute to the residual $R_{cc}$ respectively. For example, this system can be the semi-discrete form (4.7) or (4.28) with the inclusion of the discretized diffusive flux, as shown in Sections 4.1.3 and 4.2.3. Approxi-
mate the temporal derivative with the BE scheme

\[
\frac{W_{cc}^{n+1} - W_{cc}^n}{\Delta t} - \left[ R_{cc}(W^{n+1}) - R_{cc}(W^n) \right] = R_{cc}(W^n), \tag{5.3}
\]

where superscripts \( n \) and \( n+1 \) denote the time levels \( t^n \) and \( t^{n+1} \), and \( \Delta t = t^{n+1} - t^n \) is the time step. Let \( \Delta W_{cc} = W_{cc}^{n+1} - W_{cc}^n \) and \( R_{cc}(W_{cc}^n) = R_{cc}^n \).

Linearizing the residual using the Taylor expansion about \( W_{cc}^n \) gives

\[
R_{cc}^{n+1} - R_{cc}^n \approx \frac{\partial R_{cc}}{\partial W_{cc}} \bigg|_n \Delta W_{cc} + \sum_{nb} \frac{\partial R_{cc}}{\partial W_{nb}} \bigg|_n \Delta W_{nb}. \tag{5.4}
\]

Therefore, the fully linearized equations for (5.2) can be written as

\[
\left[ - \frac{\partial R_{cc}}{\partial W_{cc}} \bigg|_n + \frac{I}{\Delta t} \right] \Delta W_{cc} - \sum_{nb} \frac{\partial R_{cc}}{\partial W_{nb}} \bigg|_n \Delta W_{nb} = R_{cc}^n, \tag{5.5}
\]

where \( I \) is the identity matrix. However, Equation (5.5) requires too much memory to store the LHS implicit Jacobian matrices. Consequently, the most recent solution for the \( nb \) cells is used,

\[
\left[ - \frac{\partial R_{cc}}{\partial W_{cc}} \bigg|_n + \frac{I}{\Delta t} \right] \Delta W_{cc}^{m+1} = R_{cc}^n + \sum_{nb} \frac{\partial R_{cc}}{\partial W_{nb}} \bigg|_n \Delta W_{nb}^*, \tag{5.6}
\]

with superscript * denoting the most recent solution when doing forward and backward sweeps. Superscript \( m+1 \) refers to the actual symmetric Gauss-Seidel (SGS) sweep, i.e. \( \Delta W_{cc}^{m+1} = W_{cc}^{n+1,m+1} - W_{cc}^n \).

In Swanson et al. [169], Equation (5.6) is solved with a first order upwind scheme with the Jacobians calculated each time step and not stored. However, this may not be an efficient solution for high-order methods because the computation of the Jacobian \( \frac{\partial R_{cc}}{\partial W_{nb}} \) is time-consuming.

To avoid the computation and the storage of the off-diagonal block matrices, expression (5.6) is further manipulated as follows [128, 168]:

\[
R_{cc}^n + \sum_{nb} \frac{\partial R_{cc}}{\partial W_{nb}} \bigg|_n \Delta W_{nb}^* \approx R_{cc}(W_{cc}^*, \{W_{nb}^*\}) - \frac{\partial R_{cc}}{\partial W_{cc}} \bigg|_n \Delta W_{cc}^* = R_{cc}^* - \frac{\partial R_{cc}}{\partial W_{cc}} \bigg|_n \Delta W_{cc}^*. \tag{5.7}
\]
Let $\Delta W_{cc}^{m+1} \equiv \Delta W_{cc}^{n+1} - \Delta W_{cc}^* = W_{cc}^{n+1,m+1} - W_{cc}^*$. Consequently, combining (5.7) with (5.6) results in expression (5.8)

$$
\left[ - \frac{\partial R_{cc}}{\partial W_{cc}} \right]^{n} + \frac{I}{\Delta t} \Delta W_{cc}^{m+1} = R_{cc}^* - \frac{\Delta W_{cc}^*}{\Delta t}.
$$

(5.8)

This implicit solver is denoted as ‘LU-SGS + BE’ in the remainder of the work, where BE stands for backward Euler scheme. The linear system (5.8) is solved with multiple cell-wise symmetric forward and backward sweeps with a prescribed tolerance on the change $\Delta W_{cc}^{m+1}$ and/or a maximum number of symmetric forward and backward sweeps. The inverse of the small Jacobian matrices in the left hand side of this expression is computed using a LU decomposition with pivoting at the beginning of each time iteration, which makes the solution of the small linear algebraic systems much more efficient during subsequent SGS sweeps. If the time step $\Delta t$ is not too large, it is possible to ‘freeze’ the cell-wise Jacobian matrices during a certain number of time iterations, which also leads to an increase in efficiency. Note that solving Equation (5.8) to machine zero implies that Equation (5.2) is satisfied, i.e. the non-linear system of equations is solved exactly. For this reason the algorithm is called non-linear LU-SGS algorithm. If Equation (5.2) is satisfied and the CFL number is smaller or equal to one, the implicit LU-SGS + BE is actually a first-order accurate physical time-advancing scheme. Because of the Gauss-Seidel nature of this LU-SGS method, where the latest available solution in the neighboring cells is used to update the solution in a cell, information travels much faster through the domain than with a traditional explicit solver, where only the solution at the previous time step or stage is used to update the solution in a cell. Besides the cell-wise implicitness of the algorithm, the Gauss-Seidel nature is one reason why the LU-SGS method is much more efficient than an explicit solver.

The initial guess for $W_{cc}^{n+1}$ is $W_{cc}^n$. For steady flow problems, the $L_2$ norm of the solution variation or right-hand side (RHS) of Equation (5.8) are monitored for convergence in this work. Moreover, for steady flow simulations, it is not necessary to drive the RHS of Equation (5.8) to machine zero, but it is more efficient to limit the maximum number of SGS sweeps to damp high-frequency error components and/or set a threshold for some norm of the change $\Delta W_{cc}^{m+1}$. More details about the values of the parameters used to solve the system (5.8) are given in Chapters 7 and 8 where the results of the numerical simulations are presented.
5.1. BACKWARD EULER SCHEME

Since only the diagonal block Jacobians are required to solve the system of ODEs, the total number of real variables needed to store these Jacobians on a grid with $N$ tetrahedral cells with solution polynomial degree $p$ is

$$N \left( \frac{(p+1)(p+2)(p+3)}{6} \right) \times \# \text{physical variables}^2. \quad (5.9)$$

For a grid with $N$ hexahedral cells with solution polynomial degree $p$, this number is

$$N \left( (p+1)^3 \times \# \text{physical variables} \right)^2. \quad (5.10)$$

From expressions (5.9) and (5.10), it is clear that the non-linear LU-SGS method requires significantly less memory than classical methods that use the full Jacobian matrix, like for instance the Newton-GMRES algorithm\(^2\) [92, 151]. Nevertheless, the required amount still increases with $p$ to the power six. For instance, in expression (5.9), the coefficient, which multiplies the number of physical variables, is 16, 100, 400 respectively for $p = 1, 2, 3$.

In this study, the Jacobian matrices are numerically obtained using the following numerical differencing

$$\frac{\partial R_{cc}}{\partial W_{cc}} = \frac{R_{cc}(W_{cc} + \delta W_{cc}, W_{nb}) - R_{cc}(W_{cc}, W_{nb})}{\delta W_{cc}}, \quad (5.11)$$

where $\delta W_{cc}$ is a small value computed as

$$\delta W_{cc} = \|W_{cc} - W_{ref}\| \varepsilon. \quad (5.12)$$

$W_{ref}$ and $\varepsilon$ are respectively the reference conservative variables and a small constant set equal to something larger than the square root of machine precision $\varepsilon_{mach}$ [92]. In the present work, with 64-bit double precision, $\varepsilon$ was set to $10^{-5}$. Notice that this numerical approach results in straightforward operations for arbitrarily complex residual operator.

The cells numbering potentially has an important influence on the convergence of the scheme due to the Gauss-Seidel nature of the implicit LU-SGS, see for instance Sharov and Nakahashi [158]. However, in practice it is observed that the symmetric forward and backward sweeps strongly

\(^2\)For more details about the Newton-GMRES method the reader is referred to Appendix C.
reduce the effect of the cells numbering and consequently only a small influence on the convergence is observed. For this reason, in the present work, the order of the cells in the cell index list is not reordered with any criteria or algorithm and it corresponds to the numbering of the mesh generator. For instance, in the Gmsh software [54], the element numbers are assigned by looping over geometrical entities of increasing dimensions (points, then curves, then surfaces, then volumes) and numbering the cells ‘as they come’ in each entity.

5.2 Second-order backward difference formula

Consider again the system of ODEs (5.2). Discretizing the temporal derivative with the BDF2 with variable time step results in

\[
\frac{1 + 2\tau}{1 + \tau} W^{n+1} - (1 + \tau) W^n + \frac{\tau^2}{1 + \tau} W^{n-1} = \Delta t^n R^{n+1},
\]

where \(\Delta t^n = t^{n+1} - t^n\), \(\tau = \frac{\Delta t^n}{\Delta t_{n+1}}\). Expression (5.13) is again a non-linear algebraic system, which has to be solved at each time iteration to find the solution at the next iteration \(t^{n+1}\) starting from the two previous time iterations \(t^n\) and \(t^{n-1}\). Linearizing the residual about time iteration \(t^n\), one obtains

\[
\frac{\Delta W_{cc}}{\Delta t^n} = c_1 \left( \frac{\partial R_{cc}}{\partial W_{cc}} \right)^n \Delta W_{cc} + \sum_{nb} \frac{\partial R_{cc}}{\partial W_{nb}} \left| \frac{\Delta W_{nb}}{\Delta t^n} \right|^n
\]

where

\[
c_1 = \frac{1 + \tau}{1 + 2\tau} \quad \text{and} \quad c_2 = \frac{\tau^2}{\Delta t^n (1 + 2\tau)}\]

with \(\Delta W = W^{n+1} - W^n\). Applying a SGS algorithm to solve the linear algebraic system (5.14) results in

\[
\left[-c_1 \frac{\partial R_{cc}}{\partial W_{cc}} \right]^n + \frac{I}{\Delta t^n} \right] \Delta W_{cc}^{m+1} = c_1 \left( R_{cc}^n + \sum_{nb} \frac{\partial R_{cc}}{\partial W_{nb}} \right)^n \Delta W_{nb}^n
\]

Inserting (5.7) into (5.15) and manipulating the resulting equation, one
obtains the following final expression:

\[
\left[ -c_1 \frac{\partial R_{cc}}{\partial W_{cc}} \right]^n \frac{I}{\Delta t^n} \Delta W_{cc}^{m+1} = c_1 R_{cc}^* + c_2 \left( W_{cc}^n - W_{cc}^{n-1} \right) - \frac{\Delta W_{cc}^*}{\Delta t^n},
\]

where \( \Delta W_{cc}^{m+1} = \Delta W_{cc}^{n+1} - \Delta W_{cc}^* = W_{cc}^{n+1,m+1} - W_{cc}^* \). This implicit solver is denoted as ‘LU-SGS + BDF2’ in the remainder of the work. Note again that, the LU-SGS algorithm acts directly on the non-linear algebraic system to be solved, i.e. the system. (5.13), which is the right hand side of expression (5.16). The initial guess for \( W_{cc}^{n+1} \) is \( W_{cc}^n \). At the first time iteration \( n = 0 \), since no solution at time level \( n = -1 \) is available, the first-order BE scheme, namely Equation (5.8), is solved. Analogous properties and remarks indicated for the LU-SGS + BE are also valid for the LU-SGS + BDF2.

### 5.3 Time step

The objective of time-accurate CFD calculations is the modeling of unsteady flow phenomena, such as vortex pairing or bluff body wakes. In unsteady calculations, a global physical time step must be used throughout the entire flow-field, regardless of the local grid distribution. Because for explicit schemes the CFL number has a limit which approximates the upper limit of numerical stability (see Appendix A), the global time step must be obtained from the maximum CFL number and the smallest grid spacing in the field. Thus, the stability of the calculations in regions of dense grid packing determines the time step for the entire flow-field, in most of the cases. For example, if a wave is crossing a discrete grid, then the time step must be less than the time for the wave to travel adjacent grid points. Therefore, when the grid point separation is reduced, the upper limit for the time step also decreases. In essence, the numerical domain of dependence must include the analytical domain of dependence in order to assure that the scheme can access the information required to form the solution [75, 99]. For the 1D linear convection equation with constant advection velocity \( a \) and on uniform grids \( \Delta \chi \), the stability limit on the time step prescribed by the CFL condition is given by

\[
\Delta t \leq \frac{\Delta \chi}{a \sigma_{C,max}}.
\]
CHAPTER 5. TIME DISCRETIZATION

For 1D problems with constant diffusive coefficient $\nu_D$ on uniform grids, a similar stability limit does exist and is defined by

$$\Delta t \leq \frac{\Delta \chi^2}{\nu_D} \sigma_{D,\text{max}}. \quad (5.18)$$

Expressions (5.17) and (5.18) can be generalized to a multi-dimensional problems on general grids with variable advection velocities and diffusive coefficients:

$$\Delta t \leq \frac{\Delta \Omega_i}{\sum_{m=1}^{N_{fac}} |\vec{a}_m \cdot \vec{1}_{n,m}| \Delta S_m} \sigma_{C,\text{max}}, \quad (5.19)$$

and

$$\Delta t \leq \frac{\Delta \Omega_i^2}{\sum_{m=1}^{N_{fac}} \nu_{D,m} \Delta S_m^2} \sigma_{D,\text{max}}, \quad (5.20)$$

where $|\vec{a}_m \cdot \vec{1}_{n,m}|$ and $\nu_{D,m}$ are respectively the maximum wave propagation velocity in the direction perpendicular to the cell face $m$ and the maximum diffusive coefficient on the same face. For the N-S equations, $\nu_{D,m}$ is the maximum of kinematic viscosity $\nu$ and thermal diffusivity $\frac{k}{\rho c_P}$. For general convection-diffusion equations, the global physical time step must then be smaller or equal to the minimum between the maximum inviscid time step (Equation (5.19)) and the viscous time step (Equation (5.20)) of each cell. However, notice that for accuracy, a smaller value for $\Delta t$ may be required. In fact, the time step must also be chosen based on the physics of the problem. This criterion is generally used for implicit time marching schemes which do not suffer from a restrictive CFL condition. Consequently, for implicit methods the time step usually does not change when the grid is changed.

In case of steady computations, as one is not interested in the transient behavior of the solution, a local time step can be chosen, whereby each cell $i$ progresses at its maximum possible time step $\Delta t_i$, without locally violating the stability conditions (5.19) and (5.20). This provides significant convergence acceleration at the cost of the transient time consistency since each cell has its own time step. This approach is known as local time stepping technique and can be applied with any explicit or implicit time marching scheme.
Chapter 6

Analysis of the non-linear LU-SGS algorithm

In this chapter, the stability and the smoothing properties of the LU-SGS + BE solver in combination with the SV and SD methods is analyzed. The main goal is to investigate the behavior of the LU-SGS + BE solver for a mesh with high-aspect ratio cells. These grids typically occur near the walls to resolve the viscous boundary layer with an economical distribution of grid points.

The properties of the LU-SGS + BE are evaluated with a Von Neumann analysis for a model 2D linear advection. The analysis is performed for second- to fourth-order SV and SD schemes using the stable SV partitions and the stable SD distributions of flux points presented in Chapter 4. For more details about the methodology of this analysis technique, the reader is referred to Appendix A. The SV results that are discussed in this chapter are published in Parsani et al. [127, 130].

6.1 Summary of the methodology

The methodology of the analysis can be summarized as follows. Consider the linear advection equation, discussed in Section 3.5, and given by

$$\frac{\partial w}{\partial t} + \vec{\nabla} \cdot (\vec{a} w) = 0.\quad (6.1)$$

Equation (6.1) is discretized in space by introducing the spatial derivatives corresponding to a spatial discretization method. A uniform grid with peri-
odic boundary conditions is considered. The grid is defined by a generating pattern, which is the smallest part from which the full grid can be reconstructed by periodically repeating the pattern in all directions. For the 2D case, the generating patterns for uniform triangular and quadrilateral meshes are shown in Figures 6.1(a) and 6.1(b) respectively. The generating pattern is completely defined by the vectors $\vec{B}_1$ and $\vec{B}_2$ and their non-dimensional form is obtained by scaling them with the length of $\vec{B}_1$, denoted by $\Delta B$: $\vec{B}_1 = \Delta B \vec{B}_1'$ and $\vec{B}_2 = \Delta B \vec{B}_2'$. If the dimensionless vector $\vec{B}_1'$ is chosen as $[1 \ 0]^T$, then the dimensionless mesh is completely defined by the two components of $\vec{B}_2'$. In 2D, the advection speed $\vec{a}$ in Equation (6.1) is defined by its amplitude $a$ and orientation angle $\psi$:

$$\vec{a} = a \begin{bmatrix} \cos \psi \\ \sin \psi \end{bmatrix}. \quad (6.2)$$

After application of the space method to (6.1) on a uniform quadrilateral or triangular mesh, the following system of ODEs is obtained:

$$\frac{dW_{i,j}}{dt} + \frac{a}{\Delta B} \left( T^{0,0} W_{i,j} + T^{-1,0} W_{i-1,j} + T^{0,-1} W_{i,j-1} + T^{+1,0} W_{i+1,j} + T^{0,+1} W_{i,j+1} \right) = 0, \quad (6.3)$$

where the five dimensionless matrices $T$ are defined by the spatial discretization method. They depend on the generating pattern, the advection speed orientation angle $\psi$ and the Riemann flux. The column vector $W_{i,j}$ contains the $N^{s,GP}$ solution variables in the generating pattern with indices $i$ and $j$ (from within one quadrilateral cell or two triangular cells).
Inserting the following Fourier wave

\[ W_{i,j}(t) = \tilde{W}(t) e^{i\tilde{k} \cdot (i\vec{B}_1' + j\vec{B}_2')} \Delta B \]

into Equation (6.3) results in

\[
\begin{align*}
\frac{d\tilde{W}}{dt} + \frac{a}{\Delta B} \left( T^{0,0} + T^{-1,0} e^{-iK\tilde{k} \cdot \vec{B}_1'} + T^{0,-1} e^{-iK\tilde{k} \cdot \vec{B}_2'} + T^{+1,0} e^{iK\tilde{k} \cdot \vec{B}_1'} + T^{0,+1} e^{iK\tilde{k} \cdot \vec{B}_2'} \right) \tilde{W} &= 0,
\end{align*}
\]

(6.5)

where \( I = \sqrt{-1} \) is the imaginary unit number and \( \tilde{k} \) is the wave vector given by

\[
\tilde{k} = k \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} = k \tilde{\vec{k}}.
\]

(6.6)

Approximating the time derivative with the BE scheme, Equation (6.7) is obtained,

\[
W^{n+1} - W^n + \sigma \left( T^{0,0} + T^{-1,0} e^{-iK\tilde{k} \cdot \vec{B}_1'} + T^{0,-1} e^{-iK\tilde{k} \cdot \vec{B}_2'} + T^{+1,0} e^{iK\tilde{k} \cdot \vec{B}_1'} + T^{0,+1} e^{iK\tilde{k} \cdot \vec{B}_2'} \right) W^{n+1} = 0,
\]

(6.7)

where \( \sigma = \frac{a \Delta t}{\Delta B} \) is the CFL number. An expression for the amplification matrix \( G_d \), defined by \( \tilde{W}^{n+1} = G_d \tilde{W}^n \), can be obtained from system (6.7):

\[
G_d = \left[ I + \sigma \left( T^{0,0} + T^{-1,0} e^{-iK\tilde{k} \cdot \vec{B}_1'} + T^{0,-1} e^{-iK\tilde{k} \cdot \vec{B}_2'} + T^{+1,0} e^{iK\tilde{k} \cdot \vec{B}_1'} + T^{0,+1} e^{iK\tilde{k} \cdot \vec{B}_2'} \right) \right]^{-1}.
\]

(6.8)

The matrix \( G_d \) represents the amplification matrix of the direct inversion method which is marked as ‘direct + BE’ in this work.

Applying a Gauss-Seidel algorithm to (6.7), the expression for the amplification matrix of the LU-SGS algorithm can be found. In this case, because of the Gauss-Seidel nature of the non-linear LU-SGS algorithm, where the latest available solution in the neighboring cells is used to update the solution in a cell, the procedure to obtain the amplification matrix for the LU-SGS depends on the generating pattern structure, through the matrix \( T^{0,0} \). In fact, the column vector \( W_{i,j} \) in (6.3) contains the \( N_{s,GP} \) solution...
variables of the generating pattern which corresponds to one quadrilateral cell or two triangular cells, as illustrated in Figures 6.1(b) and 6.1(a).

Consider a mesh made by triangular cells. After application of the Gauss-Seidel algorithm, the amplification matrix of the first forward sweep is found:

\[
G_{f,1}^{TR} = \left[ I + \sigma \left( T_{1,1}^{0,0} + T_{2,1}^{0,0} + T_{2,2}^{0,0} + T^{-1,0} e^{-iK\vec{T}_k \cdot \vec{B}_i'} + T^{0,-1} e^{-iK\vec{T}_k \cdot \vec{B}_2'} \right) \right]^{-1} \\
\times \left[ I - \sigma \left( T_{1,2}^{0,0} + T^{+1,0} e^{iK\vec{T}_k \cdot \vec{B}_i'} + T^{0,+1} e^{iK\vec{T}_k \cdot \vec{B}_2'} \right) \right],
\]

(6.9)

where \( T_{1,1}^{0,0} \) and \( T_{2,2}^{0,0} \) represent respectively the contribution to the residual of the first and second cell of the generating pattern to themselves, while \( T_{1,2}^{0,0} \) and \( T_{2,1}^{0,0} \) represent the cross contributions of both cells of the generating pattern. Matrix \( I \) is the unit or identity matrix.

The amplification matrix of the first SGS sweep is given by

\[
G_{SGS,1}^{TR} = \left[ I + \sigma \left( T_{1,1}^{0,0} + T_{1,2}^{0,0} + T_{2,2}^{0,0} + T^{+1,0} e^{iK\vec{T}_k \cdot \vec{B}_i'} + T^{0,+1} e^{iK\vec{T}_k \cdot \vec{B}_2'} \right) \right]^{-1} \\
\times \left[ I - \sigma \left( T_{2,1}^{0,0} + T^{-1,0} e^{-iK\vec{T}_k \cdot \vec{B}_i'} + T^{0,-1} e^{-iK\vec{T}_k \cdot \vec{B}_2'} \right) \right].
\]

(6.10)

From the amplification matrix \( G_{SGS,1}^{TR} \), the amplification matrix of the \( m \)-th SGS sweep may be computed using the following two-step recursive procedure:

1: Compute the amplification matrix of the \( m \)-th forward sweep

\[
G_{f,m}^{TR} = \left[ I + \sigma \left( T_{1,1}^{0,0} + T_{1,2}^{0,0} + T_{2,2}^{0,0} + T^{-1,0} e^{-iK\vec{T}_k \cdot \vec{B}_i'} + T^{0,-1} e^{-iK\vec{T}_k \cdot \vec{B}_2'} \right) \right]^{-1} \\
\times \left[ I - \sigma \left( T_{1,2}^{0,0} + T^{+1,0} e^{iK\vec{T}_k \cdot \vec{B}_i'} + T^{0,+1} e^{iK\vec{T}_k \cdot \vec{B}_2'} \right) \right] G_{SGS,m-1}^{TR}.
\]

(6.11)

2: Compute the amplification matrix of the \( m \)-th SGS sweep

\[
G_{SGS,m}^{TR} = \left[ I + \sigma \left( T_{1,1}^{0,0} + T_{1,2}^{0,0} + T_{2,2}^{0,0} + T^{+1,0} e^{iK\vec{T}_k \cdot \vec{B}_i'} + T^{0,+1} e^{iK\vec{T}_k \cdot \vec{B}_2'} \right) \right]^{-1} \\
\times \left[ I - \sigma \left( T_{2,1}^{0,0} + T^{-1,0} e^{-iK\vec{T}_k \cdot \vec{B}_i'} + T^{0,-1} e^{-iK\vec{T}_k \cdot \vec{B}_2'} \right) \right] G_{f,m}^{TR}.
\]

(6.12)
6.1. SUMMARY OF THE METHODOLOGY

With quadrilateral cells, the amplification matrices for the first forward sweep and the first SGS sweep are given by Equations (6.13) and (6.14) respectively.

\[
G_{QD,f,1}^{QD} = \left[ I + \sigma \left( T_{0,0}^{0,0} + T_{1,1}^{-1,0} e^{-IKI_k \cdot \vec{B}_1'} + T_{0,0}^{0,-1} e^{-IKI_k \cdot \vec{B}_2'} \right) \right]^{-1} \\
\times \left[ I - \sigma \left( T_{1,1}^{+1,0} e^{IKI_k \cdot \vec{B}_1'} + T_{0,0}^{0,+1} e^{IKI_k \cdot \vec{B}_2'} \right) \right] \quad (6.13)
\]

\[
G_{QD,SGS,1}^{QD} = \left[ I + \sigma \left( T_{0,0}^{0,0} + T_{1,1}^{+1,0} e^{IKI_k \cdot \vec{B}_1'} + T_{0,0}^{0,+1} e^{IKI_k \cdot \vec{B}_2'} \right) \right]^{-1} \\
\times \left[ I - \sigma \left( T_{1,1}^{-1,0} e^{-IKI_k \cdot \vec{B}_1'} + T_{0,0}^{0,-1} e^{-IKI_k \cdot \vec{B}_2'} \right) \right] \quad (6.14)
\]

In this case, the matrix \( T_{0,0}^{0,0} \) contains only the contributions of the \( N_{s,GP} \) solution variables of the generating pattern which corresponds to one quadrilateral cell. From the amplification matrix \( G_{QD,SGS,1}^{QD} \), the amplification matrix of \( m \)-th SGS sweep may be computed using again the following two-step recursive procedure:

1: Compute the amplification matrix of the \( m \)-th forward sweep

\[
G_{QD,f,m}^{QD} = \left[ I + \sigma \left( T_{1,1}^{0,0} + T_{1,1}^{-1,0} e^{-IKI_k \cdot \vec{B}_1'} + T_{0,0}^{0,-1} e^{-IKI_k \cdot \vec{B}_2'} \right) \right]^{-1} \\
\times \left[ I - \sigma \left( T_{1,1}^{+1,0} e^{IKI_k \cdot \vec{B}_1'} + T_{0,0}^{0,+1} e^{IKI_k \cdot \vec{B}_2'} \right) \right] G_{QD,SGS,m-1}^{QD} \quad (6.15)
\]

2: Compute the amplification matrix of the \( m \)-th SGS sweep

\[
G_{QD,SGS,m}^{QD} = \left[ I + \sigma \left( T_{0,0}^{0,0} + T_{1,1}^{+1,0} e^{IKI_k \cdot \vec{B}_1'} + T_{0,0}^{0,+1} e^{IKI_k \cdot \vec{B}_2'} \right) \right]^{-1} \\
\times \left[ I - \sigma \left( T_{1,1}^{-1,0} e^{-IKI_k \cdot \vec{B}_1'} + T_{0,0}^{0,-1} e^{-IKI_k \cdot \vec{B}_2'} \right) \right] G_{QD,f,m}^{QD} \quad (6.16)
\]

Let \( \lambda_m (m = 1, 2, \ldots, N_{s,GP}) \) be an eigenvalue of the general amplification matrix \( G = G(K, \theta, \psi) \) and \( \lambda = \lambda(G(K, \theta, \psi)) \) represents the eigenvalue spectrum of \( G \). Then \( g = g(K, \theta, \psi) \equiv \max |\lambda(G)| \) is the amplification factor for a given \( (K, \theta, \psi) \). In order for the global discretization to be stable, \( g \leq 1 \) should be satisfied i.e. \( \lambda(G) \) lies inside the unit circle of the complex plane (stability boundary) for all \( K, \theta \) and \( \psi \). The range of \( K \) is one period of the Equation (6.7) and it is marked \( P_K \) in this work. For a fixed shape of the generating pattern, \( P_K \) is a function of \( \theta \), i.e. \( P_K = P_K(\theta) \). The expression from which \( P_K \) can be computed, is obtained by substituting \( \vec{B}_1', \vec{B}_2' \) and \( \theta \) in the exponential terms of Equation (6.7) and using the Euler's...
formula, which gives a relation between trigonometric functions and complex numbers. Following this reasoning, a trigonometric function $f$, which has exactly the same period as Equation (6.7) is obtained. In general, it is not possible to find a closed formulation for $P_K = P_K(\theta)$. Therefore, the period $P_K$ is computed taking the inverse fast Fourier transformation of the function $f$.

If the scheme is used as a smoother for multigrid, then it must have good damping of high-frequency error components, i.e. it should cluster the amplification matrix eigenvalues corresponding to the high-frequency modes towards the origin. Therefore, for high-frequency modes $q \ll 1$ for all $\theta$ and $\psi$ should be satisfied. In addition, we desire that the CFL number be sufficiently large to produce significant reduction (if not elimination) of the convergence slow-down effects that are associated with high-aspect ratio mesh cells. A large CFL number also facilitates the expulsion of error components. At the same time the capability for large CFL numbers must not compromise the high-frequency damping property of the scheme.

In the next sections, the damping properties of the LU-SGS + BE method in combination with the SV and SD schemes will be analyzed, for a high CFL number and a mesh with high aspect ratio cells. The latter choice accounts for the effects of the geometrical stiffness imposed by the Navier-Stokes grids where high-aspect ratios occur near walls. However, since the performance of the direct inversion method is better than that of any approximate method, the direct inversion method is examined as a baseline for comparison with the LU-SGS algorithm. An upwind Riemann flux is employed and the direction of the wave propagation velocity $\psi$ is set to $-\frac{\pi}{6}$ ($a_{x_1} > 0$, $a_{x_2} < 0$). A negative $\chi_2$ velocity component is chosen to avoid having the sweep directions of the LU-SGS method aligned with the flow direction. In fact, if the flow direction corresponds to the sweep directions, the LU-SGS becomes a direct solver, e.g. in case of a purely upwind scheme. The smoothing properties of the direct + BE and the LU-SGS + BE methods are shown for three values of the solution orientation (or Fourier wave angle) $\theta$, i.e. $-\frac{\pi}{6}$, 0 and $\frac{\pi}{3}$.

### 6.2 SV method for triangular cells

In this section the LU-SGS + BE solver in combination with second- to fourth-order SV schemes is analyzed. In order to explain the procedure used to construct the amplification factor as a function of the wave number,
first a grid with isotropic cells, and a small CFL number are considered. This allows to have better understanding of the necessary steps to carry out the task.

### 6.2.1 Second-order SV method

Consider a mesh built from equilateral triangles. Such a mesh is obtained with the following choice for the dimensionless vectors $\mathbf{B}_1'$ and $\mathbf{B}_2'$,

\[
\mathbf{B}_1' = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \\
\mathbf{B}_2' = \begin{pmatrix} \frac{7}{2} \\ \sqrt{3} \end{pmatrix},
\]  

and leads to a mesh with isotropic cells, i.e. the aspect ratio, $AR = \frac{|\mathbf{B}_1'|}{|\mathbf{B}_2'|} = 1$.

Figure 6.2 shows the eigenvalue spectrum of the amplification matrix of the LU-SGS + BE method and the direct + BE method on this mesh, for CFL= 1, varying the direction $\psi$ of the wave propagation velocity $\vec{a}$, the wave number $K$, and the solution orientation defined by the angle $\theta$. For the LU-SGS + BE one, two and three SGS sweeps were used. It is seen that the global discretization is stable for all $K$, $\theta$ and $\psi$. Figure 6.2 shows that with an increasing number of SGS sweeps, the eigenvalue spectrum of the LU-SGS + BE method approaches that of the direct + BE method.

The aim is to plot the amplification factor $g$ as a function of the wave number $K$. Hence, the following paragraphs first present the analysis of the smoothing properties done for the direct + BE solver and describe the methodology used to plot correctly the amplification factor $g$. First, consider a solution with an orientation $\theta = \psi = -\frac{\pi}{6}$. For this choice, Equation (6.7) is periodic in $K$ with a period equal to $\frac{4}{\sqrt{3}\pi}$. However, Equation (6.7) has $N_{s,GP}$ eigenvalues for each $K$, and each eigenvalues corresponds to a wave number $K + l\frac{4}{\sqrt{3}\pi}$, with $l$ an integer number. Consequently, to get the correct damping properties of the solver, each mode should be shifted by a multiple of $\frac{4}{\sqrt{3}\pi}$ along the wave number axis. The actual wave number $K + l\frac{4}{\sqrt{3}\pi}$ to which an eigenvalue $\lambda_m$ ($m = 1, 2, \ldots, N_{s,GP}$) corresponds has to be determined by examining the spatial shape of the eigenmodes $\mathbf{V}_m e^{iK\mathbf{k} \cdot (i\mathbf{B}_1' + j\mathbf{B}_2')}$, where $\mathbf{V}_m$ are the eigenvectors of the amplification matrix. The shape of these eigenmodes are shown in Figure 6.3 for $K = l\frac{\pi}{\sqrt{3}}$ with $l=0, \ldots, 5$.

In Figure 6.4 the eigenvalue spectrum $\lambda(G_d)$ is plotted as a function of the wave number $K$, for $\psi = \theta = -\frac{\pi}{6}$ and CFL= 1. For small wave numbers
CHAPTER 6. ANALYSIS OF THE NON-LINEAR LU-SGS ALGORITHM

(a) 1 SGS sweep.

(b) 2 SGS sweeps.

(c) 3 SGS sweeps.

(d) direct + BE

Figure 6.2: Effect of the SGS sweeps on the eigenvalue spectrum of the LU-SGS + BE scheme. Second-order \((p = 1)\) SV method, \(\psi \in [0, 2\pi], \theta \in [0, 2\pi], K \in [0, P_K(\theta)],\) \(AR = 1, CFL=1.\)

\(K\), the shape of the eigenmodes reveals the correct wave number to which an eigenvalue corresponds. In fact, three curves can readily be selected by examining the spatial shape of the eigenvector (see Figure 6.3), namely the curves marked by the plus (+), the square (□) and the circle (○) signs. The other curves, marked by the (×) sign, correspond to eigenvalues which belong to a higher wave number range of \(K\) compared to the eigenvalues of the selected curves. From Figure 6.3 it is clear that the amplitude of the discontinuities in the numerical solution at the cell interfaces grow with increasing wave number \(K\) which makes it difficult to analyze the eigenvectors and to select the actual wave number \(K + \frac{l}{\sqrt{3}} \pi\) to which an eigenvalue corresponds. Therefore, for a high-wave number, the effort of
6.2. SV METHOD FOR TRIANGULAR CELLS

Figure 6.3: Eigenmodes of the direct + BE scheme. Second-order \((p = 1)\) SV method, \(\theta = \psi = -\frac{\pi}{6}, \ AR = 1, \ CFL = 1, \ K = l \frac{\pi}{\sqrt{3}}, \ l = 0, \ldots, 5.\)

associating each eigenvalue with the corresponding wave number \(K\) has not been performed. However, the eigenvalues which correspond to a high-
wave number $K$, thus the eigenvalues marked by the sign ($\times$), are well damped as observed in Figure 6.4.

Therefore, shifting these curves in the appropriate way, leads to plots as in Figure 6.5. Because of the symmetry ($Re[\lambda(G(K))] = Re[\lambda(G(-K))]$ and $Im[\lambda(G(K))] = -Im[\lambda(G(-K))]$), the curves are only shown for positive wave numbers $K \in [0, \frac{3}{2} P_K]$.

In this case, the direct solver is a physically accurate scheme because $CFL = 1$. Consequently, its amplification factor should closely follow the exact amplification factor relation given by Equation (6.18),

$$G_{exact} = \frac{w(t + \delta t)}{w(t)} = e^{-Ika\Delta t}. \quad (6.18)$$

This expression can be approximated by a Taylor series as

$$G_{exact} \approx 1 - IKa\Delta t - \frac{K^2 a^2 \Delta t^2}{2} + o(K^3 \Delta t^3), \quad (6.19)$$

for wave numbers $K$ close to zero ($K\Delta t \ll 1$). In Figures 6.5(a) and 6.5(b), the real and the imaginary parts of the amplification factor of the direct solver for $K\Delta t$ close to zero are plotted along with their counterparts in Equation (6.19). It is seen that the direct + BE method follows closely the

![Figure 6.4: Real and imaginary parts of the eigenvalue spectrum of the amplification matrix for the direct + BE scheme. Second-order ($p = 1$) SV method, $\theta = \psi = -\frac{\pi}{6}$, $AR = 1$, CFL = 1.](image)
6.2. SV METHOD FOR TRIANGULAR CELLS

Figure 6.5: Real and imaginary parts of the shifted eigenvalues of the direct + BE scheme and corresponding amplification factor shape. Second-order \((p = 1)\) SV method, \(\theta = \psi = -\frac{\pi}{6}\), \(AR = 1\), CFL = 1.

Theoretical results for \(K \Delta t \ll 1\). In Figure 6.5(c) the amplification factor \(g\) of the direct + BE inversion method is plotted as a function of the wave number \(K \in [0, \frac{3}{2} P_K]\), for \(\theta = \psi = -\frac{\pi}{6}\). It is seen that the direct + BE solver has good damping properties for high-frequency solution components.

The procedure described above may be applied to the amplification matrix of the LU-SGS + BE scheme. In Figure 6.6 the eigenvalue spectrum of the iterative solver is plotted as a function of the wave number \(K\), for \(\psi = \theta = -\frac{\pi}{6}\), CFL = 1 and one SGS sweep. In this figure, three curves
marked with the plus sign (+), the square (□) and the circle (○) are shown (cf. Figure 6.4). Shifting these curves in the appropriate way the amplification factor curve of Figure 6.7 for $K \in [0, \frac{3}{2} P_K]$ is found. In the same plot, the amplification factor for two and three sweeps is also shown. It is seen

Figure 6.7: Effect of the SGS sweeps on the amplification factor of the LU-SGS + BE scheme. Second-order ($p = 1$) SV method, $\theta = \psi = -\frac{\pi}{6}$, $AR = 1$, $CFL = 1$. 

88
that the discontinuity, obtained with one SGS sweep, disappears when two SGS sweeps are employed. For three SGS sweeps the amplification factor of the LU-SGS + BE method is indistinguishable from that of the direct + BE inversion method. This explains why for the analysis of a mesh with $AR=1$ and $CFL=1$, a maximum number of three SGS sweeps was used.

Consider the same equilateral triangular mesh but with a CFL number of $10^6$. The eigenvalue spectrum of the amplification matrix of the direct + BE and the LU-SGS + BE methods is plotted in Figure 6.8, for this case, varying again the direction $\psi$ of the wave propagation velocity $\vec{a}$, the wave number $K$ and the angle $\theta$. For the LU-SGS + BE method, one, ten and

![Figure 6.8](image_url)

Figure 6.8: Effect of the SGS sweeps on the eigenvalue spectrum of the LU-SGS + BE scheme. Second-order ($p = 1$) SV scheme, $\psi \in [0, 2\pi]$, $\theta \in [0, 2\pi]$, $K \in [0, P_K(\theta)]$, $AR = 1$, CFL $= 10^6$. 

89
one hundred SGS sweeps are employed. It is seen that for CFL = $10^6$, the LU-SGS + BE method is still stable for all $K$, $\theta$ and $\psi$. However, using too few SGS sweeps (1-3 sweeps), the scheme exhibits poor damping behavior for some values of the angle $\theta$ and a range of frequencies $K$.

In Figures 6.9(a), 6.9(b) and 6.9(c) the amplification factor for $\theta = -\frac{\pi}{6}$, $\theta = 0$ and $\theta = \frac{\pi}{3}$ is shown for CFL= $10^6$. In these figures, the amplification factor of the direct solver is also plotted for comparison. Figure 6.9(b) shows that for $\theta = 0$ the amplification factor exhibits a discontinuity.
ity for a wave number equal to half the period of Equation (6.7), i.e. for 
\( K = 2\pi \). In fact, the amplification factor starts from one for 
\( K = 0 \), and it decreases very fast to zero for \( K > 0 \). However, when 
\( K = 2\pi \), it jumps to a value which is again close to \((0.9 - 1)\) and then it rapidly decreases again for high-wave numbers. Notice that the discontinuity occurs at a low-frequency wave number, i.e. at a frequency which is \( \frac{1}{6} \) of the whole wave number covered by the spatial scheme. By increasing the number of SGS sweeps, the amplitude of the discontinuity decreases but for one hundred SGS sweeps the curve is still discontinuous and it differs from that of the direct + BE method.

Figures 6.9(a) and 6.9(c) show that, for \( \theta = -\frac{\pi}{6} \) and \( \theta = \frac{\pi}{3} \), the high-frequency components are well damped and the amplification factor does not exhibit jumps as for \( \theta = 0 \). Moreover, for these two values of \( \theta \), the amplification factor also shows a discontinuity which disappears when five or more SGS sweeps are employed.

In order to study the damping properties of the LU-SGS + BE scheme 
when the geometrical stiffness imposed by the Navier-Stokes grids occurs 
near walls, consider a mesh obtained for the following choice of the dimen-
sionless vectors \( B'_1 \) and \( B'_2 \),

\[
B'_1 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad B'_2 = \begin{pmatrix} 0 \\ \frac{1}{100} \end{pmatrix}.
\]

(6.20)

This mesh has anisotropic cells with \( AR = 100 \). As for a mesh with \( AR = 1 \),
in Figures 6.10(a), 6.10(b), 6.10(c) the amplification factor of both direct 
and LU-SGS methods is plotted as a function of the wave number \( K \), for 
\( \theta = -\frac{\pi}{6} \), \( \theta = 0 \) and \( \theta = \frac{\pi}{3} \). In these cases, the period \( P_K \) for the selected 
values of the solution orientation is respectively, \( \frac{2\pi}{\sin \frac{\pi}{6}} \), \( 2\pi \) and \( \frac{2\pi}{\sin \frac{\pi}{3}} \).

It is seen that, when few SGS sweeps (3 - 5) are employed, the damp-
ing properties of the LU-SGS + BE solver for \( \theta = -\frac{\pi}{6} \) and \( \theta = \frac{\pi}{3} \) are similar 
to that of the direct + BE method, Figures 6.10(a) and 6.10(c). For \( \theta = 0 \) 
(Figure 6.10(b)) the behavior is different. In fact, the low-frequency com-
ponents are not damped at all and the amplification factor for a wave number 
range \([0, \pi]\) is close to one. However, for \( K > \pi \) it jumps to a value close to 
zero and the high-frequency error components are still well damped.
6.2.2 Third-order SV method

The analysis for third-order SV method was performed using the mesh with $AR = 100$ and a CFL number of $= 10^6$. For the LU-SGS + BE solver one, three, five, ten and one hundred SGS sweeps were employed. In Figure 6.11 the amplification factor is plotted as a function of the wave number, respectively for $\theta = -\frac{\pi}{6}, \theta = 0, \theta = \frac{\pi}{3}$.

For the third-order SV scheme, the damping properties of the LU-SGS +
6.2. SV METHOD FOR TRIANGULAR CELLS

Figure 6.11: Effect of the SGS sweeps on the amplification factor of the LU-SGS + BE scheme, for different Fourier wave angles $\theta$. Third-order ($p = 2$) SV method, $AR = 100$, CFL = $10^6$.

BE solver for $\theta = -\frac{\pi}{6}$ and $\theta = \frac{\pi}{3}$ are close to those of the direct + BE method when few SGS sweeps are employed, Figures 6.11(a) and 6.11(c). However, when $\theta = 0$, the low-frequency error components are not damped and the amplification factor for a wave number range $[0, \pi)$ is close to one. Subsequently, for $K > \pi$ the amplification factor jumps immediately to a value close to zero and the high-frequency error components are nicely damped, see Figure 6.11(b). This behavior has also been observed for second-order SV scheme, as discussed in the previous section. Notice that the amplification factor shows a discontinuity for $\theta = -\frac{\pi}{6}$ and $\theta = \frac{\pi}{3}$ at high-frequency,
CHAPTER 6. ANALYSIS OF THE NON-LINEAR LU-SGS ALGORITHM

when one or three SGS sweeps are employed. This discontinuity disappears by increasing the number of SGS sweeps.

6.2.3 Fourth-order SV method

The analysis of the fourth-order SV method was also performed on a mesh with $AR = 100$ and for a CFL number of $10^6$. In Figure 6.12 the amplification factor is plotted as a function of the wave number for $\theta = -\frac{\pi}{6}$,

![Figure 6.12](image)

(a) $\theta = \psi = -\frac{\pi}{6}$.

(b) $\psi = -\frac{\pi}{6}, \theta = 0$.

(c) $\psi = -\frac{\pi}{6}, \theta = \frac{\pi}{3}$.

Figure 6.12: Effect of the SGS sweeps on the amplification factor of the LU-SGS + BE scheme, for different Fourier wave angles $\theta$. Fourth-order ($p = 3$) SV method, $AR = 100$, CFL = $10^6$. 

94
6.3. SD METHOD FOR QUADRILATERAL CELLS

\[ \theta = 0, \theta = \frac{\pi}{3} \]. One, three, five, ten and hundred SGS sweeps were used for the LU-SGS + BE solver. It is seen that also for the fourth-order SV scheme, the amplification factor of the LU-SGS + BE scheme gets closer to that of the direct + BE solver for \( \theta = -\frac{\pi}{6} \) and \( \theta = \frac{\pi}{3} \) when more SGS sweeps are used. However, for \( \theta = 0 \), the same behavior observed as for the third-order SV scheme is found. In fact, the amplification factor for a wave number range \([0, \pi]\) is close to one and for \( K > \pi \) it jumps immediately to a value close to zero, showing that the high-frequencies error components are nicely damped.

6.3 SD method for quadrilateral cells

In this section the main results obtained by investigating the smoothing properties of the LU-SGS + BE solver in combination with second- to fourth-order SD schemes are shown. The aim is to show that the LU-SGS + BE solver is also a good smoother for the SD method. To construct the amplification factor \( g \) as a function of the dimensionless wave number \( K \), the procedure explained for the SV schemes is used.

A mesh with quadrilateral cells and aspect ratio \( AR \) of 100 was considered. This mesh was obtained by choosing the dimensionless vectors \( B'_1 \) and \( B'_2 \) as shown in (6.20). For the sake of consistency, the CFL number was set to \( 10^6 \) and one, three, five, ten and hundred SGS sweeps were used for the LU-SGS + BE solver.

6.3.1 Second-order SD method

The amplification factor of the LU-SGS + BE for the second-order SD scheme for quadrilateral cells is shown in Figure 6.13, for \( \theta = -\frac{\pi}{6}, \theta = 0, \theta = \frac{\pi}{3} \). In this figure, the amplification factor of the direct solver is also plotted for comparison. It is seen that the iterative solver, in combination with the SD scheme, is stable, i.e. \( g \leq 1 \), and its amplification factor gets closer to that of the direct + BE solver by increasing the number of SGS sweeps. In addition, Figure 6.13(b) shows that for \( \theta = 0 \), the low-frequency error components are weakly damped for a wave number range \([0, \pi]\). For high-wave numbers the amplification factor is close to zero, showing that the high-frequency error components are nicely damped.

Upon comparison of the amplification factor for the second-order SV scheme and the amplification factor for the second-order SD scheme, it is seen that the LU-SGS + BE solver gives similar results for both spatial operators.
CHAPTER 6. ANALYSIS OF THE NON-LINEAR LU-SGS ALGORITHM

Figure 6.13: Effect of the SGS sweeps on the amplification factor of the LU-SGS + BE scheme, for different Fourier wave angles $\theta$. Second-order ($p = 1$) SD method, $AR = 100$, CFL = $10^6$.

Note that when hundred SGS sweeps are used, the amplification is almost indistinguishable from that of the direct + BE inversion method.

6.3.2 Third-order SD method

Figure 6.14 shows the amplification factor of the LU-SGS + BE for third-order SD scheme. The plots in this figure show once more that the amplification factor of the iterative solver gets closer to that of the direct + BE
6.3. SD METHOD FOR QUADRILATERAL CELLS

solver for $\theta = -\frac{\pi}{6}$ and $\theta = \frac{\pi}{3}$ by increasing the number of SGS sweeps. For $\theta = 0$, the same behavior as for the second-order SD scheme is observed.

As for the third-order SV case, the amplification factor shows a discontinuity for $\theta = -\frac{\pi}{6}$ and $\theta = \frac{\pi}{3}$ at high-frequency, when one or three SGS sweeps are employed. The discontinuity disappears with an increasing number of SGS sweeps.

![Figure 6.14](image_url)

(a) $\theta = \psi = -\frac{\pi}{6}$.
(b) $\psi = -\frac{\pi}{6}, \theta = 0$.
(c) $\psi = -\frac{\pi}{6}, \theta = \frac{\pi}{3}$.

Figure 6.14: Effect of the SGS sweeps on the amplification factor of the LU-SGS + BE scheme, for different Fourier wave angles $\theta$. Third-order ($p = 2$) SD method, $AR = 100$, $CFL = 10^6$. 

97
6.3.3 Fourth-order SD method

The amplification factor of the LU-SGS + BE for the fourth-order SD scheme for quadrilateral cells is shown in Figure 6.15, for $\theta = -\frac{\pi}{6}$, $\theta = 0$, $\theta = \frac{\pi}{3}$. It can be seen that few SGS sweeps are in general sufficient to damp high-frequency error components and get an amplification factor fairly close to that of the direct solver, for $\theta = -\frac{\pi}{6}$ and $\theta = \frac{\pi}{3}$. Similar results as for the

![Figure 6.15: Effect of the SGS sweeps on the amplification factor of the LU-SGS + BE scheme, for different Fourier wave angles $\theta$. Fourth-order ($p = 3$) SD method, $AR = 100$, CFL = $10^6$.](image)

foregoing analysis are found for $\theta = 0$. Indeed, the low-frequency error components are weakly damped for a wave number range $[0, \pi)$ and the
6.4 Remarks

The analysis has demonstrated that the LU-SGS + BE scheme is always stable for any choice of the convective velocity direction $\psi$ and the solution orientation $\theta$ for the second- to fourth-order SV and SD schemes. Furthermore, the analysis has shown that the smoothing properties of the implicit solver depend strongly on the orientation of the solution defined by the angle $\theta$. It has been shown that by increasing the number of the SGS sweeps, the damping behavior of the LU-SGS + BE scheme gets closer to the damping properties of a direct solver + BE scheme.

The analysis was performed for a CFL of $10^6$ and on a mesh with an aspect ratio of hundred. This choice has enabled to take into account the effects of the geometrical stiffness imposed by the Navier-Stokes grids where high-aspect ratios occur near walls. It has been found that 5 – 6 sweeps are in general sufficient to get a good damping of the high-frequency error components. Moreover, it has been shown that the amplification factor may have a discontinuity for a certain wave number. This depends on the direction of the harmonic wave solution and the number of SGS sweeps. Nevertheless, the high-frequency error components are always well damped. Besides, for a specific direction of the harmonic wave solution ($\theta = 0$) of the 2D linear convection equation, the implicit iterative solver has some difficulties to damp low-frequency error components for a wave number range equal to half of the period of Equation (6.7). For wave numbers higher than half of the period the amplification factor is close to zero.
CHAPTER 6. ANALYSIS OF THE NON-LINEAR LU-SGS ALGORITHM
Chapter 7

Application I: spectral volume method

In this chapter, the main results achieved by applying the spectral volume method and the non-linear LU-SGS algorithm with backward Euler scheme to problems governed by the 2D steady compressible Navier-Stokes equations are presented. The coupling between the SV method and the non-linear LU-SGS algorithm has been limited to this type of flows because, as discussed in Section 4.1.4, no stable high-order 3D SV partition has been found until now. Therefore, since the goal of this PhD research has been the development of an efficient high-order N-S/LES solver, the application of the implicit time marching scheme for 2D and 3D unsteady laminar and turbulent flow simulations has been done for the SD method, which was proven to be stable for any order of accuracy and multi-dimensional problems by several researchers. Examples of flow simulations with the implicit SD solver will be shown in the next chapter.

For the SV method, the implementation of the LU-SGS + BE scheme was done in the 2D-SV-TRI code, which is a C++ code developed at the Vrije Universiteit Brussel, Department of Mechanical Engineering, Fluid Mechanics and Thermodynamics Research Group. In this code, the 2D SV method can also be combined with explicit Runge-Kutta (E-RK) time marching schemes and for steady-state flow simulations, the (pseudo) time integration can be accelerated by a full $p$-multigrid strategy, as discussed in Appendix B.

The grids for all the SV test cases were generated with Gmsh [54]. The cal-
calculations were performed on a Linux workstation with an Intel T2500 Core Duo (2.0 GHz) processor™. Wherever possible, direct validation against experimental data and/or reference solutions available in literature are made. The results presented in the following sections are published in Parsani et al. [127, 130].

7.1 Two-dimensional laminar steady flow simulations

The LU-SGS + BE scheme in combination with a full multigrid (FMG) V-cycle \( p \)-multigrid algorithm was used to compute the 2D steady laminar flow past a circular cylinder and over a NACA0012 airfoil, and the steady laminar flow in a channel with a backward-facing step. In order to assess the convergence properties of the LU-SGS + BE scheme, the performance of the latter scheme is compared with that of a family of E-RK smoothers available in the 2D-SV-TRI code. The E-RK schemes compute the solution at the new time \( t^{n+1} \), i.e. \( \mathbf{W}^{n+1} \), starting from the solution at time \( t^n \), i.e. \( \mathbf{W}^n \), using an explicit \( N_{RK} \)-stage procedure. In this thesis, E-RK schemes of the following form are considered [138]:

\[
\begin{align*}
W_{i,j}^0 &= W_{i,j}^n \\
W_{i,j}^m &= C_1^m W_{i,j}^0 + C_2^m W_{i,j}^{m-1} + C_3^m \Delta t \Omega_{i,j} R_{i,j}^{m-1} \\
W_{i,j}^{n+1} &= W_{i,j}^{N_{RK}},
\end{align*}
\]

where the coefficients \( C_1^m, C_2^m, C_3^m \) are listed in Table 7.1. These E-RK schemes were introduced in Ramboer et al. [138] in order to minimize the total error arising from the spatial and temporal discretizations. Because of this, they are also denoted ‘optimized E-RK schemes’ in the reminder of this work.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>( C_1^m )</th>
<th>( C_2^m )</th>
<th>( C_3^1 )</th>
<th>( C_3^2 )</th>
<th>( C_3^3 )</th>
<th>( C_3^4 )</th>
<th>( C_3^5 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>Opt. E-RK2</td>
<td>1</td>
<td>0</td>
<td>( \frac{1}{4} )</td>
<td>1</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Opt. E-RK3</td>
<td>1</td>
<td>0</td>
<td>( \frac{1}{5} )</td>
<td>( \frac{1}{2} )</td>
<td>1</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Opt. E-RK5</td>
<td>1 (if ( m = 1 ))</td>
<td>0 (if ( m = 1 ))</td>
<td>85 (if ( m = 1 ))</td>
<td>10 (if ( m = 1 ))</td>
<td>9 (if ( m = 1 ))</td>
<td>1 (if ( m = 1 ))</td>
<td>132 (if ( m = 1 ))</td>
</tr>
<tr>
<td></td>
<td>( 1 - C_3^m ) (if ( m \neq 1 ))</td>
<td>( C_3^m ) (if ( m \neq 1 ))</td>
<td>( \frac{1300}{1300} )</td>
<td>( \frac{9}{50} )</td>
<td>( \frac{1}{4} )</td>
<td>( \frac{300}{300} )</td>
<td></td>
</tr>
</tbody>
</table>
7.1. TWO-DIMENSIONAL LAMINAR STEADY FLOW SIMULATIONS

The FMG V-cycle $p$-multigrid strategies adopted here are depicted in Figure 7.1, for both E-RK and LU-SGS + BE smoothers. The numbers of smoothing sweeps were empirically determined in order to minimize the computer time (CPU-time) needed to perform the simulations. Note that these numbers are independent of the test case.

For the LU-SGS + BE scheme, the non-linear system (5.8) was solved with multiple cell-wise symmetric Gauss-Seidel (SGS) sweeps with a prescribed tolerance of $10^{-6}$ on the change of the $L_2$ norm of the solution variation $\Delta W_{cc}^{m+1}$ and/or a maximum number of 6 SGS sweeps. For all the calculations, a high CFL number, varying between $10^4$ and $10^6$, was used. The former choices allows strong damping of high-frequency error components and gives good properties in terms of CPU-time and convergence rate for the present computations. Notice that, in Chapter 6, from the Von Neumann analysis of the 2D linear advection equation, it was found that a maximum number of $5 - 6$ SGS sweeps are in general sufficient to get good damping of high-frequency error components for a CFL number of $10^6$.

All the test cases were solved with a local time stepping technique, as described in Section 5.3. A Roe FDS flux as a Riemann solver and the LSV approach for the diffusive terms were used. For convergence, the residual of the continuity equation was monitored. Here, the residuals are normalized by the corresponding residuals of the first iteration and they are shown in base ten logarithmic scale.
CHAPTER 7. APPLICATION I: SPECTRAL VOLUME METHOD

7.1.1 Flow over a circular cylinder

The 2D steady compressible viscous flow past a circular cylinder is considered in this section. The computational domain is shown in Figure 7.2 where the incoming uniform flow is from left to right, i.e. in the $\chi_1$ direction. The simulation is conducted at Reynolds number $Re$ of 40 and free-stream Mach number $M_\infty$ of 0.15. The Reynolds number is based on the free-stream velocity module $|\vec{u}_\infty|$ and the diameter of the cylinder $D$. The Prandtl number $Pr$ is set to 0.72, which is the standard value for air. At this conditions, the flow is laminar and steady, with a recirculation zone behind the cylinder.

At the cylinder wall, which is assumed to be adiabatic, the no-slip boundary condition discussed in Section 3.3.4 is imposed. Maximum second-order ($p = 1$) SV results are given because of the curved boundaries. A first-order interpolation is used for the boundary shape in the 2D-SV-TRI code, but high-order schemes would require a more accurate interpolation, especially on the relatively coarse grids that are being used in combination with high-order polynomial representations of the solution; see for instance Wang and Liu [188]. At the far field boundary, which is sufficiently far from the cylinder, the flow is prescribed to be uniform. This corresponds to a Dirichlet boundary condition, as discussed in Section 3.3.1.

First, a medium grid with 3,744 triangular cells with a maximum aspect-ratio $AR_{max}$ of 2 was used. The mesh was generated in such a way that the cell aspect ratio close to the cylinder wall could be easily controlled, as illustrated in Figure 7.3.
7.1. TWO-DIMENSIONAL LAMINAR STEADY FLOW SIMULATIONS

Figure 7.3: Detail of the grid for the steady laminar flow over a circular cylinder. 3,744 triangular cells with $AR_{max} = 2$.

In the first part of Table 7.2, the CFL numbers for both E-RK and LU-SGS + BE schemes are listed. For the E-RK scheme two CFL numbers are indicated; the first one is the convective CFL number $\sigma_C$ while the second one is the viscous CFL number $\sigma_D$. According to the local time stepping technique, two local time steps are then computed and the solution at the next time level is calculated using the minimum between these two values. In this table, the exponent $n$ denotes the time iteration index. For the LU-SGS + BE scheme both maximum convective and viscous CFL numbers were set to $10^6$, for each $p$-multigrid level.

Table 7.2: CFL number for two-level V-cycles $p$-multigrid on two different grids, for the steady laminar flow over a circular cylinder.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>3,744 cells, $AR_{max} = 2$</th>
<th>5,440 cells, $AR_{max} = 162$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>SV $p = 0$</td>
<td>SV $p = 1$</td>
</tr>
<tr>
<td>Opt. E-RK</td>
<td>7, 0.5, 4, 0.3</td>
<td>1.5, 0.01, 0.35, 0.005</td>
</tr>
<tr>
<td>LU-SGS + BE</td>
<td>$10^n$</td>
<td>$10^n$</td>
</tr>
</tbody>
</table>

The Mach number contour and the stream traces obtained with the second-order SV method and the LU-SGS + BE scheme are shown in Figure 7.4. It can be seen that, the flow field around the cylinder is well resolved and the recirculation zone behind the cylinder is clearly visible.

In Figure 7.5, the convergence histories of the E-RK and the LU-SGS + BE schemes are compared. The computations were stopped when the $L_2$ norm of the residuals was reduced 12 orders of magnitude. In Table 7.3, the
CHAPTER 7. APPLICATION I: SPECTRAL VOLUME METHOD

Figure 7.4: Mach number contour and stream traces, for the steady laminar flow over a circular cylinder, obtained with second-order \((p = 1)\) SV method and the LU-SGS + BE scheme. Grid with 3,744 triangular cells and \(AR_{\text{max}} = 2\). \(\Delta M = 0.01\).

number of V-cycles for each multigrid level (VMG1 and VMG2), the total number of V-cycles (VMG) and the total CPU-time are listed. For this test case, once the VMG2 cycle is used and the second-order accurate solution is computed, the smoothing sweeps at \(p = 0\) takes 69\% of the total CPU time listed in Table 7.3. The super-script \(*\) indicates that the residuals is not yet converged. In fact, after 10,000\* V-cycles the E-RK scheme reduced

Figure 7.5: Convergence histories of the optimized E-RK scheme and the LU-SGS + BE scheme for the steady laminar flow over a circular cylinder. Second-order \((p = 1)\) SV method, grid with 3,744 triangular cells and \(AR_{\text{max}} = 2\).

106
Table 7.3: Number of V-cycles and total CPU-time [s] of the optimized E-RK scheme and the LU-SGS + BE scheme for the steady laminar flow over a circular cylinder. Second-order ($p = 1$) SV method on two different grids.

<table>
<thead>
<tr>
<th>Grid</th>
<th>Scheme</th>
<th>VMG1</th>
<th>VMG2</th>
<th>VMG</th>
<th>CPU-time</th>
</tr>
</thead>
<tbody>
<tr>
<td>3,744 cells,</td>
<td>Opt. E-RK</td>
<td>500</td>
<td>9,500*</td>
<td>10,000*</td>
<td>145,094*</td>
</tr>
<tr>
<td>$AR_{max} = 2$</td>
<td>LU-SGS + BE</td>
<td>78</td>
<td>401</td>
<td>479</td>
<td>17,576</td>
</tr>
<tr>
<td>5,440 cells,</td>
<td>Opt. E-RK</td>
<td>500</td>
<td>9,500*</td>
<td>10,000*</td>
<td>211,859*</td>
</tr>
<tr>
<td>$AR_{max} = 162$</td>
<td>LU-SGS + BE</td>
<td>77</td>
<td>418</td>
<td>495</td>
<td>24,861</td>
</tr>
</tbody>
</table>

the $L_2$ norm of the residuals by 9 orders of magnitude. From this table it is seen that the LU-SGS + BE scheme is almost two orders of magnitude faster than the E-RK scheme in terms of CPU-time.

In order to study the convergence properties of the LU-SGS + BE solver, this test case was also computed on a grid with 5,440 triangular cells with $AR_{max} = 162$. Table 7.2 shows also the CFL number used for this new mesh. Note that the power law of the CFL number and its maximum value for LU-SGS + BE scheme were the same as for the $AR_{max} = 2$ case.

In Figure 7.6 the convergence histories of the E-RK and LU-SGS + BE
schemes for this mesh are shown. In Table 7.3, the number of V-cycles for each multigrid level, the total number of V-cycles and the total CPU-time are listed. It is observed that whereas the convergence of the E-RK scheme is significantly slowed down because of the increased $AR_{\text{max}}$, this is not the case for the LU-SGS + BE scheme. The required number of V-cycles to reduce the residual 12 orders of magnitude (495 V-cycles) is almost identical as for the $AR_{\text{max}} = 2$ case (479 V-cycles).

In Table 7.4 the drag coefficients $C_D$ for both simulations are summarized. Good agreement with the experimental data reported in Barlow et al. [11] is found.

Table 7.4: Drag coefficient for the steady laminar flow over a circular cylinder, obtained with second-order ($p = 1$) SV method and the LU-SGS + BE scheme on two different grids. Comparison with experimental measurements [11].

<table>
<thead>
<tr>
<th></th>
<th>Exp. [11]</th>
<th>3,744 cells, $AR_{\text{max}} = 2$</th>
<th>5,440 cells, $AR_{\text{max}} = 162$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_D$</td>
<td>1.536</td>
<td>1.526</td>
<td>1.531</td>
</tr>
</tbody>
</table>

To conclude this study on the cylinder, the effect of the low Mach number on the convergence rate of the LU-SGS + BE solver is presented. In Figure 7.7, the convergence histories of the LU-SGS + BE scheme for both meshes ($AR_{\text{max}} = 2$ and $AR_{\text{max}} = 162$) are shown for three values of the free-stream Mach number $M_\infty$: 0.15, 0.05, 0.005. The Reynolds and Prandtl numbers were again fixed to 40 and 0.72, respectively. The power law of the CFL number and its maximum value for LU-SGS + BE scheme was the same as used for the mesh with $AR_{\text{max}} = 2$ and $AR_{\text{max}} = 162$. The computations were stopped when the $L_2$ norm of the residuals was reduced 12 orders of magnitude.

In Table 7.5 the number of V-cycles for each multigrid level and the total number of V-cycles for the three values of the Mach number are listed. In this table, two values for the total number of V-cycles are indicated. The first one is the total number to reduce the $L_2$ norm of the residuals 10 orders of magnitude (VMG (-10)), while the second one is the total number for a reduction of 12 orders of magnitude (VMG (-12)). It is seen that with both meshes, the required number of V-cycles to transfer the solution from a first-order polynomial approximation to a second-order polynomial approximation (VMG1) decreases when the Mach number is reduced. The switch to a finer level is made when the $L_2$ norm of the coarse level resid-
7.1. TWO-DIMENSIONAL LAMINAR STEADY FLOW SIMULATIONS

Figure 7.7: Effect of the free-stream Mach number on the convergence histories of the LU-SGS + BE scheme for the steady laminar flow over a circular cylinder. Second-order \((p = 1)\) SV method on two different grids.

The residuals is smaller than a factor \(\eta_{\text{switch}}\) times the \(L_2\) norm of the fine level residual. In the present work \(\eta_{\text{switch}}\) is set to 0.001.

Figure 7.7 shows that the convergence rate of the LU-SGS + BE method for a residual norm higher than \(10^{-11}\) is not slowed by a decrease of the Mach number. On the contrary, for \(M_\infty = 0.005\), the required number of V-

Table 7.5: Effect of the free-stream Mach number on the number of V-cycles of the LU-SGS + BE scheme for the steady laminar flow over a circular cylinder. Second-order \((p = 1)\) SV method on two different grids.

<table>
<thead>
<tr>
<th>Grid</th>
<th>(M_\infty)</th>
<th>VMG1</th>
<th>VMG2</th>
<th>VMG (-10)</th>
<th>VMG (-12)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3,744 cells,</td>
<td>0.15</td>
<td>78</td>
<td>401</td>
<td>400</td>
<td>479</td>
</tr>
<tr>
<td>(AR_{max} = 2)</td>
<td>0.05</td>
<td>58</td>
<td>424</td>
<td>323</td>
<td>482</td>
</tr>
<tr>
<td>5,440 cells,</td>
<td>0.005</td>
<td>26</td>
<td>695</td>
<td>232</td>
<td>721</td>
</tr>
<tr>
<td>(AR_{max} = 162)</td>
<td>0.15</td>
<td>77</td>
<td>418</td>
<td>425</td>
<td>495</td>
</tr>
<tr>
<td></td>
<td>0.05</td>
<td>48</td>
<td>537</td>
<td>325</td>
<td>585</td>
</tr>
<tr>
<td></td>
<td>0.005</td>
<td>26</td>
<td>913</td>
<td>266</td>
<td>939</td>
</tr>
</tbody>
</table>

109
cycles to reduce the residual norm 10 orders of magnitude is smaller than the computation with $M_\infty = 0.15$. The low Mach number, combined with a mesh with high-aspect ratio, slightly affected the convergence rate of the LU-SGS + BE solver. For $M_\infty = 0.005$ and the mesh with $AR_{max} = 162$, the LU-SGS + BE scheme took 34 V-cycles more than that for the mesh with $AR_{max} = 2$. For a residual norm smaller than $10^{-11}$ the convergence rate of the LU-SGS + BE solver slows. This behavior is caused by round-off errors which are more influential at low Mach number.

### 7.1.2 Flow over a NACA0012 airfoil

The compressible steady laminar flow simulation over a NACA0012 airfoil is conducted at free-stream Mach number $M_\infty$ of 0.5, Reynolds number $Re$, based on the free-stream velocity module $|\vec{u}_\infty|$ and the airfoil chord $c$, of $5 \times 10^3$ and Prandtl number $Pr$ of 0.72. In Figure 7.8 the configuration of the test case is illustrated, where the incoming flow is from left to right. The airfoil is placed on the $\chi_1$ axis ($\chi_2 = 0$) of the computational domain.

![Figure 7.8: Configuration of the 2D NACA0012 airfoil test case.](image)

At the left-hand-side boundary (the inflow) the flow is prescribed to be uniform with zero angle of attack. The same boundary conditions are also applied to the upper and lower boundaries. At the right-hand-side boundary (the outflow), far enough from the profile, only the pressure is prescribed. At the airfoil wall, which is assumed to be adiabatic, the no-slip boundary condition is imposed. Maximum second-order ($p = 1$) SV results are given because of the curved boundaries, as discussed for the previous test case.


### 7.1. TWO-DIMENSIONAL LAMINAR STEADY FLOW SIMULATIONS

Table 7.6: CFL number for two-level V-cycles $p$-multigrid, for the steady laminar flow over a NACA0012 airfoil at zero angle of attack. Grid with 6,878 triangular cells and $AR_{\text{max}} = 2.5$.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>$SV_{p=0}$</th>
<th>$SV_{p=1}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Opt. E-RK</td>
<td>7, 0.5</td>
<td>4, 0.1</td>
</tr>
<tr>
<td>LU-SGS + BE</td>
<td>$10^n$</td>
<td>$10^n$</td>
</tr>
</tbody>
</table>

A grid with 6,878 triangular cells with $AR_{\text{max}} = 2.5$ was used.

In Table 7.6 the CFL numbers for both E-RK and LU-SGS + BE smoothers are listed. The maximum CFL number for the implicit scheme was set to $10^6$. In Figure 7.9, the convergence histories of the E-RK and LU-SGS + BE schemes are compared. The computations were stopped when the $L_2$ norm of residuals was reduced by 10 orders of magnitude.

In Table 7.7, the number of V-cycles for each multigrid level, the total number of V-cycles and the total CPU-time are listed. This table shows that the LU-SGS + BE scheme is approximately six times faster than the E-RK scheme in terms of CPU-time. For this test case, once the finest grid

![Figure 7.9: Convergence histories of the optimized E-RK scheme and the LU-SGS + BE scheme for the steady laminar flow over a NACA0012 airfoil at zero angle of attack. Second-order ($p = 1$) SV method, grid with 6,878 triangular cells and $AR_{\text{max}} = 2.5$.](#)
Table 7.7: Number of V-cycles and total CPU-time [s] of the optimized E-RK scheme and the LU-SGS + BE scheme for the steady laminar flow over a NACA0012 airfoil at zero angle of attack. Second-order ($p = 1$) SV method, grid with 6,878 triangular cells and $AR_{\text{max}} = 2.5$.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>VMG1</th>
<th>VMG2</th>
<th>VMG</th>
<th>CPU-time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Opt. E-RK</td>
<td>35</td>
<td>606</td>
<td>641</td>
<td>16,589</td>
</tr>
<tr>
<td>LU-SGS + BE</td>
<td>13</td>
<td>49</td>
<td>62</td>
<td>2,784</td>
</tr>
</tbody>
</table>

solution is reached, i.e. the VMG2 cycle is used and the second-order accurate solution is computed, the coarse grid solution takes 67% of the total CPU-time listed in Table 7.7.

In Figure 7.10, the Mach number contour and the stream traces are shown. This plot visually compares well with the results shown in Sun et al. [167].

The distribution of the skin friction coefficient $C_f$ and the pressure coefficient $C_p$ on the airfoil surface are plotted in Figure 7.11, where the horizontal axis is the normalized coordinate respect to the leading edge of the airfoil ($x_1/c$). The skin friction coefficient is defined by

$$C_f = \frac{\tau_w}{\frac{1}{2} \rho |\bar{u}_\infty|^2}, \quad (7.1)$$
7.1. TWO-DIMENSIONAL LAMINAR STEADY FLOW SIMULATIONS

where \( \tau_w \) is the local wall shear stress. Figure 7.11 shows good agreement with the experimental measurements presented in the AGARD Report AR-138 [1].

![Figure 7.11](image)

(a) Skin friction coefficient \( (C_f \text{ vs. } \chi'_{1/c}) \).

(b) Pressure coefficient \( (C_P \text{ vs. } \chi'_{1/c}) \).

Figure 7.11: Distribution of skin friction and pressure coefficients on the NACA0012 airfoil surface at zero angle of attack, obtained with second-order \((p = 1)\) SV method and the LU-SGS + BE scheme on a grid with 6,878 triangular cells and \( AR_{\text{max}} = 2.5 \). Comparison with experimental measurements [1].

To conclude this study on the NACA0012 airfoil, a comparison between the convergence behavior of the LU-SGS + BE scheme and the RK3/Implicit Residual scheme, proposed in Swanson et al. [169], is presented. The latter scheme is combined with a classical second-order FV scheme. In [169], the convergence of a three-stage E-RK scheme with \( h \)-multigrid is accelerated by preconditioning with a fully implicit operator, whose inverse is approximated with three point-wise SGS iterations. The aim is to compare the convergence behavior of the schemes, i.e. only the number of multigrid cycles to reduce the \( L_2 \) norm of the residuals 12 orders of magnitude. It should be noted that a comparison of the efficiency is difficult as both simulations have a different number of degrees of freedom (DOFs): the second-order \((p = 1)\) SV scheme for 2D problems on triangular grids has six times the number of DOFs of a classical FV scheme on quadrilateral grids. For this reason and because the simulations were done on different computers, no CPU-time comparisons is shown. The same conditions as in the previous computation were used except for the angle of attack which was set to 2.5\(^\circ\).

The RK3/implicit scheme used a structured quadrilateral mesh with 16,384 cells while the LU-SGS + BE scheme used a triangular mesh constructed from the quadrilateral mesh, as illustrated in Figure 7.12. The number of
triangular cells for the LU-SGS + BE scheme is then 32,768. The maximum aspect ratio is $AR_{max} = 136$. In the calculation of the RK3/Implicit Residual the CFL number was 16 during the first 8 multigrid cycles and then, it was increased to $10^3$. For the LU-SGS + BE scheme the CFL number was set to $10^6$ during the entire calculation. In Figure 7.13, the convergence histories of the RK5 Standard scheme, the RK3/Implicit Residual scheme and the LU-SGS + BE scheme are compared.

In Table 7.8 the total number of V-cycles are listed. The LU-SGS + BE requires 19 cycles more than the RK3/Implicit Residual code and nearly one order of magnitude less than the RK5 Standard code. In terms of multigrid

<table>
<thead>
<tr>
<th>Scheme</th>
<th>Total Number of Cycles</th>
</tr>
</thead>
<tbody>
<tr>
<td>RK5 Standard</td>
<td>773</td>
</tr>
<tr>
<td>RK3/Implicit Residual, Swanson et al. [169]</td>
<td>58</td>
</tr>
<tr>
<td>LU-SGS + BE</td>
<td>77</td>
</tr>
</tbody>
</table>
7.1. TWO-DIMENSIONAL LAMINAR STEADY FLOW SIMULATIONS

Figure 7.13: Convergence histories of the RK5 Standard scheme, the RK3/Implicit Residual scheme and the LU-SGS + BE scheme for the flow over a NACA0012 airfoil at $2.5^\circ$ angle of attack. Second-order FV/SV methods, structured quadrilateral/triangular grid, $AR_{max} = 136$.

cycles, the RK3/Implicit Residual scheme is more efficient than the LU-SGS + BE scheme. However, the work per DOF of the RK3/Implicit Residual scheme is approximately twice that of the LU-SGS + BE scheme. The RK3/Implicit Residual scheme employs 12 point-wise SGS (3 SGS sweeps plus a residual calculation times three RK stages) per DOF, while the LU-SGS + BE scheme requires only 6 SGS sweeps per DOF. However, the LU-SGS + BE scheme requires the computation of the LHS of Equation (5.8), which is time consuming since its size is quite large. In fact, the total number of diagonal block matrix elements that have to be stored for a tetrahedral mesh with $N$ cells with polynomial degree $p$ increases with $p^4$ in 2D and $p^6$ in 3D.

Table 7.9: Drag coefficient for the flow over a NACA0012 airfoil at $2.5^\circ$ angle of attack obtained with the RK5 Standard scheme, the RK3/Implicit Residual scheme and the LU-SGS + BE scheme. Second-order FV/SV methods, structured quadrilateral/triangular grid, $AR_{max} = 136$.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>$C_D$</th>
<th>$C_L$</th>
</tr>
</thead>
<tbody>
<tr>
<td>RK5 Standard</td>
<td>0.0568425</td>
<td>0.0331927</td>
</tr>
<tr>
<td>RK3/Implicit Residual, Swanson et al. [169]</td>
<td>0.0568523</td>
<td>0.0333319</td>
</tr>
<tr>
<td>LU-SGS + BE</td>
<td>0.0562196</td>
<td>0.0331650</td>
</tr>
</tbody>
</table>
Finally, in Table 7.9 the drag coefficient $C_D$ and the lift coefficient $C_L$ for the three simulations are summarized. This table shows good agreement between the results of the three codes.

### 7.1.3 Flow in a channel with a backward-facing step

This test case is fundamental in design and geometry and consequently it is found in a variety of engineering applications. The flow separation process caused by the sudden change in geometry has been used extensively in applications, usually in order to create a recirculation region or a sudden change in pressure.

In Figure 7.14 the characteristic lengths of the channel and a sketch of the main flow features are illustrated. The incoming flow is from left to right. A sudden expansion of $H/h = 2$ is used, see Barton [12]. The position of the inlet section ($L_{in}$) is fixed to $10h$. An inlet Mach number $M_{inlet}$ of 0.2, a Reynolds number $Re$ of 800, based on twice the inlet channel height and the inlet bulk velocity, and a Prandtl number $Pr$ of 0.72 are imposed. Uniform mass density and velocity profiles are prescribed at the inlet section so that the Reynolds number is equal to the prescribed value. At the channel walls, which are assumed to be adiabatic, no-slip boundary conditions are imposed. At the right-hand-side boundary (the outflow), a pressure outlet boundary condition is used.

![Figure 7.14: Configuration of the 2D backward facing step test case and locations of the recirculating regions.](image-url)
Table 7.10: CFL number for three-level V-cycles $p$-multigrid, for the backward facing step flow.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>SV $p = 0$</th>
<th>SV $p = 1$</th>
<th>SV $p = 2$</th>
<th>SV $p = 3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Opt. E-RK</td>
<td>3.5, 0.7</td>
<td>0.5, 1.0</td>
<td>0.05, 0.001</td>
<td>0.01, 0.0001</td>
</tr>
<tr>
<td>LU-SGS + BE</td>
<td>$10^n$</td>
<td>$10^n$</td>
<td>$10^n$</td>
<td>$10^n$</td>
</tr>
</tbody>
</table>

Firstly, an unstructured grid with 8,503 triangular cells with a maximum aspect ratio of 42 was used. In Table 7.10, the CFL numbers are listed for both E-RK and LU-SGS + BE schemes. The solution was computed with third- ($p = 2$) and fourth-order ($p = 3$) SV methods. For the LU-SGS + BE smoother the CFL number started from one and it reached its maximum value using the CFL-law indicated in Table 7.10. The maximum CFL number was set to $10^6$ for first- and second-order SV schemes and to $10^4$ for third- and fourth-order SV schemes. This choice allows to have stable computations and a better convergence in terms of CPU-time. In Figure 7.15, the convergence histories of the optimized E-RK scheme and the LU-SGS scheme are compared, for third- and fourth-order SV methods.

Tables 7.11 and 7.12 list the number of V-cycles for each multigrid level,

![Figure 7.15](image-url)  
(a) $p = 2$.  
(b) $p = 3$.  

Figure 7.15: Convergence histories of the optimized E-RK scheme and the LU-SGS + BE scheme for the backward-facing step flow. Third- ($p = 2$) and fourth-order ($p = 3$) SV methods, grid with 8,503 triangular cells and $AR_{max} = 42$.  

117
the total number of V-cycles and the total CPU-time to obtain a reduction of 10 orders of magnitude of the $L_2$ residual norms, resp. for the third- and fourth-order SV schemes. The super-script * indicates that the residuals are not yet converged. From this table it is seen that the LU-SGS + BE scheme is more than two orders of magnitude faster than the E-RK scheme for both third- and fourth-order SV methods.

Table 7.11: Number of V-cycles and total CPU-time [s] of the optimized E-RK scheme and the LU-SGS + BE scheme for the backward-facing step flow. Third-order ($p = 2$) SV method, grid with 8,503 triangular cells and $AR_{max} = 42$.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>VMG1</th>
<th>VMG2</th>
<th>VMG3</th>
<th>VMG4</th>
<th>VMG</th>
<th>CPU-time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Opt. E-RK</td>
<td>250</td>
<td>421</td>
<td>9,329*</td>
<td>10,000*</td>
<td>50,264*</td>
<td></td>
</tr>
<tr>
<td>LU-SGS + BE</td>
<td>32</td>
<td>20</td>
<td>244</td>
<td>296</td>
<td>12,032</td>
<td></td>
</tr>
</tbody>
</table>

Table 7.12: Number of V-cycles and total CPU-time [s] of the optimized E-RK scheme and the LU-SGS + BE scheme for the backward-facing step flow. Fourth-order ($p = 3$) SV method, grid with 8,503 triangular cells and $AR_{max} = 42$.

<table>
<thead>
<tr>
<th>Scheme</th>
<th>VMG1</th>
<th>VMG2</th>
<th>VMG3</th>
<th>VMG4</th>
<th>VMG</th>
<th>CPU-time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Opt. E-RK</td>
<td>250</td>
<td>421</td>
<td>491</td>
<td>8,838*</td>
<td>10,000*</td>
<td>85,426*</td>
</tr>
<tr>
<td>LU-SGS + BE</td>
<td>32</td>
<td>20</td>
<td>79</td>
<td>315</td>
<td>446</td>
<td>29,651</td>
</tr>
</tbody>
</table>

In order to study the convergence properties of the LU-SGS + BE solver this test case was also computed using a finer grid, obtained by doubling the number of cells of the previous mesh in each direction. Therefore, the new mesh has 34,015 triangular cells with still a maximum aspect ratio of 42. The power law of the CFL number and its maximum value for LU-SGS + BE scheme are the same as for the previous grid.

Table 7.13: Number of V-cycles and total CPU-time [s] of the LU-SGS + BE scheme for the backward-facing step flow. Third- ($p = 2$) and fourth-order ($p = 3$) SV methods, grid with 34,015 triangular cells and $AR_{max} = 42$.

<table>
<thead>
<tr>
<th>SV method</th>
<th>VMG1</th>
<th>VMG2</th>
<th>VMG3</th>
<th>VMG4</th>
<th>VMG(-10)</th>
<th>CPU-time</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p = 2$</td>
<td>40</td>
<td>23</td>
<td>249</td>
<td>-</td>
<td>312</td>
<td>62,566</td>
</tr>
<tr>
<td>$p = 3$</td>
<td>40</td>
<td>23</td>
<td>83</td>
<td>320</td>
<td>465</td>
<td>160,115</td>
</tr>
</tbody>
</table>
7.1. TWO-DIMENSIONAL LAMINAR STEADY FLOW SIMULATIONS

In Figure 7.16, the convergence histories of the LU-SGS + BE scheme are compared for both grids. It can be seen that, according to the multigrid theory, the asymptotic convergence rate is unaffected by the mesh size. Notice that, as shown in Table 7.13, there is a small grid effect on the FMG procedure, where slightly more cycles are needed on the fine grid to reach the maximum order of accuracy. This feature was also reported in Van den Abeele et al. [174] and Bassi et al. [14], respectively for explicit and implicit smoothers.

![Convergence histories for different grids](image1)

(a) $p = 2$.

(b) $p = 3$.

Figure 7.16: Convergence histories of the LU-SGS + BE scheme for the backward-facing step flow on two different grids. Third- ($p = 2$) and fourth-order ($p = 3$) SV methods.

In Figure 7.17, the stream traces obtained with fourth-order SV method and the LU-SGS + BE scheme are shown.

![Stream traces](image2)

Figure 7.17: Stream traces for the backward facing step flow obtained with fourth-order ($p = 3$) SV method and the LU-SGS + BE scheme. Grid with 8,503 triangular cells and $AR_{max} = 42$. 

119
Table 7.14 compares the re-attachment and separation positions for both grids with those reported in Barton [12]. For both grids, the results are in good agreement with the reference data.

Table 7.14: Re-attachment and separation positions for the backward-facing step flow obtained with third- \((p = 2)\) and fourth-order \((p = 3)\) SV methods and the LU-SGS + BE scheme on two different grids. Comparison with reference numerical solution [12].

<table>
<thead>
<tr>
<th></th>
<th>(l_1 h)</th>
<th>(l_2 h)</th>
<th>(l_3 h)</th>
</tr>
</thead>
<tbody>
<tr>
<td>SV (p = 2), 8,503 cells</td>
<td>11.76</td>
<td>9.28</td>
<td>20.74</td>
</tr>
<tr>
<td>SV (p = 3), 8,503 cells</td>
<td>11.72</td>
<td>9.24</td>
<td>20.71</td>
</tr>
<tr>
<td>SV (p = 2), 34,015 cells</td>
<td>11.73</td>
<td>9.27</td>
<td>20.70</td>
</tr>
<tr>
<td>SV (p = 3), 34,015 cells</td>
<td>11.69</td>
<td>9.24</td>
<td>20.69</td>
</tr>
</tbody>
</table>
Chapter 8

Application II: spectral difference method

In this chapter, the main results achieved by applying the spectral difference method and the non-linear LU-SGS algorithm to problems governed by 2D and 3D compressible Navier-Stokes and filtered Navier-Stokes equations (LES approach) are presented.

The non-linear LU-SGS algebraic solver and the LES approach for the SD method have been implemented in the COOLFluiD code which was developed at the von Karman Institute for Fluid Dynamics. COOLFluiD is a collaborative simulation environment written in C++. It focuses on complex computational fluid dynamics (CFD) involving multi-physics phenomena. Every physical model or numerical method is a separate plug-in component which is loaded on demand. This creates high-performant solvers, each dedicated to a specific application, while reusing the same components. The COOLFluiD platform is also a collaborative platform, where diverse European research centers join together their developments. More information about COOLFluiD can be found in the PhD theses of T. Quintino [136], A. Lani [95] and T. Wuilbaut [195], and on the COOLFluiD project web site [137].

A Roe FDS flux as a Riemann solver and the BR2 approach for the diffusive terms were used. The grids for all the test cases were generated with Gmsh [54] and Fluent/Gambit™. The former code allows second-order polynomial approximation of the boundary elements. In fact, high-order schemes require an accurate representation of curved boundaries, especially on the
relatively coarse grids that are being used in combination with high-order polynomial representations of the solution. The calculations were done on a server with Dual Core AMD Opteron™ processors with a clock-speed of 2,412MHz. Wherever possible, direct validation against experimental data and/or reference solutions available in literature are made.

8.1 Steady laminar flow simulations

In this section, the 2D steady laminar flow around a NACA0012 airfoil at zero angle of attack and the 3D steady laminar flow through a 90° bending square duct are considered. In order to assess the convergence properties of the LU-SGS + BE scheme, a Newton-Raphson GMRES algebraic solver with BE scheme, as discussed in Appendix C, is used as reference efficient time marching scheme. In COOLFluiD, the PETSc suite\(^1\) is used for the GMRES algorithm [3].

For convergence, the residual of the continuity equation was considered. In this work, the residuals are normalized by the corresponding residuals of the first iteration and they are shown in base ten logarithmic scale.

8.1.1 Flow over a NACA0012 airfoil

The simulation is conducted at Reynolds number \(Re\) of 5,000 and free-stream Mach number \(M_\infty\) of 0.5. The Reynolds number is based on the module of the free-stream velocity vector \(|\vec{u}_\infty|\) and the airfoil chord \(c\). The Prandtl number \(Pr\) is set to 0.72, which is the standard value for air. In Figure 8.1 the configuration of the test case is illustrated, where the incoming flow is from left to right. The airfoil is placed on the \(\chi_1\) axis (\(\chi_2 = 0\)) of the computational domain. At the left-hand-side boundary (the inflow) the flow is prescribed to be uniform with zero angle of attack, as described in Section 3.3.1. The same boundary conditions are also applied to the upper and lower boundaries. At the right-hand-side boundary (the outflow), sufficiently far from the airfoil, only the pressure is prescribed, as discussed in Section 3.3.3. At the airfoil wall, which is assumed to be adiabatic, the no-slip boundary condition described in Section 3.3.4 is imposed.

A grid with 2,925 quadrilateral cells with \(AR_{max} = 9\) near the airfoil wall was used. In Figure 8.2, two details of this grid are shown. The compu-

\(^1\)PETSc suite is a data structures and routines for the scalable (parallel) solution of scientific applications modeled by partial differential equations.
8.1. STEADY LAMINAR FLOW SIMULATIONS

\[ R = 6c \]

\[ c \]

\[ \chi_1 \]

\[ \chi_2 \]

Figure 8.1: Configuration of the 2D NACA0012 airfoil test case.

Simulations were performed using second- \((p = 1)\) to fourth-order \((p = 3)\) SD methods, with the airfoil wall represented by a quadratic mapping. Freestream flow conditions were used as initial solution.

Figure 8.3 compares the experimental distribution of the pressure coefficient \(C_P\) reported in [1] with those obtained using second- to fourth-order SD methods and the LU-SGS + BE scheme. In this figure, the horizontal axis is the normalized coordinate respect to the leading edge of the airfoil \((\chi_1/c)\). It can be seen that, the SD results get closer to the experimental data by increasing the order of the polynomial reconstruction. Note that, third- and fourth-order accurate solutions are nearly indistinguishable, indicating the order-independent flow convergence.

Figure 8.2: Two details of the grid for the steady laminar flow over a NACA0012 airfoil at zero angle of attack. 2,925 quadrilateral cells with \(AR_{max} = 9\).
The Reynolds number is near to the upper limit for a steady laminar flow. A feature of this test case is the separation region of the flow occurring near the trailing edge, which causes the formation of a small recirculation bubble that extends in the near-wake region of the airfoil. The comparison between the separation point locations and the drag coefficients $C_D$ computed using second- to fourth-order SD methods are summarized in Table 8.1. It is seen that the order-independent convergence is also verified for these quantities.

Table 8.1: Effect of the $p$-refinement on the drag coefficient and the location of the separation point, for the steady laminar flow over a NACA0012 airfoil at zero angle of attack, obtained with SD methods and the LU-SGS + BE scheme.

<table>
<thead>
<tr>
<th>$p$</th>
<th>$C_D$</th>
<th>Separation point</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.05448</td>
<td>86.81%</td>
</tr>
<tr>
<td>2</td>
<td>0.05476</td>
<td>81.43%</td>
</tr>
<tr>
<td>3</td>
<td>0.05476</td>
<td>81.42%</td>
</tr>
</tbody>
</table>
Algebraic solver performance

The non-linear LU-SGS and the Newton-Raphson GMRES solvers, combined with the BE scheme, were both used to solve the non-linear algebraic system arising from the spatial and temporal discretizations. A local time stepping technique, as described in Section 5.3, was also used. The CFL number was computed using the following power law

\[
\sigma = \min \left[ \frac{1}{4} \times 2^{n-1}, \sigma_{\text{max}} \right],
\]

where the superscript \( n \) indicates the time iteration index. The upper limit \( \sigma_{\text{max}} \) depends on the specific test case and will be defined further on. For the non-linear LU-SGS algorithm, a maximum of thirty SGS sweeps per time iteration and/or a prescribed tolerance of \( 10^{-5} \) on the change of the \( L_2 \) norm of the solution variation \( \Delta W_{cc}^{m+1} \) were imposed (see system (5.8) in Section 5.1). The GMRES algorithm was preconditioned with the additive Schwarz method \cite{cite} and its convergence criterion was a linear system residual drop of five orders of magnitude. An upper limit of two thousand of Krylov sub-spaces was used. The computations were done on two processors.

The convergence history versus the number of iterations and the wall time to drop the \( L_2 \) norm of the residuals 10 orders of magnitude are plotted in Figures 8.4 and 8.5, for second- to fourth-order SD methods. The upper limit of the CFL number \( \sigma_{\text{max}} \) for the non-linear LU-SGS + BE scheme was set to 80, 40 and 20, respectively for second- third- and fourth-order accurate solutions. For the Newton-Raphson GMRES solver these limits were fixed to 150, 100 and 80. This was necessary to obtain a fast convergence and to maintain stability.

Figure 8.4 shows that, for second- and third-order SD methods, the Newton-Raphson GMRES solver needs fewer iterations as compared to the non-linear LU-SGS solver. On the contrary, for fourth-order SD scheme, the non-linear LU-SGS solver converged in fewer iterations as compared to the Newton-Raphson GMRES solver. In term of wall time, the Newton-Raphson GMRES solver is three times faster than the non-linear LU-SGS solver, for second-order SD method. However, for third-order SD method, the former algorithm needs almost the same amount of time as compared to the non-linear LU-SGS solver. In fact, the GMRES method had difficulties to converge and used about one thousand Krylov subspace at almost each Newton iteration. This explains the large computational time. For
fourth-order accurate calculation, the non-linear LU-SGS solver performed better, although it used almost thirty SGS sweeps at each time iteration.

Figure 8.4: Non-linear LU-SGS and Newton-Raphson GMRES convergence histories for the steady laminar flow over a NACA0012 airfoil at zero angle of attack. SD method with $p$-refinement.

Figure 8.5: Non-linear LU-SGS and Newton-Raphson GMRES convergence histories for the steady laminar flow over a NACA0012 airfoil at zero angle of attack. SD method with $p$-refinement.
8.1. STEADY LAMINAR FLOW SIMULATIONS

The total memory requirements for these computations are summarized in Table 8.2. It can be seen that the Newton-Raphson GMRES method requires far more memory than the non-linear LU-SGS method.

Table 8.2: Non-linear LU-SGS and Newton-Raphson GMRES memory requirements [MB] for the steady laminar NACA0012 airfoil computations. SD method with \( p \)-refinement.

<table>
<thead>
<tr>
<th>SD method</th>
<th>Non-linear LU-SGS</th>
<th>Newton-Raphson GMRES</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p = 1 ) (11,700 DOFs)</td>
<td>30</td>
<td>174</td>
</tr>
<tr>
<td>( p = 2 ) (26,325 DOFs)</td>
<td>65</td>
<td>771</td>
</tr>
<tr>
<td>( p = 3 ) (46,800 DOFs)</td>
<td>155</td>
<td>2,325</td>
</tr>
</tbody>
</table>

8.1.2 Flow through a 90° bending square duct

The experiment of Humphrey et al. [78], which measured the flow through a strongly curved 90° bend square duct, is used as a 3D laminar steady state test case in the present section. In Figure 8.6 the configuration of the test case is illustrated. The side length of the square cross section is

![Figure 8.6: Configuration of the 3D 90° bending square duct test case.](image)
denoted by $H$. A straight duct of $10H$ length is located ahead of the bend in order to produce inlet laminar flow that is fully developed. The bend is located in the vertical plane with a $5H$ length of straight duct attached to its downstream. The bend has an inner radius of $r_i = 1.8H$ and an outer radius of $r_o = 2.8H$.

A Reynolds number $Re$ of 790 and a Prandtl number $Pr$ of 0.72 are imposed. Uniform mass density and velocity profiles are prescribed at the inlet section so that the Reynolds number based on the side of the square cross section $H$ is equal to the prescribed value. In order to obtain a quasi-incompressible flow field, the inlet Mach number $M_{inlet}$ is set to 0.1. At the outflow the pressure is prescribed to be uniform. At the walls, which are assumed to be adiabatic, no-slip boundary conditions are imposed.

![Figure 8.7: Grid for the steady laminar flow in square pipe with 90° bend. 4,505 hexahedral cells with $AR_{max} = 13$.](image)

A coarse grid with 4,505 hexahedral cells and $AR_{max} = 13$ was used (see Figure 8.7). This test case was computed with second- ($p = 1$) and third-order ($p = 2$) SD methods. In Figure 8.8, the Mach number contours obtained with these schemes are shown. It can be seen that the solution gets smoother by increasing the order of the polynomial reconstruction, indicating the solution is more accurate. Note that, the third-order solution was computed using a second-order polynomial approximation (quadrating mapping) of the boundary elements.
8.1. STEADY LAMINAR FLOW SIMULATIONS

Figure 8.8: Mach number contours in the symmetry plane under $p$-refinement, for the steady laminar flow in square pipe with 90° bend, obtained with SD methods and the LU-SGS + BE scheme. $\Delta M = 0.02$.

In order to study the order-independent flow convergence, the stream-wise (axial) velocity profile at four locations, namely at 0°, 30°, 60° and 90° along the bend, was analyzed. In Figure 8.9, the numerically obtained stream-wise velocity profiles in the symmetry plane are shown. In this figure the experimental data of Humphrey et al. [78] are also plotted for comparison. The horizontal axis is the normalized radial distance and the vertical axis is the normalized velocity in the stream-wise direction. The second-order computation, with only 36,040 DOFs, is under-resolved. For the third-order computation, which had 121,635 DOFs, the match with the experiment is very good at the first two stations. However, when the flow begins to form swirls at $\theta = 60^\circ$, a slight discrepancy between the computed and experimental results appears. This deviation can also be found in other numerical calculations [145, 196] and is thought to be caused by any small
change in the Reynolds number. Nevertheless, the peak of stream-wise velocity near the outside wall and the second peak near the inside wall at $\theta = 60^\circ$ and $\theta = 90^\circ$ are well captured by third-order SD scheme.

Figure 8.9: Axial velocity profiles at four locations along the bend under $p$-refinement, for the steady laminar flow in square pipe with $90^\circ$ bend, obtained with SD methods and the LU-SGS + BE scheme. Comparison with experimental measurements [78].
8.1. STEADY LAMINAR FLOW SIMULATIONS

Algebraic solver performance

The non-linear LU-SGS and the Newton-Raphson GMRES algorithms, combined with the BE scheme, were both used to perform the pseudo time integration to get steady-state solutions. A local time stepping technique was used. The CFL-law was the same as for the steady laminar flow past a NACA0012 airfoil presented in the previous section. For the non-linear LU-SGS algorithm, a maximum of hundred SGS sweeps per time iteration and/or a prescribed tolerance of $10^{-5}$ on the change of the $L_2$ norm of the solution variation $\Delta W_{cc}^{m+1}$ were imposed. The GMRES linear system solver was used in combination with the additive Schwarz preconditioning. Its convergence criterion was a linear system residual drop of five orders of magnitude, with a maximum of two thousand Krylov sub-spaces. The computations were done on nine processors. For both algebraic solvers, the upper limit of the CFL number $\sigma_{max}$ was set to 800 and 20 for second- and third-order SD methods respectively.

The convergence history versus the number of iterations and the wall time to drop the $L_2$ norm of the residuals 10 orders of magnitude are plotted in Figures 8.10 and 8.11 respectively. It can be observed that the non-linear LU-SGS solver converged with both second- and third-order SD methods, whereas the Newton-Raphson GMRES solver did converge only for second-order SD method. Moreover, for second-order accurate solution the non-linear LU-SGS solver performed better than the Newton-Raphson GMRES solver. The GMRES solver had some difficulties to converge and used almost thousand five hundred Krylov sub-spaces at each Newton iteration.

Note that, for the non-linear LU-SGS solver, third-order accurate solution required twice the number of iterations needed by the second-order accurate computation. However, since the total computational load required by both SD method and non-linear LU-SGS solver does not scale linearly with the order of accuracy, the wall time required by the third-order accurate solution is not twice of that of the second-order accurate simulation.

The total memory requirements for these computations are summarized in Table 8.3. It can be seen that, the non-linear LU-SGS solver needs much less memory than the Newton-RGMRES solver and the difference between them is more pronounced than for the previous test case, which was two-dimensional. Nevertheless, the amount of memory required by the non-linear LU-SGS method is still quite large and it increases rapidly with increasing polynomial orders $p$, as discussed in Section 5.1.
CHAPTER 8. APPLICATION II: SPECTRAL DIFFERENCE METHOD

Figure 8.10: Non-linear LU-SGS and Newton-Raphson GMRES convergence histories for the steady laminar flow in square pipe with 90° bend. SD method with \(p\)-refinement.

Figure 8.11: Non-linear LU-SGS and Newton-Raphson GMRES convergence histories for the steady laminar flow in square pipe with 90° bend. SD method with \(p\)-refinement.
8.2. UNSTEADY LAMINAR FLOW SIMULATIONS

Table 8.3: Non-linear LU-SGS and Newton-Raphson GMRES memory requirements [MB] for the steady laminar square pipe with 90° bend computations. SD method with \( p \)-refinement.

<table>
<thead>
<tr>
<th>SD method</th>
<th>Non-linear LU-SGS</th>
<th>Newton-Raphson GMRES</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p = 1 ) (36,040 DOFs)</td>
<td>250</td>
<td>2,648</td>
</tr>
<tr>
<td>( p = 2 ) (121,635 DOFs)</td>
<td>1,919</td>
<td>11,452</td>
</tr>
</tbody>
</table>

8.2 Unsteady laminar flow simulations

In this section, the 2D unsteady laminar flow over an open cavity, a square cylinder and a circular cylinder are considered. Time marching scheme was done using the LU-SGS + BDF2 scheme discussed in Section 5.2. The non-linear system (5.16) was solved with a prescribed tolerance of \( 10^{-6} \) on the change of the \( L_2 \) norm of the solution variation \( \Delta \mathbf{w}_{cc}^{m+1} \) and/or a maximum number of hundred SGS sweeps. During the calculations, the maximum number of SGS sweeps was never required. However, during the initial time steps (depending on the test case and the initial solution) the number of inner LU-SGS sweeps was between thirty and forty. Afterwards, the number decreased and reached a values which was between seven and ten (depending again on the test case). The computations were done with eight processors.

The flow solution of the first two test cases was used by the present author to provide the acoustic sources for aerodynamic sound field simulation in the time domain with a Ffowcs-Williams Hawkings (FW-H) approach, see Parsani et al. [123].

8.2.1 Flow over an open cavity

The flow in an open cavity is dominated by vorticity production and transport and is very complex. So far, several numerical and experimental investigations have been performed by researchers [4, 8, 55, 96, 141, 148]. The case studied in the present section has a length to depth ratio \( L/D = 4 \), a inlet Mach number of \( M_{inlet} = 0.15 \), and a Reynolds number \( Re = 1.5 \times 10^3 \). The Reynolds number is based on the module of the far-field velocity vector \( |\mathbf{u}_\infty| \) and the depth of the cavity \( D \). At this Reynolds number, the flow is laminar. Because no experimental data exist for this case, the 2D DNS data presented in Larsson et al. [96] are used as reference solution.
The computational domain is shown in Figure 8.12, where the flow is from left to right. The cavity is discretized in the stream-wise and normal directions with 50 and 30 points respectively. The maximum aspect ratio $AR_{max}$ of the first layer of the cells close to the walls is 30. At the right boundary, a damping zone (buffer layer) is introduced by progressively increasing the size of the cells towards the outflow. This type of buffer layer increases the numerical damping introduced by the solution method, avoiding strong spurious wave reflections which might contaminate the flow field. The buffer layer contains 5 cells and its length is $4D$. The total number of quadrilateral cells is 11,586. At the inflow boundary, the incoming boundary layer thickness and the mass density profiles are set equal to those of the reference DNS solution [96]. At the outlet, which is sufficiently far from the cavity, only the pressure is prescribed. At the top boundary the 1D characteristic boundary condition is used. The walls are assumed to be adiabatic and the no-slip boundary condition is used there.

The test case was solved with third- ($p = 2$) and fourth-order ($p = 3$) SD methods. Therefore, the total number of DOFs is 104,274 and 185,376, respectively for third- and fourth-order computations. Notice that, the number of DOFs for the highest-order SD scheme is much lower than the number of DOFs used in the DNS of Larsson et al. [96] and Ask et al. [9]. In the former reference, the number of DOFs is 1,120,080, whereas in the latter one, it is of the order of 500,000. The time step $\Delta t$ used for the computations started from 0.00001 and increased linearly till 0.005. This time step allowed about 3,280 time samples per flow period.
8.2. UNSTEADY LAMINAR FLOW SIMULATIONS

Figure 8.13: Mach number contour and stream traces, for the unsteady laminar flow over an open cavity, obtained with fourth-order \((p = 3)\) SD method. \(\Delta M = 0.02\).

In Figure 8.13, the Mach number contour and the stream traces obtained with fourth-order SD method are plotted. This figure shows that, despite the geometric simplicity, many complicated flow patterns characterize the flow.

In Figure 8.14, the cavity drag coefficient per unit width \(C_D\) at statistically stationary state computed with fourth-order SD method is shown. The Strouhal number associated to the frequency of the fluctuating drag coefficient \(f_{C_D}\) and defined by

\[
St = \frac{f_{C_D} L}{|\bar{u}_\infty|},
\]

the time averaged drag coefficient \(\langle C_D \rangle\) and the mean drag pressure coefficient\(^2\) \(\langle C_{Dp} \rangle\) are listed in Table 8.4. In this table, reference coefficients are also presented for comparison [96]. It can be seen that the mean drag coefficients and the Strouhal number are in good agreement with the DNS solution. Moreover, the accuracy of the solution improves by increasing the order of the SD scheme. In fact, the coefficients obtained with fourth-order accurate solution are very close to those reported in Larsson et al. [96]. This is a excellent achievement since the number of spectral difference DOFs is significantly lower than that of the DNS computations [9, 96].

\(^2\)The pressure drag coefficient is evaluated through integration of the components in the free-stream direction of the pressure forces acting on the after-body.
CHAPTER 8. APPLICATION II: SPECTRAL DIFFERENCE METHOD

Figure 8.14: Cavity drag coefficient obtained with fourth-order \((p = 3)\) SD method.

Table 8.4: Effect of the \(p\)-refinement on the Strouhal number and the mean drag coefficients for the unsteady laminar flow over an open cavity, obtained with SD methods. Comparison with DNS reference solution [96].

<table>
<thead>
<tr>
<th>Solution</th>
<th>(St)</th>
<th>(\langle C_D \rangle)</th>
<th>(\langle C_P \rangle)</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNS [96]</td>
<td>0.243</td>
<td>0.377</td>
<td>0.402</td>
</tr>
<tr>
<td>SD (p = 2) (104, 274 DOFs)</td>
<td>0.246</td>
<td>0.385</td>
<td>0.409</td>
</tr>
<tr>
<td>SD (p = 3) (185, 376 DOFs)</td>
<td>0.243</td>
<td>0.379</td>
<td>0.403</td>
</tr>
</tbody>
</table>

The total memory requirements for the present computations are summarized in Table 8.5. It should be noted that the fourth-order accurate simulation required about three times more memory as compared to the third-order one.

Table 8.5: Non-linear LU-SGS memory requirements [MB] for the unsteady laminar flow simulations over an open cavity. SD method with \(p\)-refinement.

<table>
<thead>
<tr>
<th>SD method</th>
<th>Non-linear LU-SGS</th>
</tr>
</thead>
<tbody>
<tr>
<td>(p = 2) (104, 274 DOFs)</td>
<td>257</td>
</tr>
<tr>
<td>(p = 3) (185, 376 DOFs)</td>
<td>612</td>
</tr>
</tbody>
</table>
8.2. UNSTEADY LAMINAR FLOW SIMULATIONS

8.2.2 Flow past a square cylinder

The flow past a square cylinder represents a benchmark test case for external flow past bluff bodies. This flow has been the subject of intense experimental and numerical research in the past [117, 118, 161] and it is a relevant application in the field of aeroacoustics. A practical example can be found in automotive applications such as the noise generated by a luggage carrier system.

In Figure 8.15 the configuration of the test case is illustrated. At the left boundary (the inflow), the flow is prescribed to be uniform. The same boundary condition is applied to the upper and lower boundaries of the domain. At the right boundary, which is sufficiently far from the cylinder, uniform pressure profile is prescribed. At the cylinder wall, which is assumed to be adiabatic, the no-slip boundary condition is imposed.

The flow has a free-stream Mach number of $M_\infty = 0.5$, and a Reynolds number of $Re = 200$. The Reynolds number is based on the module of the free-stream velocity vector $|\vec{u}_\infty|$ and the height of the cylinder $H$. At this Reynolds number, the regime is laminar and it usually persists up to a Reynolds number of about 400. Moreover, the vortex shedding is characterized by one very well-defined frequency [117] and the flow field can be described well with a 2D model. In fact, in Murakami et al. [114], it has been shown that the 2D approach gives an accurate description of the physical behavior of the problem, when the vortex shedding is concentrated around one frequency, which is the case for $Re \lesssim 400$. Because no
experimental data exist for this case, the 2D DNS results presented in De Roeck et al. [144] are used as reference solution.

The test case was solved using second- ($p = 1$) to fifth-order ($p = 4$) SD methods on a very coarse mesh with 2,205 quadrilateral cells, as shown in Figure 8.16. Notice that, the number of DOFs for the fifth-order accurate solution is $55,125$ which is about 22% of the number of DOFs used for the reference DNS solution [144]. The maximum cell aspect ratio $AR_{\text{max}}$ close to the cylinder was small. The time step $\Delta t$ used for the computations started from $0.00001$ and increased linearly till $0.08$. This allowed about 180 time samples per period of the vortex shedding.

![Figure 8.16: Grid for the unsteady laminar flow past a square cylinder. 2,205 quadrilateral cells and $AR_{\text{max}} = 3$.](image)

The instantaneous Mach number contours at the end of the simulations, obtained with second- to fifth-order accurate SD schemes, are shown in Figure 8.17. The initial phase of the computations, where the solution changes from a uniform flow field, to a fully developed unsteady flow field with a von Karman vortex street, is difficult to resolve and changes significantly for different order of accuracy. Because of this, there is always a phase shift between the solutions at the end of each calculation. The improvement in the quality of the solution is however clearly visible.

The evolution of the lift $C_L$ and the drag $C_D$ coefficients obtained with fourth-order SD scheme, are shown in Figure 8.18. The Strouhal number associated to the frequency of the fluctuating lift coefficient $f_{C_L}$, the time averaged drag coefficient $\langle C_D \rangle$, the root mean square (RMS) value of the
8.2. UNSTEADY LAMINAR FLOW SIMULATIONS

Figure 8.17: Instantaneous Mach number contours under $p$-refinement, for the unsteady laminar flow past a square cylinder, obtained with SD methods. $\Delta M = 0.05$.

Figure 8.17: Instantaneous Mach number contours under $p$-refinement, for the unsteady laminar flow past a square cylinder, obtained with SD methods. $\Delta M = 0.05$.

<table>
<thead>
<tr>
<th>p</th>
<th>Mach Number Contours</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><img src="image1.png" alt="Mach Number Contours p=1" /></td>
</tr>
<tr>
<td>2</td>
<td><img src="image2.png" alt="Mach Number Contours p=2" /></td>
</tr>
<tr>
<td>3</td>
<td><img src="image3.png" alt="Mach Number Contours p=3" /></td>
</tr>
<tr>
<td>4</td>
<td><img src="image4.png" alt="Mach Number Contours p=4" /></td>
</tr>
</tbody>
</table>

The drag coefficient $C_D^{RMS}$, and the RMS value of the lift coefficient $C_L^{RMS}$ are listed in Table 8.6. The Strouhal number associated to the frequency of the fluctuating lift coefficient $f_{C_L}$ is given by

$$St = \frac{f_{C_L} H}{|\bar{u}_\infty|}.$$  

In Table 8.6, the aerodynamic results presented in De Roeck et al. [144] are also listed for comparison. From this table, it can be seen that the accuracy of the results improve by increasing the order of the SD scheme. Moreover, although the grid is very coarse, with fifth-order SD scheme, the accuracy of the aerodynamic coefficients is comparable with that of the LES solutions.
Figure 8.18: Evolution of the lift $C_L$ and the drag $C_D$ coefficient in time, for the unsteady laminar flow past a square cylinder, obtained with fifth-order ($p = 4$) SD method.

Table 8.6: Effect of the $p$-refinement on the Strouhal number and the aerodynamic coefficients for the unsteady laminar flow past a square cylinder, obtained with SD methods. Comparison with DNS and LES reference solutions [144].

<table>
<thead>
<tr>
<th>Solution</th>
<th>$St$</th>
<th>$\langle C_D \rangle$</th>
<th>$C_{RMSError}^D$</th>
<th>$C_{RMSError}^L$</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNS [144]</td>
<td>0.146</td>
<td>1.55</td>
<td>0.019</td>
<td>0.232</td>
</tr>
<tr>
<td>Fine LES: non refl. BC</td>
<td>0.158</td>
<td>1.51</td>
<td>0.009</td>
<td>0.135</td>
</tr>
<tr>
<td>Coarse LES: non refl. BC</td>
<td>0.178</td>
<td>1.46</td>
<td>0.004</td>
<td>0.103</td>
</tr>
<tr>
<td>SD $p = 1$ (8,820 DOFs)</td>
<td>0.136</td>
<td>1.06</td>
<td>0.000</td>
<td>0.060</td>
</tr>
<tr>
<td>SD $p = 2$ (19,845 DOFs)</td>
<td>0.143</td>
<td>1.35</td>
<td>0.001</td>
<td>0.103</td>
</tr>
<tr>
<td>SD $p = 3$ (35,280 DOFs)</td>
<td>0.142</td>
<td>1.49</td>
<td>0.008</td>
<td>0.194</td>
</tr>
<tr>
<td>SD $p = 4$ (55,125 DOFs)</td>
<td>0.143</td>
<td>1.50</td>
<td>0.012</td>
<td>0.201</td>
</tr>
</tbody>
</table>
8.2. UNSTEADY LAMINAR FLOW SIMULATIONS

The total memory requirements for the present computations are summarized in Table 8.7. It should be noted that the required memory increases rapidly with the polynomial order $p$.

Table 8.7: Non-linear LU-SGS memory requirements [MB] for the unsteady laminar flow simulations past a square cylinder. SD method with $p$-refinement.

<table>
<thead>
<tr>
<th>SD method</th>
<th>Non-linear LU-SGS</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p = 1$ (8,820 DOFs)</td>
<td>23</td>
</tr>
<tr>
<td>$p = 2$ (19,845 DOFs)</td>
<td>49</td>
</tr>
<tr>
<td>$p = 3$ (35,280 DOFs)</td>
<td>117</td>
</tr>
<tr>
<td>$p = 4$ (55,125 DOFs)</td>
<td>319</td>
</tr>
</tbody>
</table>

8.2.3 Flow past a circular cylinder

The experimental work of Roshko [147] locates the beginning of the laminar-to-turbulent transition at Reynolds numbers $200 - 300$ based on the diameter of the cylinder $D$ and the module of the free-stream velocity vector $|\vec{u}_\infty|$. Beyond this Reynolds number but less than $3 \times 10^5$ the wake of the cylinder is completely turbulent and the boundary layer separation is laminar. It is known that the flow around the cylinder is 2D only when $Re < 200$. For larger Reynolds number, the vortex shedding is completely 3D. However, in this section the results obtained for the flow over a circular cylinder for Reynolds numbers $75, 150, 300, 800, 10^3$ are investigated using a 2D formulation with fourth-order ($p = 3$) SD scheme and quadratic boundary elements representation. The free-stream Mach number $M_\infty$ and the Prandtl $Pr$ are set to 0.05 and 0.72 respectively. The aim is to compare the present results with the experimental measurements of Wieselberger [192] and the 2D numerical solution of Henderson [72].

In Figure 8.19 the test problem is illustrated. At the left-hand-side boundary (the inflow) the flow is prescribed to be uniform and the same boundary conditions are applied to the upper and lower boundaries. At the right-hand-side boundary (the outflow) only the pressure is prescribed. At the cylinder wall, which is assumed to be adiabatic, the no-slip boundary condition is imposed. A mesh with 4,298 quadrilateral cells with a maximum aspect ratio $AR_{max}$ of 9 close to the circular cylinder was used for all computations. A detailed view of the grid is shown in Figure 8.20.
The time step $\Delta t$ started from 0.00025 and increased linearly till 0.25. However, when the flows reached a statistically steady state solutions (after the transient), the time step could be increased up to 0.65 without any stability problems. Nevertheless, for these calculations a time step of 0.25 was employed to guarantee about 150 samples for each period of the vortex shedding.
8.2. UNSTEADY LAMINAR FLOW SIMULATIONS

Figure 8.21: Instantaneous entropy contours, for the unsteady flow past a circular cylinder at different Reynolds numbers, obtained with fourth-order ($p = 3$) SD method.

(a) $Re = 300$.

(b) $Re = 800$.

(c) $Re = 10^3$. 

Figure 8.21: Instantaneous entropy contours, for the unsteady flow past a circular cylinder at different Reynolds numbers, obtained with fourth-order ($p = 3$) SD method.
In Figure 8.21 the instantaneous entropy contours are shown to give an impression of the flow field at three different Reynolds numbers, namely 300, 800 and $10^3$. In Table 8.8 the Strouhal numbers associated to the fluctuating lift coefficient are compared with the experimental results of Wieselberger [192]. From this table, it can be seen that the agreement between the present 2D simulations and the experimental values is good up to $Re = 300$. However, for higher Reynolds number the frequency of the vortex shedding is overestimated.

Table 8.8: Strouhal numbers for the unsteady flow past a circular cylinder at different Reynolds numbers, obtained with fourth-order ($p = 3$) SD method. Comparison with experimental measurements [192].

<table>
<thead>
<tr>
<th>$Re$</th>
<th>$St_{Exp. [192]}$</th>
<th>$St_{SD \ p = 3}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>75</td>
<td>0.148</td>
<td>0.148</td>
</tr>
<tr>
<td>150</td>
<td>0.180</td>
<td>0.181</td>
</tr>
<tr>
<td>300</td>
<td>0.200</td>
<td>0.201</td>
</tr>
<tr>
<td>800</td>
<td>0.204</td>
<td>0.233</td>
</tr>
<tr>
<td>1000</td>
<td>0.209</td>
<td>0.240</td>
</tr>
</tbody>
</table>

In Figure 8.22 the time averaged drag coefficient $\langle C_D \rangle$ of the circular cylinder is plotted in function of the Reynolds number. This figure shows that

Figure 8.22: Variation of the time averaged drag coefficient with Reynolds number for the flow past a circular cylinder, obtained with fourth-order ($p = 3$) SD method. Comparison with experimental measurements [147, 192] and the 2D numerical simulations [72].
8.3. LARGE EDDY SIMULATIONS

for $Re = 75$, the mean drag coefficient of the present computation matches very well with the experimental results of Wieselberger [192] and Roshko [147]. For higher Reynolds numbers, the mean drag coefficients of the 2D simulations overestimate the experimental ones but they agree very well with the 2D numerical results of Henderson [72], in which high-resolution computer simulations is used to quantify the change in drag around a 2D circular cylinder, for different Reynolds numbers. The reason is that for low Reynolds number ($Re \lesssim 100$), the 3D flow features are still weak and the vortex axes are nearly aligned with the cylinder axis in the near wake. For higher $Re$ the boundary layer is sharper, the velocity gradients are larger, resulting in the release of stronger vortices which are significantly distorted and possess components besides that in the span-wise direction and which cannot be taken into account by a 2D simulation.

The total memory requirement for the present computations is summarized in Table 8.9.

<table>
<thead>
<tr>
<th>SD method</th>
<th>Non-linear LU-SGS</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p = 3$ (68,768 DOFs)</td>
<td>228</td>
</tr>
</tbody>
</table>

Table 8.9: Non-linear LU-SGS memory requirements [MB] for the unsteady flow simulations past a circular cylinder.

8.3 Large eddy simulations

In this section, two 2D turbulent flows, namely the flow past a square cylinder at two different Reynolds numbers, and the 3D turbulent flow in a muffler are considered. These test cases are computed by solving the filtered compressible Navier-Stokes equations in combination with the wall-adapted local eddy-viscosity model (WALE), as discussed in Sections 3.2 and 3.2.2. However, since one property of a good LES model is that its use in a laminar or low Reynolds number flow results in a solution which is very close to the solution obtained by solving the Navier-Stokes equations [82], the sensitivity of the SD-LES method is first investigated by solving the 2D laminar flow around a NACA0012 airfoil.

Time marching scheme was done using the LU-SGS + BDF2 scheme. The non-linear system (5.16) was solved with a prescribed tolerance of $10^{-6}$ on the change of the $L_2$ norm of the solution variation $\Delta W_{cc}^{m+1}$ and/or a
maximum number of hundred SGS sweeps. During the calculations, the maximum number of symmetric Gauss-Seidel sweeps was never required. However, during the initial time steps (depending on the test case and the initial solution) the number of inner LU-SGS sweeps was between forty and forty-five. Afterwards, the number decreased and it reached a values which was between eight and twelve (depending again on the test case).

The 2D test cases were done with eight processors, while the 3D one was carried out using forty processors.

8.3.1 Flow around a NACA0012 airfoil

The compressible laminar flow simulation over a symmetric NACA0012 airfoil is conducted at zero angle of attack and free-stream Mach number of $M_\infty = 0.4$. The characteristic Reynolds number based on the chord $c$ and the module of the free-stream velocity $|\vec{u}_\infty|$ is specified as $Re = 5 \times 10^4$. The configuration of the test case is the same of that used in the previous NACA0012 test cases, i.e. Figure 8.1. Therefore, the airfoil is placed on the $\chi_1$ axis ($\chi_2 = 0$) of the computational domain. At the left-hand-side boundary (the inflow) the flow is prescribed to be uniform and the same boundary conditions are applied to the upper and lower boundaries. At the right-hand-side boundary (the outflow), sufficiently far from the airfoil, only the pressure is prescribed. At the airfoil wall, which is assumed to be adiabatic, the no-slip boundary condition is imposed.

The test case was solved using fourth-order ($p = 3$) SD scheme and a mesh with 19,874 quadrilateral cells and quadratic boundary elements. The maximum aspect ratio $AR_{\text{max}}$ of the first layer of the cells close to the airfoil is 30. The time step used for the computation started from 0.00001 and increased linearly till 0.00125. This time step allowed about 300 time samples per period of the vortex shedding.

The instantaneous entropy contour obtained using the WALE model is shown in Figure 8.23 to give an impression of the flow field around the airfoil. The flow around the body is almost symmetric on the top and the bottom sides of the airfoil and an unsteady wake develops downstream of the trailing edge.

The non-dimensional value of the frequency associated to the oscillating
3. LARGE EDDY SIMULATIONS

Figure 8.23: Instantaneous entropy contour for the unsteady laminar flow over a NACA0012 airfoil at zero angle of attack, obtained with fourth-order \( p = 3 \) SD-LES method.

The lift coefficient \( f_{CL} \), i.e. the reduced frequency

\[
f_{CL,r} = \frac{f_{CL} c}{c_\infty (1 - M_\infty^2)^{1/2}},
\]

where \( c_\infty \) is the free-stream speed of sound, is listed in Table 8.10. In this table the 2D DNS result of Sandberg et al. [153], where a fourth-order numerical scheme without upwinding, artificial dissipation or explicit filtering [154] is used, is also indicated. For DNS, a mesh with \( 243 \times 692 \) non-equidistantly spaced points in the tangential and in the normal direction with respect to the airfoil surface was employed. From Table 8.10 it can be seen that the reduced frequency is in good agreement with the DNS value for both computations. In fact, the error with and without subgrid-scale model is respectively 1.04% and 1.06%. This is a good achievement since the grid employed is very coarse compared to the grid used for a DNS computation. Moreover, as was expected, the subgrid-scale model does not affect the frequency of the vortex shedding when the laminar flow is well resolved.

To conclude this study, in Figure 8.24 the non-dimensional time averaged center-line \( (\chi_2 = 0) \) velocity component \( \langle \tilde{u}_1 \rangle / |\tilde{u}_\infty| \) is compared with the 2D DNS solution of Sandberg et al. [153]. It can be seen that, after an initial peak, \( \langle \tilde{u}_1 \rangle \) becomes negative before increasing to positive values further downstream, i.e. a reverse flow region is present. This reverse flow region is caused by flow oscillating around the trailing-edge corner at the wake frequency. Figure 8.24 shows also that the solutions obtained with
Table 8.10: Reduced frequencies for the unsteady laminar flow over a NACA0012 airfoil at zero angle of attack, obtained with fourth-order \((p = 3)\) SD and SD-LES methods. Comparison with DNS reference solution [153].

<table>
<thead>
<tr>
<th>Solution</th>
<th>(\bar{f}<em>{C</em>{L,r}})</th>
</tr>
</thead>
<tbody>
<tr>
<td>DNS [153]</td>
<td>6.580</td>
</tr>
<tr>
<td>SD no model</td>
<td>6.650</td>
</tr>
<tr>
<td>SD-LES</td>
<td>6.649</td>
</tr>
</tbody>
</table>

and without subgrid-scale model are almost indistinguishable. Therefore, the new coupling between the SD method and LES replicates the laminar flows obtained by solving the pure Navier-Stokes equations. Moreover, the comparisons show a good agreement between the present numerical results and the 2D DNS solution of Sandberg et al. [153].

![Time averaged center-line](image)

Figure 8.24: Time averaged center-line \((\chi_2 = 0)\) velocity profiles \(\langle \tilde{u}_1 \rangle / |\tilde{u}_\infty|\), for the unsteady laminar flow over a NACA0012 airfoil at zero angle of attack, obtained with fourth-order \((p = 3)\) SD and SD-LES methods. Comparison with DNS reference solution [153].

The total memory requirement for the present computations is summarized in Table 8.11. The SD-LES calculation required slightly more memory as compared to the calculation without subgrid-scale model because few more variable
8.3. LARGE EDDY SIMULATIONS

Table 8.11: Non-linear LU-SGS memory requirement [MB] for the unsteady flow simulations over a NACA0012 airfoil at zero angle of attack.

<table>
<thead>
<tr>
<th>SD method</th>
<th>Non-linear LU-SGS</th>
</tr>
</thead>
<tbody>
<tr>
<td>$p = 3$ (317,984 DOFs)</td>
<td>1,060</td>
</tr>
</tbody>
</table>

8.3.2 Flow around a square cylinder at $Re = 10^4$

The purpose of this test case is to compare the quality of the SD-LES approach with the 2D DNS solution presented by Wissink in [194]. In Figure 8.25, the configuration of the test case is illustrated. The cylinder is placed

![Figure 8.25: Configuration of the 2D square cylinder test case.](image)

... on the length axis ($\chi_2 = 0$) of the computational domain. At the left-hand-side boundary (the inflow) the flow is prescribed to be uniform. The same conditions are applied to the upper boundary and to the lower boundary. At the right-hand-side boundary, far enough from the cylinder, only the pressure is prescribed. For the cylinder wall, which is assumed to be adiabatic, the no-slip boundary condition is used.

The free-stream Mach number $M_\infty$ is set to 0.05, so that the flow is almost incompressible. The Prandtl number is set to the standard value for air, i.e. 0.72. The Reynolds number $Re$, based on the module of free-stream velocity $|\vec{u}_\infty|$ and the height of the square cylinder $H$, is $10^4$. The DNS of Wissink [194] was performed on a structured mesh with $400 \times 400$ grid points using a spatial discretization that consists of a combination of a seventh-order upwind-biased method for the convective terms and an...
The third-order (\( p = 2 \)) SD-LES method with an eighth-order central method for the diffusive terms. In the present work, the test case was solved using third-order (\( p = 2 \)) SD scheme with and without the subgrid-scale model. A mesh with 12,622 quadrilateral cells with a maximum aspect ratio \( AR_{\text{max}} \) of 33 close to the square cylinder was used. Notice that the total number of DOFs is 113,598 which is almost 71% of the DOFs used in [194]. The time step used for the computation started from 0.00001 and increased linearly up to 0.0025.

At the beginning of the simulation (starting with uniform flow field), rather small vortices of opposite sign are shed in parallel. However, this symmetric flow is not a stable solution at \( Re = 10^4 \). After breaking the symmetry, the flow behind the body behaves very chaotically and is fully turbulent. A lot of very small vortices and filaments appear just behind the cylinder as shown in Figure 8.26. Further away from the cylinder these small vortices merge to form larger vortical structures.

In Figure 8.27, the number of inner LU-SGS sweeps as a function of the time iteration is shown, for the calculation with subgrid-scale model. It can be seen that, the maximum number of SGS sweeps was never used. However, during the initial time steps the number of inner LU-SGS sweeps increased up to forty-six. Afterwards, it decreased to an averaged value of ten. For this reason, the evolution of the LU-SGS sweeps in Figure 8.27 is shown up to hundred and fifty time iterations. Note that, the transitional behavior in Figure 8.27 depends on the initial solution. In the present work, an uniform velocity field equals to the inlet boundary condition was imposed at the beginning of the computation.
After breaking the symmetry, the resolved turbulent properties were obtained by statistically averaging, for approximately 28 shedding cycles. In Figure 8.28, the non-dimensional time averaged velocity profiles $\langle \tilde{u}_1 \rangle / |\tilde{u}_\infty|$ at the upper side of the cylinder are plotted. The 2D DNS solution presented in Wissink [194] is also shown for comparison. It can be observed that, the SD method in combination with LES works very well and improves the accuracy of the results obtained without subgrid-scale model. The improvement is clearly visible at trailing edge of the cylinder (Figures 8.28(c) and 8.28(d)), where the LES solution captures quite well the variation of the velocity profiles close to the wall, whereas the solution obtained without subgrid-scale model differs from the DNS solution. Nevertheless, it is very important to note that the SD scheme is already able to predict all the features of the flows, showing the potential of high-order methods.

The $\langle \tilde{u}_1 \rangle / |\tilde{u}_\infty|$ profiles plotted in Figure 8.29 were gathered at various stations $\chi_1$ = constant behind the cylinder. This figure shows once more that the SD-LES simulation works well and improves the accuracy of the results obtained solving the pure Navier-Stokes equations. In fact, the velocity profiles obtained with subgrid-scale model agree well with those computed with the DNS.
Figure 8.28: Time averaged velocity profiles $\langle \tilde{u}_1 \rangle / |\tilde{u}_\infty|$ at various stations along the upper side of the square cylinder at $Re = 10^4$, obtained with third-order ($p = 2$) SD and SD-LES methods. Comparison with DNS reference solution [194].
8.3. LARGE EDDY SIMULATIONS

Figure 8.28: Time averaged velocity profiles $\langle \tilde{u}_1 \rangle / |\tilde{u}_\infty|$ at various stations along the upper side of the square cylinder at $Re = 10^4$, obtained with third-order ($p = 2$) SD and SD-LES methods. Comparison with DNS reference solution [194].
CHAPTER 8. APPLICATION II: SPECTRAL DIFFERENCE METHOD

Figure 8.29: Time averaged velocity profiles $\langle \tilde{u}_1 \rangle / |\vec{u}_\infty|$ at various stations through the turbulent wake of the square cylinder at $Re = 10^4$, obtained with third-order ($p = 2$) SD and SD-LES methods. Comparison with DNS reference solution [194].
In order to study the effect of the polynomial order on the LES solution, a simulation with second-order spatial discretization \((p = 1)\) on the same grid was also performed. Thus, the number of DOFs was reduced by a factor of 2.25. In Figures 8.30 and 8.31, the results obtained at three locations on the side of the square cylinder and at two locations downstream in the wake are shown. One can compare the solutions and see that the solution obtained with the second-order SD-LES scheme is much less accurate than the one obtained with the third-order SD-LES scheme. The second-order accurate solution captures the main flow features but the velocity profiles show a deviation from the DNS solution.

Figure 8.30: Time averaged velocity profiles \(\langle \tilde{u}_1 \rangle / |\tilde{u}_{\infty}|\) at two selected locations along the upper side of the square cylinder at \(Re = 10^4\), obtained with second-\((p = 1)\) and third-order \((p = 2)\) SD-LES method. Comparison with DNS reference solution [194].
The total memory requirement for the present computations is summarized in Table 8.12.

Table 8.12: Non-linear LU-SGS memory requirement [MB] for the turbulent flow simulations over a square cylinder at \( Re = 10^4 \).

<table>
<thead>
<tr>
<th>SD method</th>
<th>Non-linear LU-SGS</th>
</tr>
</thead>
<tbody>
<tr>
<td>( p = 2 ) (113,598 DOFs)</td>
<td>284</td>
</tr>
</tbody>
</table>

**Estimation of the SD-LES computational cost**

To conclude the study of this test case, a simple estimation of the Reynolds number scaling of the SD-LES computational cost is given. In Pope [132] it is shown that the computational cost for a DNS, i.e. the total number of modes to resolve the entire energy spectrum in 3D, is proportional to a power of the Reynolds number

\[
N^{mod,3D} \sim Re_L^{9/4},
\]

where \( Re_L \) is the Reynolds number based on the length scale \( L \) characterizing the large eddies. For the present test cases \( L \) can be chosen equal to the height of the square cylinder, i.e. \( H \). For a classical FV scheme,

\[
\langle \tilde{u}_1 \rangle / |\bar{u}_\infty|\text{ at two selected locations through the turbulent wake of the square cylinder at } Re = 10^4, \text{ obtained with second- } (p = 1) \text{ and third-order } (p = 2) \text{ SD-LES method. Comparison with DNS reference solution [194].}
\]
8.3. LARGE EDDY SIMULATIONS

\(N_{\text{mod,3D}}\) corresponds to the number of DOFs. Using the same relation presented above, the number of DOFs needed to perform a DNS with the SD scheme can be roughly estimated as

\[
N_{\text{cells,3D,SD-DNS}} \sim \frac{Re_L^{9/4}}{N_s (p, \text{dim})} \quad \text{dim} = 3, \quad N_{\text{cells,2D,SD-DNS}} \sim \frac{Re_L^{6/4}}{N_s (p, \text{dim})} \quad \text{dim} = 2.
\]

Consequently, for the flow past a square cylinder at \(Re = 10^4\), in order to perform a 2D DNS with a third-order SD scheme, \(3.7 \times 10^4\) cells are needed. Hence, the averaged CPU-time per iteration, to perform the 2D DNS simulation can be computed as

\[
CPU_{\text{iter,SD-DNS}} = \frac{N_{\text{cells,2D,SD-DNS}}}{N_{\text{cells,2D,LES}}} \frac{CPU_{\text{iter,SD-no model}}}{CPU_{\text{iter,SD-LES}}},
\]

where \(CPU_{\text{iter,SD-no model}}\) and \(CPU_{\text{iter,SD-LES}}\) are the averaged CPU-times per time iteration required by the calculations without and with subgrid-scale model. For the present test case, \(CPU_{\text{iter,SD-DNS}}\) is about 2.5. Therefore, the SD-LES simulation is almost 2.5 times cheaper than the SD-DNS.

As mentioned in Section 3.2, in this work the LES approach is used as a cheaper alternative to DNS. However, for sake of consistency, one should compare the LES cost with that of a RANS calculation, which represents the cheapest approach to numerically simulate the behavior of unsteady turbulent flows. With the RANS approach, the Navier-Stokes equations are ensemble-averaged, converting turbulent fluctuations into Reynolds stresses, while leaving the large scale, rotational motions to be resolved as unsteady phenomena. The LES approach, on the other hand, employs a spatial averaging over a scale sufficient to remove scales not resolved by the particular grid being used. The subgrid scale turbulence is then modeled. A practical difference is in the degree of mesh resolution required: LES resolves the larger eddies of the turbulence itself, whereas the unsteady RANS approach models the turbulence and resolves only unsteady, mean flow structures - primarily larger than the turbulent eddies. Consequently, LES typically requires much higher grid resolution, at least locally, and is therefore more costly. In addition, LES resolves the complete range of scales of random motion, up to the cut-off frequency, while unsteady RANS aims to capture a single frequency (e.g. corresponding to coherent shedding) and to model the random motions using standard turbulence closures. Therefore, LES requires very long integration time with a small time step to build a statistically-averaged solution; on the
CHAPTER 8. APPLICATION II: SPECTRAL DIFFERENCE METHOD

other hand, a few shedding periods are usually enough to obtain accurate phase-averaged solution with RANS, thus limiting its overall computational cost. Because of the grid resolution and the long integration time, an LES simulation may be two or more orders of magnitude slower as compared to a RANS simulation. However, one should take into account that the classical RANS approaches are usually not able to distinguish between quasi-periodic large scale and turbulent chaotic small scale features of the flow field. This leads to huge problems when the flow field is governed by both phenomena. A typical representative is a bluff-body flow. Generally, the RANS approach is not able to reproduce the unsteady characteristics of the flow field reasonably, resulting in an inadequate description of unsteady phenomena, such as vortex formation and shedding behind bluff or inclined bodies [81].

8.3.3 Flow around a square cylinder at $Re = 2.2 \times 10^4$

The purpose of this test case is to evaluate the quality of the SD-LES approach by comparing the mean flow field and the turbulent kinetic energy results with some reference solutions. In literature 2D and 3D large eddy simulations of the flow past a square cylinder have been performed, for instance see Murakami et al. [114] and Rodi [142]. Moreover, Breuer and Pourquie [26] and Bouris et al. [23] also performed 2D and 3D LES computations and compared the results with the experimental measurements of Durão et al. [47] and Lyn et al. [106]. It was demonstrated that the characteristic of this type of flow is its quasi-two-dimensional character and the presence of periodic vortex shedding from the front corners of the square rod which introduces a low-frequency variation of the velocity field behind the rod in addition to the high-frequency turbulence fluctuations. In Breuer and Pourquie [26] and Bouris et al. [23], it has been stated that 2D LES calculations are clearly inferior to 3D ones since certain important features of 3D turbulence are not resolved. The three dimensionality of turbulence cannot be questioned, however, in the present work, we want to show that the importance of detailed simulation of the quasi-two-dimensional mechanisms can be achieve performing a 2D LES with the new combination of SD and subgrid-scale model.

The same domain, boundary conditions and grid used for the previous test case were employed. The free-stream Mach number $M_\infty$ is set to 0.05 so that the flow is again almost incompressible. The Reynolds number $Re$, based on the module of the free-stream velocity $|\bar{u}_\infty|$ and height of the square cylinder $H$, is $2.2 \times 10^4$. The flow was computed using third-order
(\(p = 2\)) SD scheme with and without the subgrid-scale model. The time step used for the computation started from 0.00001 and increased linearly up to 0.002. Figure 8.32 shows the number of inner LU-SGS sweeps in function of the time iteration. One can see that the maximum number of symmetric Gauss-Seidel sweeps was never reached. However, during the initial steps the number of inner LU-SGS sweeps increased up to forty five. Afterwards, the number of sweeps decreased to nine and it oscillated around this value. For this reason, the evolution of the LU-SGS sweeps in Figure 8.32 is shown up to hundred and fifty time iterations. Note once more, that the transitional behavior in Figure 8.32 depends on the initial solution. A uniform velocity field equals to the inlet boundary condition was imposed at the beginning of the computation.

Figure 8.32: Number of inner LU-SGS sweeps as a function of the time iteration, for the turbulent flow past a square cylinder at \(Re = 2.2 \times 10^4\). Third-order \((p = 2)\) SD-LES method.

In the present test case, the 2D LES calculation with a second-order finite volume scheme and the Smagorinsky-Lilly model of Bouris et al. [23] is used as reference 2D LES solution. In the latter work, a mesh with \(350 \times 300\) points is employed. Therefore, the total number of DOFs of the present SD calculation is practically the same as compared to the number of DOFs used in Bouris et al. [23].

As for the previous calculation, inlet conditions are based on a uniform flow with non-fluctuating velocity profiles. Table 8.13 shows the present predictions of the dominant vortex shedding frequency in non-dimensional form \(St\) and the time averaged drag coefficient \(\langle C_D \rangle\), as well as the results.
reported in Bouris et al. [23]. In this table, the abbreviation \((k - \varepsilon)^*\), RSE1 and RSE2 stand, respectively, for \(k - \varepsilon\) model with 2 layers, Reynolds stress equation with wall function and Reynolds stress equation with two layers [49, 142]. The results of the SD-LES are in good agreement with the experimental measurements while the solution without the subgrid-scale model underestimates both the Strouhal number \(St\) and the time averaged drag coefficient \(\langle C_D \rangle\).

Table 8.13: Strouhal numbers and mean drag coefficients, for the turbulent flow past a square cylinder at \(Re = 2.2 \times 10^4\), obtained with third-order \((p = 2)\) SD and SD-LES methods. Comparison with other numerical simulations [23] and experimental measurements [47, 106].

<table>
<thead>
<tr>
<th>((k - \varepsilon)^*)</th>
<th>RSE1</th>
<th>RSE2</th>
<th>2D LES [23]</th>
<th>SD</th>
<th>SD-LES</th>
<th>Exps.</th>
</tr>
</thead>
<tbody>
<tr>
<td>(St)</td>
<td>0.124</td>
<td>0.136</td>
<td>0.159</td>
<td>0.134</td>
<td>0.121</td>
<td>0.133</td>
</tr>
<tr>
<td>(\langle C_D \rangle)</td>
<td>1.179</td>
<td>2.15</td>
<td>2.43</td>
<td>2.18</td>
<td>1.98</td>
<td>2.21</td>
</tr>
</tbody>
</table>

Time averaged results were obtained integrating the data over approximately 20 shedding cycles and the mean center-line velocity is presented in Figure 8.33. Note that, the horizontal normalized coordinate is denoted by \(\chi'_1\) and corresponds to the coordinate \(\chi_1\) shifted by \(H/2\) so that \(\chi'_1 = \chi_1 + H/2\). Although the SD scheme without subgrid-scale model overpredicts the value of the reverse velocity, further downstream it gives a solution which is close to the 2D LES solution of Bouris et al. [23]. Moreover, it can be clearly seen that the modeling of the subgrid-scale stress tensor improves the accuracy of the results. In fact, good agreement between the predicted results and experimental data of Durão et al. [47] throughout the comparison domain shows that the quality of a high-order SD method increases when it is coupled with large eddy simulation. In the other parts of the comparison domain the method without subgrid-scale model is in a good agreement with LES of Bouris et al. [23]. Note that, the SD-LES method captures the peak of the mean stream-wise velocity considerably better than the others.

Figure 8.34 shows the 2D resolved total turbulent kinetic energy defined by

\[
k_{tot} = \frac{1}{2} \left( \langle u_1'^2 \rangle + \langle u_2'^2 \rangle \right),
\]  

(8.3)
8.3. LARGE EDDY SIMULATIONS

Figure 8.33: Time averaged center-line ($\chi_2 = 0$) velocity profiles $\langle \tilde{u}_1 \rangle / |\tilde{u}_\infty|$ behind the square cylinder at $Re = 2.2 \times 10^4$, obtained with third-order ($p = 3$) SD and SD-LES methods. Comparison with other numerical simulations [23] and experimental measurements [47, 106].

Figure 8.34: Total turbulent kinetic energy $k_{tot} / |\tilde{u}_\infty|^2$ along the center-line ($\chi_2 = 0$) behind the square cylinder at $Re = 2.2 \times 10^4$, obtained with third-order ($p = 3$) SD and SD-LES methods. Comparison with other numerical simulations [23] and experimental measurements [47, 106].
where $\langle u'_1 \rangle$ and $\langle u'_2 \rangle$ are the time averaged fluctuations in $\chi_1$ and $\chi_2$ directions respectively. One can compare the results with the experimental data and see that the SD-LES method undershoots the peak experimental value somewhat less than the 2D LES of Bouris et al. [23] and predicts the results well elsewhere. The approach without subgrid-scale model overshoots the peak experimental value less than does the 2D LES of Bouris et al. [23], whereas throughout most of the domain both approaches are on the top of each other.

Overall, the results indicate that although the SD method without subgrid-scale model is already able to predict all the features of the flow, the SD-LES approach gives better results, for the stream-wise mean velocity and the total kinetic energy.

8.3.4 Flow in a muffler

Mufflers are commonly used in a wide variety of applications. Industrial flow ducts as well as internal combustion engines frequently make use of silencing elements to attenuate the noise levels carried by the fluids and radiated to the outside atmosphere by the exhausts. Restrictive environmental legislation requires that silencer designers use high performance and reliable techniques. Various techniques are currently available for the modeling and testing of duct mufflers. Empirical, analytical and numerical techniques have been used and proven reliable under controlled conditions. Design of a complete muffler system is, usually, a very complex task. Each element is selected by considering its particular performance, cost and its interaction effects on the overall system performance and reliability.

The main purpose of this section is to evaluate the accuracy and the reliability of the implicit SD-LES solver by performing the simulation of a confined flow in an industrial geometry. The 3D turbulent flow in a muffler is considered as a test case. The results are compared with the particle image velocimetry (PIV) measurement performed by Dr. Michael Bilka at the Department of Environmental and Applied Fluid Dynamics of the von Karman Institute for Fluid Dynamics [22]. In Figure 8.35, the geometry of the muffler and its characteristic dimensions are illustrated, where the flow is from left to right. This configuration was selected because is representative of the physical mechanisms involved in a generic muffler and it is a benchmark test case of the IWT Project SBO 050163 ("Simulation and design tools towards the reduction of aerodynamic noise in confined flows").
8.3. LARGE EDDY SIMULATIONS

Figure 8.35: Configuration of the 3D muffler test case.

At the inlet, mass density and velocity profiles are imposed. The inlet velocity profile in the $\chi_3$ direction is given by

$$u_3 = u_{\text{max}} \left\{ \frac{1}{2} - \frac{1}{2} \tanh \left[ 2.2 \left( \frac{r}{d/2} - \frac{d/2}{r} \right) \right] \right\}.$$ 

At the outlet only the pressure is prescribed. In accordance with the experiments, the inlet Mach number $M_{\text{inlet}}$ and the Reynolds number, based on maximum velocity at the inlet $u_{\text{max}}$ and the diameter of the inlet/outlet $d$, are set respectively to $4.64 \times 10^4$ and 0.05.

The flow is computed using second-order ($p = 1$) SD scheme on two grids with 195,072 and 337,505 hexahedral elements respectively. Therefore, the total number of DOFs is approximately 1.6 and 2.7 millions. The maximum aspect ratio is 22 for the coarse mesh and 15 for the other one. The fine grids was obtained by refining the coarse one mainly in the expansion chamber, where the flow gets turbulent. In Figure 8.36 the coarsest mesh is shown. The time-step used for the computations started from 0.00001 and increased linearly up to 0.001. Notice that, although the Mach number is rather small, no specific difficulties of convergence and/or accuracy were observed.

The computation is validated on the center plane of the expansion coinciding with the center planes of the inlet and outlet pipes. The laser sheet was generated by a 200 mJ Nd-YAG double-pulsed laser passing through a spherical and cylindrical lens. A smoke generator was used to see the flow with fine oil droplets (1 $\mu$m). A digital CCD camera was used (PCO Sensicam, resolution - $1280 \times 1024$) to record the images with a region of interest limited to $1280 \times 832$. The pulse separation of the between the images of
the PIV couple correspond to a maximum displacement of 6 pixels. In order to keep a high spatial resolution over the measurement domain, the measurements are divided into 3 zones and with a field of view of $100 \times 55 \, mm^2$ with a resolution of 12 pixels per millimeter. All of these measurements are taken on the symmetrical center plane of the muffler. It should be noted that the circular nature of the geometry acts as a lens causing a change in magnification which prevents from capturing images close to the wall. It is found that outside 1 cm from the wall the magnification effect is negligible and as the mean flow direction is in the direction of constant magnification (i.e along the height of the cylinder) no corrections are deemed necessary. For each region, 1,000 image pairs have been acquired and pre-processed with the home-made cross-correlation algorithm WIDIM (Window Displacement Iterative Multigrid). This program is based on an iterative multigrid predictor-corrector method, handling the window distortion, for better resolution of shear flows, and the sub-pixel window displacement, to limit pixel-locking. The predictor-corrector method is then validated for each grid size if the signal-to-noise (SN) ratio is above 1.5.
Figure 8.37: Contour of the time averaged velocity vector magnitude for the turbulent flow in a muffler, obtained with second-order \((p = 1)\) SD-LES method. Grid with 337,505 hexahedral cells and \(AR_{max} = 12\).

[156]. Mean quantities are then computed from the PIV data and profiles are extracted to compare with the LES.

Figure 8.37 shows the contour of the time averaged velocity vector magnitude in the symmetry plane obtained with second-order \((p = 1)\) SD-LES method on the grid with 337,505 hexahedral cells. In Figure 8.38, the non-dimensional mean velocity profile in the axial direction \(\langle u_3 \rangle/u_{max}\) is shown for six different cross sections in the expansion chamber, where the PIV measurements were done. In this figure, the PIV data are also plotted for comparison. Figure 8.39 shows the non-dimensional Reynolds stress \(\langle u_2' u_3' \rangle/u_{max}^2\) at the same cross sections. It can be seen that the mean velocity profiles computed with the SD-LES approach, are in good agreement with the experiments throughout the comparison domain for both grids. The effect of the mesh refinement is clearly visible for the Reynolds stresses. In fact, although the computation with the coarse mesh captures reasonably well the two peaks in the shear layer, it underestimates widely the Reynolds stress away from the ‘jet’ region. On the contrary, the calculation with the fine grid gives good results throughout the domain of interest. This is a good achievement since the number of DOFs for the fine mesh is only 2.7 millions.
Figure 8.38: Time-averaged velocity profile in the axial direction at six cross sections in the expansion chamber \( \langle \tilde{u}_3 \rangle / u_{max} \) vs. \( \chi_2 / d \), for the turbulent flow in a muffler, obtained with second-order \( (p = 1) \) SD-LES method on coarse and fine grids. Comparison with experimental measurements (PIV) [22].
8.3. LARGE EDDY SIMULATIONS

Figure 8.39: Reynolds stress at six cross sections in the expansion chamber \((\langle u'_2 u'_3 \rangle / u_{max}^2 \text{ vs. } \chi_2/d)\), for the turbulent flow in a muffler, obtained with second-order \((p = 1)\) SD-LES method on coarse coarse and fine grids. Comparison with experimental measurements (PIV) [22].
CHAPTER 8.  APPLICATION II: SPECTRAL DIFFERENCE METHOD

The total memory requirement for the present computations are summarized in Table 8.14. This table clearly illustrates that, although the simulations were done with a second-order SD-LES scheme, a large amount of memory is required.

Table 8.14: Non-linear LU-SGS memory requirements [MB] for the turbulent flow simulations in a muffler. Second-order \((p = 1)\) SD method with grid refinement.

<table>
<thead>
<tr>
<th>Grid</th>
<th>Non-linear LU-SGS</th>
</tr>
</thead>
<tbody>
<tr>
<td>195,072 hexahedral cells (1,560,576 DOFs)</td>
<td>10,831</td>
</tr>
<tr>
<td>337,505 hexahedral cells (2,700,040 DOFs)</td>
<td>18,740</td>
</tr>
</tbody>
</table>
Chapter 9

Conclusions and future directions

The present thesis is focused on the development of an efficient Navier-Stokes (N-S) / large eddy simulation solver on unstructured grids for high-order accurate schemes. Two spatially high-order accurate methods for unstructured grids have been considered, namely the spectral volume (SV) and the spectral difference (SD) methods. To fully exploit the potential benefits of such high-order spatial accurate operators, an efficient non-linear algebraic solver, denoted as non-linear LU-SGS algorithm, has been investigated. This solver has been used to invert the non-linear algebraic systems arising from implicit time discretizations with the well-known backward Euler (BE) scheme and the second-order backward difference formula (BDF2). The evaluation of the solver performances, both with analysis and computation, have been two main cores of this thesis.

The need for paying particular attention to subgrid models for the simulation of general turbulent has been recognized. In this framework, a high-order SD scheme has been coupled with the local eddy-viscosity (WALE) model to perform large eddy simulations and a new procedure to calculate the grid filter width has been proposed. The accuracy and the reliability of the implicit SD solver have been tested by solving two-dimensional (2D) and three-dimensional (3D) turbulent test cases.
CHAPTER 9. CONCLUSIONS AND FUTURE DIRECTIONS

9.1 Achievements

The damping properties of the non-linear LU-SGS algorithm combined with the BE scheme (LU-SGS + BE) have been analyzed by means of a Von Neumann analysis for a model 2D linear advection. The analysis has been performed with a CFL number of one million, on a mesh with cell aspect ratio of hundred. The latter choice has enabled to account for the effects of the geometrical stiffness imposed by the Navier-Stokes grids where high-aspect ratios occur near walls. The most important results of this analysis can be summarized as follows:

- The LU-SGS + BE scheme is always stable for any choice of the convective velocity direction and the solution orientation for second- to fourth-order SV and SD methods.

- Five/six symmetric Gauss-Seidel (SGS) sweeps are in general sufficient to get a good damping of high-frequency error components.

These results have been used to simulate some 2D steady compressible flows with the 2D-SV-TRI in-house code. 2D-SV-TRI is a 2D SV-based code which was developed at the Vrije Universiteit Brussel, Department of Mechanical Engineering. In combination with a full multigrid (FMG) V-cycle $p$-multigrid algorithm, the LU-SGS + BE scheme was used to compute the 2D steady laminar flow past a circular cylinder at $Re = 40$ and different Mach numbers and the flow over a NACA0012 airfoil at $Re = 5 \times 10^3$ and $M_\infty = 0.5$. A steady laminar flow in a channel with a backward-facing step at $Re = 800$ and $M_{inlet} = 0.2$ was also considered. In order to assess the convergence properties of the LU-SGS + BE scheme, the performance of the latter scheme has been compared with that of a family of explicit Runge-Kutta (E-RK) smoothers available in the 2D-SV-TRI code. The most important results of this study can be summarized as follows:

- The convergence rate of the LU-SGS + BE scheme is not greatly affected by the cell aspect ratio of the mesh.

- The convergence rate of the LU-SGS + BE scheme is slightly affected by the low Mach number when meshes with isotropic cells are used.

- Effects of the low Mach number on the convergence rate of the LU-SGS + BE scheme are visible when meshes with anisotropic cells and high-aspect ratio are used.
• For second-order SV method, the LU-SGS + BE scheme reduces the computational time by a factor of up to $5 - 100$ (depending on the aspect ratio of the mesh) as compared to a well tuned E-RK scheme.

• For third- and fourth-order SV methods, the LU-SGS + BE scheme reduces the computational time by 2 orders of magnitude as compared to a well tuned E-RK scheme.

• The convergence rate of the LU-SGS + BE for the NACA0012 airfoil simulation is comparable with that of a very efficient reference implicit scheme proposed by Swanson, Turkel and Rossow [169].

The coupling between the SV method and the non-linear LU-SGS algorithm has been limited to 2D steady flow problems because no stable high-order 3D SV partition has been found to date.

The non-linear LU-SGS solver has also been implemented in a C++ code, namely the COOLFluiD collaborative simulation environment developed at the Von Karmann Institute for Fluid Dynamics. The algorithm has been combined with the BE scheme and the BDF2, for a SD method. The LU-SGS + BE scheme has been tested by solving 2D steady laminar flow over a NACA0012 airfoil at $Re = 5 \times 10^3$ and $M_\infty = 0.5$ and the 3D steady laminar flow through a $90^\circ$ bending square duct. In order to assess the convergence properties of the LU-SGS + BE scheme, a Newton-Raphson GMRES algebraic solver, has been used as reference efficient algebraic solver. A study of the methods performances has shown that:

• In term of wall time LU-SGS + BE scheme for high-order SD discretizations performs better than the Newton-Raphson GMRES solver with the BE scheme.

• The non-linear LU-SGS algorithm needs far less memory than the Newton-Raphson GMRES method, however its application to 3D flow problems still requires a large amount of memory, when high-order polynomial reconstructions are used.

To demonstrate the capabilities of the SD method in combination with the non-linear LU-SGS solver for the solution of unsteady flow problems, the following 2D test cases have been investigated: flow over an open cavity at $Re = 1.5 \times 10^3$ and $M_{inlet} = 0.15$, flow past a square cylinder at $Re = 200$ and $M_\infty = 0.5$, and flow past a circular cylinder at $Re = 75, 150, 300, 800, 10^3$. 171
CHAPTER 9. CONCLUSIONS AND FUTURE DIRECTIONS

Time marching was done with the BDF2. The main conclusions of this study can be summarized as follows:

- For a given coarse grid the accuracy of the flow solution improves by increasing the order of the SD scheme.
- For each test case, the solution computed with highest order polynomial interpolation agrees well with experimental data and/or reference numerical solutions available in literature.
- Very good results can be achieved using a number of degrees of freedom which is significantly lower than that of numerical solutions computed with classical spatial discretizations, e.g. finite volume (FV) or finite difference (FD).

An important achievement of the present work is the coupling of a high-order SD scheme with the WALE model (SD-LES) to perform large eddy simulations. The accuracy and the reliability of the SD-LES approach have been tested by solving the 2D flow over a NACA0012 airfoil at $Re = 5 \times 10^5$, $M = 0.4$ and zero degree angle of attack, the 2D turbulent flow past a square cylinder at $Re = 10^4$ and $Re = 2.2 \times 10^4$ and $M_\infty = 0.05$, and the 3D turbulent flow in a muffler at $Re = 4.64 \times 10^4$ and $M_{inlet} = 0.05$. Simulations without subgrid-scale model were also performed for the 2D test cases. Time marching was done with BDF2 in combination with the nonlinear LU-SGS solver. The main results of this analysis can be summarized as follows:

- The subgrid-scale model does not affect the numerical solution when the laminar flow is well resolved, i.e. the SD-LES computation replicates the laminar flow obtained by solving the pure Navier-Stokes equations.
- No specific difficulties of convergence have been observed for low Mach number flows.
- During the initial time steps the number of inner LU-SGS sweeps oscillates between forty and fifty; afterwards, it decreases to an averaged value of about ten.
- Although the SD method without subgrid-scale model is already able to predict all the features of the flows, the SD-LES approach is more accurate.
9.2. **FUTURE WORK**

- An estimation of the SD-LES computational cost has shown that the new approach is about 2.5 times faster than a DNS performed with the same SD scheme.

- The SD-LES results agree well with numerical or experimental data reported in literature.

## 9.2 Future work

With this work, the initial development of an efficient Navier-Stokes/LES solver on unstructured grids for high-order accurate schemes was established. The capabilities and the advantages of the current available tool have been clearly illustrated by its application to different types of fluid flow problems. However, more research and development are needed in order to make this solver - and in general high-order solvers - a valuable tool for the industry. A personal vision and an overview of the main tasks and issues that need to be solve for the application of any high-order methods for unstructured grids to practical applications is given below.

### 9.2.1 Compact high-order accurate spatial methods

The stability problems that occur with high-order SV schemes for tetrahedral cells, is the most critical issue that remains to be resolved for the usability of the SV method. Several new SV partitions were proposed which have a reduced maximum real part of the Fourier footprint by up to 20% over the original un-optimized partition reported in literature. Numerical simulations showed that the strength of the instability was weakened by about an order of magnitude for some cases by employing a constrained minimization approach. However no fully stable third-order SV scheme for tetrahedral cells has been found to date. Consequently, high-order SV schemes for tetrahedral cells are not yet robust. Therefore, they can not be routinely employed in practical calculations with complicated geometries and complex physics.

Although the SD schemes for quadrilateral and hexahedral cells are very promising, a crucial issue for the application of the SD method to real complex geometries and industrial flow problems is the solution of the stability problems with high-order SD schemes for triangular and tetrahedral cells. In fact, only if this issue will be solved, high-order SD schemes could be used in modern applications where unstructured *hybrid grids* are required.
CHAPTER 9. CONCLUSIONS AND FUTURE DIRECTIONS

for efficient automated grid generations of complex geometries. Nevertheless, the flexibility of the SD method could already be improved by extending the scheme’s formulation to more general quadrilateral and hexahedral cells, where hanging nodes are allowed. Although several algorithms have been developed for the generation of pure hexahedral meshes for complex geometries, see for instance Yasushi et al. [197], hanging nodes are in fact absolutely necessary for an efficient automated grid generation for general geometries, when hybrid grids can not be used.

Recently, Huynh [79, 80] developed a new high-order formulation for 1D conservation laws based on the idea of flux reconstruction. The formulation was capable of unifying several high-order methods including the DG method and the SV and SD methods into a single family. The idea of flux reconstruction was generalized into a lifting collocation penalty (LCP) approach on hybrid grids by Wang and Gao [52, 184] and Haga et al. [61], respectively for 2D Euler and 3D Navier-Stokes equations. The LCP method has a nodal formulation, with a cell-wise discontinuous polynomial solution space. The solution is interpolated from the solution at a set of solution points. In addition, a new flux polynomial is reconstructed. The LCP method does not require the evaluation of any integrals. With a careful selection of the solution and flux points, the approach can be made simple and efficient to implement for hybrid grids. For these reasons the LCP approach is a very promising method, which could be used as a valid alternative to the DG method and the SV and SD methods.

Although LES is by now a well researched approach for spatially low-order accurate methods, specific studies on the capabilities of their high-order counterparts to resolve accurately the small scales with an unresolved DNS type approach or a LES approach are necessary. In this context, it would be very interesting to investigate in more detail the computational cost of SD schemes to perform DNS, unresolved DNS and LES. In order to achieve that, an analysis of the number of Fourier modes in each direction for adequate resolution of isotropic turbulence could be performed, as shown by Pope [132]. These studies should be carried out considering the dispersion and the dissipative properties of high-order schemes. These results are expected to give a better understanding of the capabilities of spatially high-order schemes for the simulation of turbulent flows. In addition, the outcome of this analysis will also be useful in mesh generation with high-order schemes. Further investigation of the subgrid scale models is also necessary.
9.2. FUTURE WORK

Recently, Zhou and Wang [201] applied a SD method in combination with a perfectly matched layer (PML) to the simulation of CAA benchmark problems, while the author of this thesis has exploited the strength of a high-order spectral difference method to provide the acoustic sources for aerodynamic sound field simulations with a Ffowcs-Williams Hawking (FW-H) approach. However, with the aim of applying high-order methods to computational aeracoustics (CAA), further investigation of the capabilities of these schemes to simulate CAA problems or to provide acoustic sources for aerodynamic sound field simulation is also very important.

9.2.2 Time integration/solution iterative approaches

The spatial discretization of nonlinear hyperbolic PDE to high order of accuracy on unstructured meshes can be accomplished by local discretization methods, such as the DG method, the SV and the SD methods or the new promising LCP approach. While a high-order spatial discretization enables one to achieve better resolution with less degrees of freedom as compared to current state-of-the-art second-order methods, the overall efficiency and robustness of a numerical scheme for large-scale applications depends on the solution methodology for the (non-linear) system of equations arising from the discretization. High-order spatial discretization operators are usually much stiffer than their low-order counterparts. Classical explicit time marching algorithms, such as explicit Runge-Kutta schemes, have an upper limit for the time step that can be taken for stability reasons. Such classical algorithms can be very inefficient in combination with high-order spatial schemes, where the maximum time step tends to be very small. Consequently, high-order spatial discretizations necessitate the use of an implicit method for robust simulations at a reasonable cost. Several implicit time integration/solution iterative approaches have been used in literature, e.g. the preconditioned Newton-Raphson GMRES solver, the non-linear LU-SGS algorithm, the line-implicit algorithm, to list a few.

In this thesis, the non-linear LU-SGS algorithm, in combination with two implicit time discretization schemes, has been investigated extensively. This algebraic solver proved to be very efficient and robust. However, additional studies can be performed, for instance applying the Fourier analysis to the discretized form of the linearized time-dependent Euler equations when solved with the LU-SGS + BE scheme. The outcome of this analysis is expected to give additional information on the properties of the time marching/iterative technique. Moreover, specific studies on the capabil-
CHAPTER 9. CONCLUSIONS AND FUTURE DIRECTIONS

ities of the implicit solver to deal with high Reynolds number and very large aspect ratio are necessary.

Although non-linear LU-SGS algorithm requires significantly less memory than the classical methods that use the full Jacobian matrix (for instance the Newton-Raphson GMRES algorithm), the required amount still increases with order of accuracy to the power six, for 3D computations. This large memory requirement can be a serious limitation for the applicability of high-order methods to very large 3D problems. The key issue is then to develop an algebraic solver that requires a small amount of memory and is able to solve efficiently the algebraic non-linear systems. Good candidates may be matrix-free Newton-Krylov methods and the more recently developed RK3/Implicit Residual scheme which was proposed by Swanson, Turkel and Rossow [169]. With the latter approach, the preconditioner would be based on a first-order upwind scheme instead of high-order spatial discretization operators. This way the complete Jacobian with no simplifications could be easily calculated on the fly and requires no storage. In addition, these small calculations could be accelerated by using a graphics processing unit (GPU). For these reasons, the RK3/Implicit Residual scheme is a very promising method, which is certainly worth further investigation in combination with spatially high-order compact schemes. Geometric and $p$-multigrid methods are also very promising acceleration approaches and can be used to exploit the strength of the algebraic solvers. These two techniques are essential to perform both steady and unsteady simulations at a reasonable cost for industrial flow applications.

A last topic of future interesting research coming to mind is the aspect of implicit high-order accurate temporal methods. Since time accuracy is also an important aspect in unsteady numerical simulations, it would be very interesting to investigate the performances of these implicit schemes in combination with compact high-order accurate spatial methods. Successful application of them to the simulation of compressible turbulence, vortex dominated flows and aeroacoustics is expected to have an important impact on the usability of high-order solvers for unstructured grids. The fourth-order explicit-first-stage singly diagonally implicit Runge-Kutta (ES-DIRK4) scheme or the fourth-order modified extended backward differentiation formula (MEBDF4) may be good starting candidates. It is very important to point out that although research in this direction is needed to achieve the full potential of high-order schemes, it already appears that algebraic solver technology is again the area that needs significant attention.
Appendix A

Time integration methods for space-discretized equations

In this appendix, a method to analyze the linear stability for the combination of a space and a time discretization will be discussed. It is based on the method of lines (MOL) introduced in the early 1960s by Sarmin and Chudov [155]. The MOL most often refers to the construction or analysis of numerical methods for partial differential equations that proceeds by first discretizing the spatial derivatives only and leaving the time variable continuous. This leads to a system of ordinary differential equations to which a numerical method for initial value ordinary equations can be applied. This way, the analysis of the space discretization and time integration can be performed separately with a Von Neumann type of approach. This procedure allows to answer to the fundamental question on the criteria to be satisfied by a time integration method, applied to a given space discretization, so that it leads to a stable fully discretization. It can be applied to any system of linear PDEs. In the present appendix, this procedure is applied to 2D linear advection equation.

The methodology to analyze the stability properties and the amplification matrix of the non-linear LU-SGS algorithm in combination with the backward Euler difference is also discussed. The method corresponds to an application of the Von Neumann analysis and uses the data obtained from the stability analysis of the spatial discretization.
More information on the numerical solution of partial differential equations, including other methods for the stability analysis of spatial and temporal schemes can be found in Isaacson and Keller [83], Smith [160] and Hirsch [75].

### A.1 Stability of spatial discretizations

Consider the linear advection equation, discussed in Section 3.5, and given by

\[ \frac{\partial w}{\partial t} + \vec{\nabla} \cdot (\vec{a} w) = 0. \]  

(A.1)

The initial solution \((t = 0)\) is a spatial Fourier wave defined by

\[ W(\vec{\chi}, 0) = W^0(\vec{\chi}) = A^0 e^{i\vec{k} \cdot \vec{\chi}} \]  

(A.2)

where \(\vec{\chi}\) is the position vector, \(I \equiv \sqrt{-1}\) is the imaginary unit number and \(\vec{k}\) is the wave vector given by

\[ \vec{k} = k \begin{bmatrix} \cos \theta \\ \sin \theta \end{bmatrix} = k \vec{1}_k. \]  

(A.3)

Equation (A.1) is discretized in space by introducing the spatial derivatives corresponding to a spatial discretization method. A uniform grid with periodic boundary conditions is considered. The grid is defined by a generating pattern, which is the smallest part from which the full grid can be reconstructed by periodically repeating the pattern in all directions. For the 2D case, the generating patterns for uniform triangular and quadrilateral meshes are shown in Figures A.1(a) and A.1(b). The generating pattern is completely defined by the vectors \(\vec{B}_1\) and \(\vec{B}_2\) and their non-dimensional form is obtained by scaling them with the length of \(\vec{B}_1\), denoted by \(\Delta B\): \(\vec{B}_1 \equiv \Delta B \vec{B}_1'\) and \(\vec{B}_2 \equiv \Delta B \vec{B}_2'\). If the dimensionless vector \(\vec{B}_1'\) is chosen as \([1 \ 0]^T\), then the dimensionless mesh is completely defined by the two components of \(\vec{B}_2'\).

In 2D, the advection speed \(\vec{a}\) in Equation (A.1) is defined by its amplitude \(a\) and orientation angle \(\psi\):

\[ \vec{a} = a \begin{bmatrix} \cos \psi \\ \sin \psi \end{bmatrix}. \]  

(A.4)
A.1. STABILITY OF SPATIAL DISCRETIZATIONS

At the cell faces, which lie on a grid face between two cells, two values for the convected variables are available, from the solution corresponding to the two neighboring cells. In order to ensure numerical conservation, the contributions of a face to its two neighboring cells should be equal in magnitude and opposite in sign. Thus, a unique flux should be computed from the two available solutions. In the present analysis, the following approximate Riemann flux is used:

\[
\vec{F}^{AR}(W_L, W_R) \cdot \vec{1}_n = \vec{a} \cdot \vec{1}_n \frac{W_L + W_R}{2} - \varphi \left| \vec{a} \cdot \vec{1}_n \right| \frac{W_R - W_L}{2},
\]

where \(\vec{1}_n\) is the unit normal oriented from the left to the right side and indices \(L\) and \(R\) indicate respectively the left and right neighboring cell to a face. \(\varphi\) is an upwinding parameter, with \(\varphi = 1\) resulting in an upwind flux and \(\varphi = 0\) in a central flux.

After application of the space method to (A.1) on a uniform quadrilateral or triangular mesh, the following system of ODEs is obtained:

\[
\frac{dW_{i,j}}{dt} + \frac{a}{\Delta B} \left( T^{0,0} W_{i,j} + T^{-1,0} W_{i-1,j} + T^{0,-1} W_{i,j-1} + T^{1,0} W_{i+1,j} + T^{0,1} W_{i,j+1} \right) = 0,
\]

where the five dimensionless matrices \(T\) are defined by the spatial discretization method. They depend on the generating pattern, the advection speed orientation angle \(\psi\) and the upwinding parameter \(\varphi\). The column vector \(W_{i,j}\) contains the \(N^{s,GP}\) solution variables in the generating pattern with indices \(i\) and \(j\) (from within one quadrilateral cell or two triangular cells).
We seek for a spatial Fourier wave solution of the form

\[ W_{i,j}(t) = \tilde{W}(t) e^{i K \cdot (i \vec{B}_1' + j \vec{B}_2')} \Delta B, \quad (A.7) \]

where \( K \) is the dimensionless wave number, defined as \( k \Delta B \). Inserting expression (A.7) in Equation (A.6) results in

\[ \frac{d\tilde{W}}{dt} = \frac{a}{\Delta B} \mathbf{T} \tilde{W}, \quad \mathbf{T}' = \frac{a}{\Delta B} \mathbf{T} \quad (A.8) \]

with the matrix \( \mathbf{T} \) defined by

\[ \mathbf{T} = -\left( \mathbf{T}^{0,0} + \mathbf{T}^{-1,0} e^{-i K \cdot \vec{B}_1'} + \mathbf{T}^{0,-1} e^{-i K \cdot \vec{B}_2'} + \mathbf{T}^{+1,0} e^{i K \cdot \vec{B}_1'} + \mathbf{T}^{0,+1} e^{i K \cdot \vec{B}_2'} \right). \quad (A.9) \]

The stability analysis of the space discretization is based on the eigenvalues structure of the matrix \( \mathbf{T}' \), since the exact solution of the system (A.8) is directly determined by the eigenvalues and eigenvectors of \( \mathbf{T}' \). Let \( \Theta'_m = \Theta'_{m,R} + i \Theta'_{m,I}, m = 1, \ldots, N_{s,GP} \) be the complex eigenvalues of the matrix \( \mathbf{T}' \) solution of the eigenvalue equation

\[ \det [\mathbf{T}' - \Theta' \mathbf{I}] = 0 \quad (A.10) \]

and \( \mathbf{V}'_m \), the associated eigenvectors. Expression (A.10) defines the numerical dispersion relation of the spatial scheme, i.e. the relation between eigenvalues \( \Theta' \), the dimensionless wave number \( K \), the wave orientation angle \( \theta \) and the the convection speed orientation angle \( \psi \). This relation should be compared with the exact dispersion relation \( \Theta_{ex,R} = 0, \Theta_{ex,I} = K \cos (\psi - \theta) \) to assess the capability of the spatial scheme to model wave propagation [75, 76, 176].

Since, the eigenvectors \( \mathbf{V}'_m \) form a complete set of basis vectors in the considered space of solution variables, the exact solution of (A.8) can always be written as a linear combination of these eigenvectors:

\[ \tilde{W}(t) = \sum_{m=1}^{N_{s,GP}} \tilde{W}'_m(t) \mathbf{V}'_m, \quad (A.11) \]
where the $\dot{W}'_m$ coefficients depend only on time. These coefficients are obtained from the differential system (A.8), by inserting (A.11), leading to $N_{s,GP}$ homogeneous modal equations:

$$
\frac{d\dot{W}'_m}{dt} = \Theta'_m \dot{W}'_m = \frac{a}{\Delta B} \Theta_m \dot{W}'_m, \quad m = 1, \ldots, N_{s,GP},
$$

(A.12)

where $\Theta_m = \Theta'_m \frac{\Delta B}{a}$, $m = 1, \ldots N_{s,GP}$ are the eigenvalues of the matrix $T$ defined by Equation (A.9). The solution of these ODEs is of the form

$$
\dot{W}'_m(t) = e^{\Theta'_m t} = e^{(a/\Delta B)\Theta_m t}, \quad m = 1, \ldots, N_{s,GP}.
$$

(A.13)

Therefore, expression (A.11) can be written as linear combination of the $N_{s,GP}$ eigenmode solutions $e^{\Theta'_m t} V'_m$:

$$
\dot{W}(t) = \sum_{m=1}^{N_{s,GP}} \dot{W}^{0}_m e^{\Theta'_m t} V'_m,
$$

(A.14)

where the coefficients $\dot{W}^{0}_m$ ensure that the initial condition (A.2) is satisfied:

$$
W^0(\vec{\chi}) = \sum_{m=1}^{N_{s,GP}} \dot{W}^{0}_m V'_m.
$$

(A.15)

It can be easily seen from expression (A.14) that, the ODE system is well-posed, or stable, if the exact solution remains bounded. This implies that all the modal components are also bounded, since if any of them would grow exponentially with time, the full solution would also have this unwanted behavior. Hence, for stability, the real part of the eigenvalues $\Theta'_m$ must be negative or zero. In addition, if an eigenvalue is zero, it has to be a simple eigenvalue. Therefore, the left-half of the complex plane, including the imaginary axis, is the region of stability for the exact solution of the semi-discretized system of equations.

The collection of all eigenvalues $\Theta_m$ for all wave numbers $K$ is often called ‘Fourier footprint’. The Fourier footprint of a spatial scheme gives an indication of the stiffness of the problem and grows very fast with increasing the order of the method. For instance, for the linear advection equation the growing is faster than proportionally with the order of accuracy of the spatial method, as illustrated in Figures A.2 and A.3 for second-, third- and

\(^1\) An ordinary differential equation problem is stiff if the solution being sought is varying slowly, but there are nearby solutions that vary rapidly, so the numerical method must take small steps to obtain satisfactory results. Stiffness is an efficiency issue. If we weren’t concerned with how much time a computation takes, we wouldn’t be concerned about stiffness.

181
fourth-order accurate spectral volume and spectral difference schemes for triangular and quadrilateral cells respectively.

Figure A.2: Fourier footprints of second- to fourth-order SV schemes for triangular cells for the linear advection equation.

Figure A.3: Fourier footprints of second- to fourth-order SD schemes for quadrilateral cells for the linear advection equation.

**A.2 Stability of time discretizations**

In the previous section it has been shown that, the real part of the eigenvalues $\Theta'_m$ of the Fourier footprint must be negative or zero, for stability. Only if this condition is satisfied the exact solution of the semi-discretized system of equations remains bounded. However, it is quite possible for the numerical solution of an ODE to grow unbounded even though its exact solution is well behaved. Of course, there are cases for which the exact solution is well grows unbounded, but for the discussion of the stability, only cases in which the exact solution is bounded are considered. Consider a differential equation of the form

$$\begin{cases} \frac{dy}{dt} = f(y, t), & t > 0 \\ y(0) = y_0. \end{cases}$$
The aim of all numerical methods for solution of this differential equation is to obtain the solution at time $t^{n+1} = t^n + \Delta t$, given the solution for $0 \leq t \leq t^n$. This process continues; i.e., once $Y^{n+1} = Y(t^{n+1})$ is obtained, then $Y^{n+2}$ is calculated and so on until the final time, $t^{end}$. The stability analysis of the numerical method, is performed in terms of the parameters of the numerical method (mainly the step size $\Delta t$) for which the numerical solution remains bounded. In this context we have three classes of numerical methods:

- **Unconditionally stable**: numerical solution does not grow unbounded (blow up) with any choice of parameters such as the time step. The cost of such robustness must be investigated.

- **Unstable scheme**: numerical solution grows unbounded with any choice of parameters. Such numerical schemes are not useful.

- **Conditionally stable scheme**: with certain choices of the parameters the numerical solution remains bounded. Hopefully, the cost of the calculation does not become prohibitively large.

For convenience and feasibility of analytical treatment, stability analysis of time discretization schemes is usually performed on the model problem, consisting of

$$\begin{cases}
\frac{dy}{dt} = \Theta y(t), & t > 0 \\
y(0) = y_0.
\end{cases}$$  \hspace{1cm} (A.16)

The model equation has an analytical solution $y(t) = e^{\Theta t}$. Here, $\Theta$ is a complex constant, i.e $\Theta = \Theta_R + I\Theta_I$, with the real part $\Theta_R < 0$ to ensure that the solution does not grow with $t$. This generalization allows to readily apply the results of the next analysis to system of ODEs. In fact, in Section A.1, it has been shown that systems of ODEs, obtained for instance from spatial discretizations of PDEs, can reduce to uncoupled ODEs of the form (A.16) with complex coefficients. These coefficients correspond to the eigenvalues $\Theta_m$ of the matrix $T$, which is defined by Equation (A.9).

### A.2.1 Forward Euler scheme

The **forward Euler scheme** (FE) uses the first-order forward difference formula to approximate the time derivatives:

$$Y^{n+1} = Y^n + \Delta t f(Y^n, t^n).$$  \hspace{1cm} (A.17)

Applying this scheme to the model problem (A.16) leads to

$$Y^{n+1} = Y^n + \Theta \Delta t Y^n = (1 + \Theta \Delta t) Y^n.$$  \hspace{1cm} (A.18)
For complex $\Theta$, we have
\[
Y^{n+1} = (1 + \Theta_R \Delta t + I \Theta_I \Delta t) Y^n = G Y^n,
\]
where $G = (1 + \Theta_R \Delta t + I \Theta_I \Delta t)$ is called amplification factor. The numerical solution is stable (i.e., remains bounded as time becomes large) if
\[
|G| \leq 1.
\]
For the system (A.8), the stability condition is then given by
\[
|G| = \max_m |1 + \sigma \Theta_m| \leq 1,
\]
where $\Theta_m$ are the complex eigenvalues of the matrix $T$ and $\sigma = \frac{a \Delta t}{\Delta B}$ is Courant-Friedrichs-Lewy (CFL) number.

The general form of the amplification factor of the FE scheme can be written as
\[
G(z) = 1 + z.
\]

The region in the complex plane defined by $|G(z)| \leq 1$ is called stability region. For the FE scheme, it is a circle with a radius of one in the complex plane.
A.2. STABILITY OF TIME DISCRETIZATIONS

plane, centered around $-1$, as shown in Figure A.4. The FE scheme is then a conditionally stable scheme. The stability condition for the complete discretization is that the Fourier footprint of the spatial scheme scaled with the CFL number, i.e. $\sigma \Theta_m$ for $m = 1, \ldots, N_{s, GP}$, lies entirely inside the stability region of the FE scheme. Comparing the Fourier footprints shown in Figures A.2 and A.3 with the stability region of the FE scheme, one can conclude that, combinations of high-order accurate spatial discretizations with FE method lead to stable complete discretizations only when a very small CFL number is used.

A.2.2 Backward Euler scheme

The backward Euler (BE) scheme is given by the following formula:

$$Y^{n+1} = Y^n + \Delta t f (Y^{n+1}, t^{n+1}). \quad (A.23)$$

Note that in contrast with the FE scheme, the BE scheme does not allow to easily obtain the solution at the next time step. If $f$ is non-linear, then a non-linear algebraic equation must be solved at each time step to obtain $Y^{n+1}$. Therefore, the computational cost per time step for this scheme is, apparently, much higher than that for the FE scheme. An efficient algebraic solver is then needed to fully fulfill the potential of the Euler method, which, as it will be shown further on, has better stability properties.

Applying the BE scheme to the model problem (A.16), one obtains

$$Y^{n+1} = Y^n + \Theta \Delta t Y^{n+1}. \quad (A.24)$$

Solving for $Y^{n+1}$ produces

$$Y^{n+1} = \frac{1}{1 - \Theta \Delta t} Y^n \quad (A.25)$$

or

$$Y^{n+1} = G Y^n, \quad (A.26)$$

where

$$G = \frac{1}{1 - \Theta \Delta t}. \quad (A.27)$$

Considering complex $\Theta$, the following expression for the amplification factor $G$ is obtained

$$G = \frac{1}{(1 - \Theta R \Delta t) - I \Theta I \Delta t}. \quad (A.28)$$
The denominator is a complex number and can be written as

\[ G = \frac{1}{Ae^{i\theta}}, \]  
(A.29)

where

\[ A = \sqrt{(1 - \Theta_R)^2 + \Theta_I^2 \Delta t^2}, \quad \theta = -\tan^{-1} \frac{\Theta_I \Delta t}{1 - \Theta_R \Delta t}. \]  
(A.30)

For stability, the modulus of \( G \) must be less than or equal to 1; i.e.,

\[ |G| = \frac{e^{-i\theta}}{A} = \frac{1}{A} \leq 1 \quad \forall \Theta_R \leq 0, \]  
(A.31)

where

\[ \theta = \tan^{-1} \frac{G_I}{G_R}. \]  
(A.32)

The amplification factor of the BE scheme can be written in the following general form:

\[ G(z) = \frac{1}{1 - z}. \]  
(A.33)

The stability region of the BE scheme is then the entire complex plane minus the region enclosed by a circle with a radius of one and centered around +1, as shown in Figure A.5.

Combining the spatially stability condition derived in Section A.1 (i.e. \( \Theta_{m,R} \leq 0 \) for \( m = 1, ..., N_{s,G_P} \)) for the 2D linear advection equation with the stability condition of the BE scheme, one can conclude that there is no restriction on the CFL number for the stability of the complete space and time discretization. In addition, the numerical solution \( Y^n \) obtained with the BE scheme satisfies the condition \( |Y^n| \rightarrow 0 \), for \( t^n \rightarrow \infty \), i.e. the BE scheme is \( A \)-stable [135]. The \( A \)-stability property is important to solve the systems of stiff ODEs arising from the discretization of the fluid dynamic equations with a spatially high-order numerical scheme. In fact, \( A \)-stable methods do not exhibit instability problems. Notice that \( |G(z)| \) tends also to zero for large values of \( z \). Thus the BE scheme is also \( L \)-stable [135].

### A.2.3 Second-order backward difference formula

The backward difference formulae (BDF) represent a class of linear multi-step methods which was discovered in 1952, together with the phenomenon
of stiffness, by Curtiss & Hirschfelder [45] in calculations for chemistry, and their extreme importance for stiff problems has been recognized since the work of Gear in 1971 [53]. With the BDF, the time derivative $\frac{dy}{dt}$ at time $t^{n+1}$ is approximated with the derivative of the $k + 1$ degree polynomial, which is constructed by interpolating the solution at $k + 2$ time points $t^n, t^{n-1}, \ldots, t^{n-k}$. This procedure leads to schemes of the following form [135]:

$$Y^{n+1} = \sum_{j=0}^{k} a_j Y^{n-j} + \Delta t \ b_{-1} f (Y^{n+1}, t^{n+1}), \quad (A.34)$$

with $b_{-1} \neq 0$. For $k = 0$ (first-order interpolation polynomial), assuming $a_0 = 1$ and $b_{-1} = 1$, one obtains the BE scheme (A.23). Thus, the BE scheme can be considered as a first-order backward difference formula.

The second-order ($k = 1$) backward difference scheme (BDF2) with constant time step is defined by $a_0 = \frac{4}{3}, a_1 = -\frac{1}{3}$ and $b_{-1} = \frac{2}{3}$, i.e.

$$Y^{n+1} = \frac{4}{3} Y^n - \frac{1}{3} Y^{n-1} + \frac{2}{3} \Delta t f (Y^{n+1}, t^{n+1}). \quad (A.35)$$
Applying this scheme to the model problem (A.16), one obtains

\[ Y^{n+1} = \frac{4}{3} Y^n - \frac{1}{3} Y^{n-1} + \frac{2}{3} \Theta \Delta t Y^{n+1}. \]  

(A.36)

The amplification factor of the BDF2 is then the solution of the following equation

\[ (3 - 2z) G^2 - 4G + 1 = 0, \]  

(A.37)

which has been obtained by substituting the general expression

\[ G = \frac{Y^{n+1}}{Y^n} \]

in (A.36). Equation (A.37) has two roots:

\[ G_{\pm} = \frac{2 \pm \sqrt{1 + 2z}}{3 - 2z}. \]  

(A.38)

The amplitude of \( G_- \) is always smaller than one, while |\( G_+ (z) \)| is smaller than one if \( z_R \leq 0 \). The module of \( G_+ (z) \) is shown in Figure A.6. The BDF2 is both \( A \)-stable and \( L \)-stable.

![Figure A.6: Amplification factor module for the second-order backward difference formula.](image)

A.2.4 Higher-order backward difference formulae

Following the procedure described at the beginning of Section A.2.3, higher-order BDF (order of accuracy higher than two) can be constructed. However, implicit multi-step methods can only be \( A \)-stable if their order is at
A.2. STABILITY OF TIME DISCRETIZATIONS

most two. This result is known as the second Dahlquist barrier [135]. In Table A.1 the coefficients of all high-order zero-stable BDF are listed. A scheme for which perturbations remain bounded in the limit $\Delta t \to 0$ is said to be zero-stable [135]. The high-order BDF considered here are also $L$-stable. The limit of the stability zone ($|G(z)| = 1$) of the BDF for

$k = 2, 3, 4, 5$ are shown in Figure A.7. It can be easily seen that these schemes are not $A$-stable (second Dahlquist barrier). In fact, the curves for which $|G(z)| = 1$ cross the imaginary axis. In addition, the region of instability in the left-half complex plane increases by increasing the order of accuracy of the BDF, as illustrated in Figure A.7(b).

As shown in Figures A.2 and A.3, both high-order SV and SD schemes have a wide range of eigenvalues in the vertical direction and very close to
the imaginary axis. This might lead to an unwanted reduction of the maximum CFL value when associated with high-order accurate BDF\textsuperscript{2}. In fact, for the complete discretization with a general spatial scheme and a general time marching scheme, the stability condition is then that the Fourier footprint of the spatial scheme, scaled with the CFL-number, lies entirely inside the stability zone of the time marching scheme.

Notice that the amplification factor $G(z)$ can be defined for a general time marching scheme and the stability zone, defined by $|G(z)| \leq 1$, can always be determined. Therefore, it is always possible to find a value of the time step $\Delta t$ and/or the CFL number which leads to a stable complete discretization. However, for spatially high-order accurate schemes, this value might be very small because such spatial discretizations generate very often systems of stiff ODEs. This is very common in computational fluid dynamics where the time integration scheme must also be able to deal with the geometrical stiffness imposed by the Navier-Stokes grids. In the case of compressible solvers there is an additional stiffness when solving for low speed flows caused by the disparate eigenvalues of the system. The A-stability property is then very important to solve the systems of stiff ODEs.

### A.3 Method for the analysis of the non-linear LU-SGS algorithm

The aim of this section is to present a methodology to analyze the smoothing properties of the non-linear LU-SGS algorithm when combined with the backward Euler scheme. This scheme, denoted as ‘LU-SGS + BE’, was presented in Section 5.1 and is used in Chapters 7 and 8 as a smoother to solve steady flow problems. Its analysis is then fundamental because essential for efficiency is that the solver is a good smoother of high-frequency error components.

The damping properties of the LU-SGS + BE are evaluated with a Von Neumann stability analysis for a model 2D linear advection. This model problem was already discussed in Section 3.5. The starting point of the analysis is the 2D semi-discretized linear advection equation (A.8). Since the direct inversion method performs better than any approximate method, the procedure to derive its amplification matrix is also illustrated.

\textsuperscript{2}For accuracy, a smaller value for $\Delta t$ may be required.
A.3. ANALYSIS OF THE NON-LINEAR LU-SGS ALGORITHM

A.3.1 Direct inversion method

Consider the 2D semi-discretized linear advection equation (A.8) obtained by substituting the spatial Fourier wave (A.7) in Equation (A.6). For convenience, this equation is repeated here:

\[
\frac{d\tilde{W}}{dt} + \frac{a}{\Delta B} \left( T_{0,0}^0 + T_{-1,0}^{-1} e^{-IK\hat{r}_k \cdot \vec{B}'_1} + T_{0,-1}^0 e^{-IK\hat{r}_k \cdot \vec{B}'_2} + T_{+1,0}^0 e^{IK\hat{r}_k \cdot \vec{B}'_1} + T_{0,+1}^0 e^{IK\hat{r}_k \cdot \vec{B}'_2} \right) \tilde{W} = 0, \tag{A.39}
\]

where \(a\) and \(\Delta B\) are the module of the advection speed vector and the reference length scale respectively. In (A.39), the five dimensionless matrices are defined by the spatial discretization method and correspond to the discretized spatial derivatives. They depend on the generating pattern, the advection speed orientation angle \(\psi\) and the upwinding parameter \(\phi\) of the approximate Riemann flux (A.5).

Approximating the time derivative with the BE scheme, discussed in Section A.2.2 and defined by (A.23), Equation (A.40) is obtained,

\[
\tilde{W}^{n+1} - \tilde{W}^n + \sigma \left( T_{0,0}^0 + T_{-1,0}^{-1} e^{-IK\hat{r}_k \cdot \vec{B}'_1} + T_{0,-1}^0 e^{-IK\hat{r}_k \cdot \vec{B}'_2} + T_{+1,0}^0 e^{IK\hat{r}_k \cdot \vec{B}'_1} + T_{0,+1}^0 e^{IK\hat{r}_k \cdot \vec{B}'_2} \right) \tilde{W}^{n+1} = 0, \tag{A.40}
\]

where \(\sigma = \frac{a \Delta t}{\Delta B}\) is the CFL number. The linear system (A.40) may also be written in a more compact form, i.e.

\[
E \tilde{W}^{n+1} = \tilde{W}^n, \tag{A.41}
\]

where the matrix \(E\) is obtained by assembling the matrices \(T_{0,0}^0, T_{-1,0}^{-1}, T_{0,-1}^0, T_{+1,0}^0\) and \(T_{0,+1}^0\). An expression for the amplification matrix \(G_d\), defined by \(\tilde{W}^{n+1} = G_d \tilde{W}^n\), can be obtained from system (A.40):

\[
G_d = \left[ I + \sigma \left( T_{0,0}^0 + T_{-1,0}^{-1} e^{-IK\hat{r}_k \cdot \vec{B}'_1} + T_{0,-1}^0 e^{-IK\hat{r}_k \cdot \vec{B}'_2} + T_{+1,0}^0 e^{IK\hat{r}_k \cdot \vec{B}'_1} + T_{0,+1}^0 e^{IK\hat{r}_k \cdot \vec{B}'_2} \right) \right]^{-1}. \tag{A.42}
\]

The matrix \(G_d\) represents the amplification matrix of the direct inversion method which is marked as ‘direct + BE’ in this work.

A.3.2 Non-linear LU-SGS algorithm

The analysis described for the direct inversion method may be applied to the LU-SGS + BE scheme to get its amplification matrix. The LU-SGS +
APPENDIX A. TIME INTEGRATION METHODS FOR ODEs

BE method is used to solve a non-linear system of equations. However, when it is employed to solve a linear problem, it is identical to the classical LU-SGS algorithm proposed by Jameson and Turkel [88]. The main idea of the linear LU-SGS method is to split the matrix $E$ in Equation (A.41) into a diagonal, a strictly lower and a strictly upper matrix

$$E = D + L + U$$  \hspace{1cm} (A.43)

and solve the linear system with forward and backward Gauss-Seidel sweeps:

$$\begin{align*}
(D + L) \tilde{W}_{m+1/2}^n + U \tilde{W}_m^n &= \tilde{W}_n^n, \\
(D + U) \tilde{W}_{m+1}^n + L \tilde{W}_{m+1/2}^n &= \tilde{W}_n^n,
\end{align*}$$

where $m = 0, 1, 2, 3, \ldots$ is the actual SGS sweep index. The first equation, with solution $\tilde{W}_{m+1/2}^n$, corresponds to the forward sweep and the second one, with solution $\tilde{W}_{m+1}^n$ represents the backward sweep. Note that $\tilde{W}_{m=0}^n = \tilde{W}_n^n$.

Because of the Gauss-Seidel nature of the non-linear LU-SGS algorithm, where the latest available solution in the neighboring cells is used to update the solution in a cell, the procedure to obtain the amplification matrix for the LU-SGS depends on the generating pattern structure, through the matrix $T_{0,0}$. In fact, the column vector $\tilde{W}_{i,j}$ in Equation (A.6) contains the $N_{s,GP}$ solution variables of the generating pattern which corresponds to one quadrilateral cell or two triangular cells, as illustrated in Figures A.1(b) and A.1(a). In the next sections, the procedure for both grids is discussed.

**Uniform triangular mesh**

In case of uniform triangular grid, the solution variables of the generating pattern are the solution variables of two triangular cells. Consequently, the matrix $T^{0,0}$ has to be split in the following form:

$$T^{0,0} = T^{0,0}_{1,1} + T^{0,0}_{1,2} + T^{0,0}_{2,1} + T^{0,0}_{2,2},$$

where $T^{0,0}_{1,1}$ and $T^{0,0}_{2,2}$ represent respectively the contribution to the residual of the first and second cell of the generating pattern to themselves, while $T^{0,0}_{1,2}$ and $T^{0,0}_{2,1}$ represent the cross contributions of both cells of the generating pattern. After substituting Equation (A.45) into Equation (A.40), the
A.3. ANALYSIS OF THE NON-LINEAR LU-SGS ALGORITHM

equation of the amplification matrix for the first forward sweep becomes

\[
\begin{align*}
\mathbf{W}^{1/2} - \mathbf{W}^n + \sigma \left( T_{1,1}^{0,0} \mathbf{W}^{1/2} + T_{1,2}^{0,0} \mathbf{W}^n + T_{2,1}^{0,0} \mathbf{W}^{1/2} + T_{2,2}^{0,0} \mathbf{W}^n \right) \\
+ T^{-1,0} e^{-IK \mathbf{\tilde{B}}_1} \mathbf{W}^{1/2} + T^{1,0} e^{-IK \mathbf{\tilde{B}}_2} \mathbf{W}^{1/2} \\
+ T^{0,1} e^{IK \mathbf{\tilde{B}}_1} \mathbf{W}^n + T^{0,0} e^{IK \mathbf{\tilde{B}}_2} \mathbf{W}^n 
\end{align*}
\]

(A.46)

from which the amplification matrix of the first forward sweep (A.47) is found.

\[
\begin{align*}
\mathbf{G}^{TR}_{f,1} &= \left[ \mathbf{I} + \sigma \left( T_{1,1}^{0,0} + T_{1,2}^{0,0} + T_{2,1}^{0,0} + T_{2,2}^{0,0} \right) \right]^{-1} \\
&\times \left[ \mathbf{I} - \sigma \left( T_{1,1}^{0,0} + T_{1,2}^{1,0} e^{IK \mathbf{\tilde{B}}_1} + T_{2,1}^{0,0} + T_{2,2}^{0,1} e^{IK \mathbf{\tilde{B}}_2} \right) \right] \quad (A.47)
\end{align*}
\]

For the first backward sweep, in accordance with Equation (A.44b), one obtains

\[
\begin{align*}
\mathbf{W}^1 - \mathbf{W}^n + \sigma \left( T_{1,1}^{0,0} \mathbf{W}^1 + T_{1,2}^{0,0} \mathbf{W}^1 + T_{2,1}^{0,0} \mathbf{W}^1 + T_{2,2}^{0,0} \mathbf{W}^1 \right) \\
+ T^{-1,0} e^{-IK \mathbf{\tilde{B}}_1} \mathbf{W}^{1/2} + T^{0,1} e^{-IK \mathbf{\tilde{B}}_2} \mathbf{W}^{1/2} \\
+ T^{1,0} e^{IK \mathbf{\tilde{B}}_1} \mathbf{W}^1 + T^{0,0} e^{IK \mathbf{\tilde{B}}_2} \mathbf{W}^1 
\end{align*}
\]

(A.48)

Hence, the amplification matrix of the first SGS sweep is given by

\[
\begin{align*}
\mathbf{G}^{TR}_{SGS,1} &= \left[ \mathbf{I} + \sigma \left( T_{1,1}^{0,0} + T_{1,2}^{0,0} + T_{2,1}^{0,0} + T_{2,2}^{0,0} \right) \right]^{-1} \\
&\times \left[ \mathbf{I} - \sigma \left( T_{1,1}^{0,0} + T_{1,2}^{1,0} e^{IK \mathbf{\tilde{B}}_1} + T_{2,1}^{0,0} + T_{2,2}^{0,1} e^{IK \mathbf{\tilde{B}}_2} \right) \right] \quad (A.49)
\end{align*}
\]

From the amplification matrix \( \mathbf{G}^{TR}_{SGS,1} \), the amplification matrix of \( m \)-th SGS sweep may be computed using the following two-step recursive procedure:

1: Compute the amplification matrix of the \( m \)-th forward sweep

\[
\begin{align*}
\mathbf{G}^{TR}_{f,m} &= \left[ \mathbf{I} + \sigma \left( T_{1,1}^{0,0} + T_{1,2}^{0,0} + T_{2,1}^{0,0} + T_{2,2}^{0,0} \right) \right]^{-1} \\
&\times \left[ \mathbf{I} - \sigma \left( T_{1,1}^{0,0} + T_{1,2}^{1,0} e^{IK \mathbf{\tilde{B}}_1} + T_{2,1}^{0,0} + T_{2,2}^{0,1} e^{IK \mathbf{\tilde{B}}_2} \right) \right] \mathbf{G}^{TR}_{SGS,m-1} \quad (A.50)
\end{align*}
\]
APPENDIX A. TIME INTEGRATION METHODS FOR ODEs

2: Compute the amplification matrix of the $m$-th SGS sweep

$$
\mathbf{G}^{TR}_{SGS,m} = \left[ \mathbf{I} + \sigma \left( \mathbf{T}^{0,0}_{1,1} + \mathbf{T}^{0,0}_{1,2} + \mathbf{T}^{0,0}_{2,2} + \mathbf{T}^{+1,0} e^{IK\mathbf{1}_k \cdot \mathbf{B}_i^1} + \mathbf{T}^{-1,0} e^{-IK\mathbf{1}_k \cdot \mathbf{B}_i^1} \right) \right]^{-1}
\times \left[ \mathbf{I} - \sigma \left( \mathbf{T}^{0,0}_{2,1} + \mathbf{T}^{-1,0} e^{-IK\mathbf{1}_k \cdot \mathbf{B}_i^1} + \mathbf{T}^{0,0} e^{IK\mathbf{1}_k \cdot \mathbf{B}_i^1} \right) \right].
$$

(A.51)

**Uniform quadrilateral mesh**

In contrast with triangular meshes, the analysis of the amplification matrix for quadrilateral grids does not require the splitting of matrix $\mathbf{T}^{0,0}$. In fact, the generating pattern corresponds to only one quadrilateral cell. Therefore, the equation for the first forward sweep becomes

$$
\overline{\mathbf{W}}^{1/2} - \overline{\mathbf{W}}^n + \sigma \left( \mathbf{T}^{0,0} \overline{\mathbf{W}}^{1/2} + \mathbf{T}^{-1,0} e^{-IK\mathbf{1}_k \cdot \mathbf{B}_i^1} \overline{\mathbf{W}}^{1/2} + \mathbf{T}^{0,0} e^{IK\mathbf{1}_k \cdot \mathbf{B}_i^1} \overline{\mathbf{W}}^{1/2} \right)
+ \mathbf{T}^{+1,0} e^{IK\mathbf{1}_k \cdot \mathbf{B}_i^1} \overline{\mathbf{W}}^n + \mathbf{T}^{0,0} e^{IK\mathbf{1}_k \cdot \mathbf{B}_i^1} \overline{\mathbf{W}}^n,
$$

(A.52)

from which the amplification matrix for the first forward sweep (A.53) is found.

$$
\mathbf{G}^{QD}_{f,1} = \left[ \mathbf{I} + \sigma \left( \mathbf{T}^{0,0} + \mathbf{T}^{-1,0} e^{-IK\mathbf{1}_k \cdot \mathbf{B}_i^1} + \mathbf{T}^{0,0} e^{IK\mathbf{1}_k \cdot \mathbf{B}_i^1} \right) \right]^{-1}
\times \left[ \mathbf{I} - \sigma \left( \mathbf{T}^{+1,0} e^{IK\mathbf{1}_k \cdot \mathbf{B}_i^1} + \mathbf{T}^{0,0} e^{IK\mathbf{1}_k \cdot \mathbf{B}_i^1} \right) \right].
$$

(A.53)

For the first backward sweep, in accordance with Equation (A.44b), one obtains

$$
\overline{\mathbf{W}} - \overline{\mathbf{W}}^n + \sigma \left( \mathbf{T}^{0,0} \overline{\mathbf{W}} + \mathbf{T}^{-1,0} e^{-IK\mathbf{1}_k \cdot \mathbf{B}_i^1} \overline{\mathbf{W}}^{1/2} + \mathbf{T}^{0,0} e^{IK\mathbf{1}_k \cdot \mathbf{B}_i^1} \overline{\mathbf{W}}^{1/2} \right)
+ \mathbf{T}^{+1,0} e^{IK\mathbf{1}_k \cdot \mathbf{B}_i^1} \overline{\mathbf{W}}^n + \mathbf{T}^{0,0} e^{IK\mathbf{1}_k \cdot \mathbf{B}_i^1} \overline{\mathbf{W}}^n.
$$

(A.54)

Therefore, the amplification matrix of the first SGS sweep is given by

$$
\mathbf{G}^{QD}_{SGS,1} = \left[ \mathbf{I} + \sigma \left( \mathbf{T}^{0,0} + \mathbf{T}^{+1,0} e^{IK\mathbf{1}_k \cdot \mathbf{B}_i^1} + \mathbf{T}^{0,0} e^{IK\mathbf{1}_k \cdot \mathbf{B}_i^1} \right) \right]^{-1}
\times \left[ \mathbf{I} - \sigma \left( \mathbf{T}^{-1,0} e^{-IK\mathbf{1}_k \cdot \mathbf{B}_i^1} + \mathbf{T}^{0,0} e^{-IK\mathbf{1}_k \cdot \mathbf{B}_i^1} \right) \right].
$$

(A.55)

From the amplification matrix $\mathbf{G}^{QD}_{SGS,1}$, the amplification matrix of $m$-th SGS sweep may be computed using again the following two-step recursive procedure:
A.3. ANALYSIS OF THE NON-LINEAR LU-SGS ALGORITHM

1: Compute the amplification matrix of the \(m\)-th forward sweep

\[
G_{f,m}^{QD} = \left[ I + \sigma \left( T_{1,1}^{0,0} + T_{1,1}^{-1,0} e^{-IK\vec{t}_k \cdot \vec{B}_1'} + T_{0,-1}^{0,1} e^{-IK\vec{t}_k \cdot \vec{B}_2'} \right) \right]^{-1} \\
\times \left[ I - \sigma \left( +T_{1,1}^{+1,0} e^{IK\vec{t}_k \cdot \vec{B}_1'} + T_{0,+1}^{0,1} e^{IK\vec{t}_k \cdot \vec{B}_2'} \right) \right] g_{f,m}^{QD}.
\] (A.56)

2: Compute the amplification matrix of the \(m\)-th SGS sweep

\[
G_{SGS,m}^{QD} = \left[ I + \sigma \left( T_{1,1}^{0,0} + T_{1,1}^{+1,0} e^{IK\vec{t}_k \cdot \vec{B}_1'} + T_{0,+1}^{0,1} e^{IK\vec{t}_k \cdot \vec{B}_2'} \right) \right]^{-1} \\
\times \left[ I - \sigma \left( +T_{1,1}^{-1,0} e^{-IK\vec{t}_k \cdot \vec{B}_1'} + T_{0,-1}^{0,1} e^{-IK\vec{t}_k \cdot \vec{B}_2'} \right) \right] G_{f,m}^{QD}.
\] (A.57)

A.3.3 Eigenvalue spectrum of the amplification matrix

It is well known in many matrix theory that the properties of a matrix are fully contained in its eigenvalue and eigenvector spectrum, as shown in Section A.1 for the system of semi-discretized PDEs. The analysis of a matrix, or an operator, through the eigenvalues and eigenvectors, represents therefore a most profound investigation of their properties. Application of this analysis to the amplification matrix can can be used to as sess the stability and smoothing properties of the solver.

Let \(\lambda_m\) \((m = 1, 2, \ldots, N^{s,GP})\) be an eigenvalue of the general amplification matrix \(G = G(K, \theta, \psi)\) and \(\lambda = \lambda(G(K, \theta, \psi))\) represent the eigenvalue spectrum of \(G\). Then \(g = g(K, \theta, \psi) \equiv \max |\lambda(G)|\) is the amplification factor for a given \((K, \theta, \psi)\). In order for the global discretization to be stable, \(g \leq 1\) should be satisfied i.e. \(\lambda(G)\) lies inside the unit circle of the complex plane (stability boundary) for all \(K, \theta\) and \(\psi\). The range of \(K\) is one period of the Equation (A.40) and it is marked \(P_K\) in this work. For a fixed shape of the generating pattern, \(P_K\) is a function of \(\theta\), i.e. \(P_K = P_K(\theta)\). The expression from which \(P_K\) can be computed, is obtained by substituting \(\vec{B}_1', \vec{B}_2'\) and \(\theta\) in the exponential terms of Equation (A.40) and using the Euler’s formula, which gives a relation between trigonometric functions and complex numbers. Following this reasoning, a trigonometric function \(f\), which has exactly the same period as Equation (A.40) is obtained. In general, it is not possible to find a closed formulation for \(P_K = P_K(\theta)\). Therefore, the period \(P_K\) is computed taking the inverse fast Fourier transformation of the function \(f\).

Equation (A.40) is periodic in \(K\) with a period \(P_K\). However, for each \(K\),
there are $N_s,_{GP}/2$ and $N_s,_{GP}$ eigenmodes for triangular and the quadrilateral meshes respectively. Consequently, each eigenmode can be interpreted as corresponding to a certain wave number $K + l P_K$, with $l$ an integer number. The actual wave number $K + l P_K$ to which an eigenvalue $\lambda_m$ corresponds should be determined by examining the accompanying eigenmode shape $\tilde{V}_m e^{iK\tilde{r}_k \cdot (i\tilde{B}_1' + j\tilde{B}_2')}$, where $\tilde{V}_m$ are the eigenvectors of the amplification matrix.

If the scheme is used as a smoother for multigrid, then it must have good damping of high-frequency error components, i.e. it should cluster the amplification matrix eigenvalues corresponding to the high-frequency modes towards the origin. Therefore, for high-frequency modes $g \ll 1$ for all $\theta$ and $\psi$ should be satisfied. In addition, we desire that the CFL number be sufficiently large to produce significant reduction (if not elimination) of the convergence slow-down effects that are associated with high-aspect ratio cells. A large CFL number also facilitates the expulsion of error components. At the same time the capability for large CFL numbers must not compromise the high-frequency damping property of the scheme.
Appendix B

$p$-Multigrid

In this appendix a brief introduction to the $p$-multigrid algorithm is given. The main idea of multigrid is based on the observation that error-smoothing operators are generally efficient in eliminating high-frequency errors, but less adequate for the low-frequency errors. The multigrid strategy is to switch to a coarser representation of the solution, where the low-frequency errors of the fine representation occur as high-frequency modes, which can thus be efficiently damped out. In the traditional $h$-multigrid approach, this is done by switching to a coarser spatial grid. With a $p$-multigrid algorithm, a high-order solution representation is transferred to a lower-order one.

The $p$-multigrid is an iterative algorithm in which systems of equations arising from compact, high-order space discretizations, such as spectral volume and spectral difference formulations, are solved by recursively iterating on solution approximations of different polynomial order. For example, to solve equations derived using a polynomial approximation order of $p = 4$, the solution can be iterated on at an approximation order of $p = 4, 3, 2, 1$. The $p$ component of this algorithm was proposed by Rønquist and Patera [146] and analyzed by Maday and Munoz [107] for 1D, Galerkin spectral element discretization of Laplace equation. Helenbrook [68] combined $p$-multigrid with standard low-order multigrid and applied it to an unstructured stream-wise-upwind-Petrov-Galerkin (SUPG) discretization of the incompressible Navier-Stokes equations. In the context of DG approximations, Hemker et al. [71] analyzed block Jacobi smoothing strategies with $h$-multigrid for 1D diffusion problems. In addition, $p$-multigrid, or multi-order, solution strategies have been studied for high-order DG by...
Helenbrook et al. [70], Bassi and Rebay [18] and Fidkowski et al. [48], showing several advantages such as ease of implementation and order-independent convergence rates. It was also applied to the SV method by Van den Abeele et al. [174], Parsani et al. [128, 130] and Kannan et al. [90], and to the SD method by Premasuthan et al. [133] and May et al. [110].

B.1 Full approximation scheme

A two-level full approximation scheme algorithm (FAS) at $p$ and $p-1$ levels can be summarized in the following way [24]. To solve a fine level problem $R^p(W^p) = 0$ perform the following operations:

- Perform $\nu_1$ smoothing sweeps on the fine level $p$ to improve the solution $W^p$: $W^p \leftarrow G^p W^p$.
- Transfer the state and the residual to the coarse level $p-1$:
  
  \[
  W_0^{p-1} \leftarrow I_p^{p-1} W^p,
  \]
  
  \[
  f^{p-1} \leftarrow R^{p-1} (W_0^{p-1}) - R^{p-1} (W^p) = R^{p-1} (W_0^{p-1}) - I_p^{p-1} R^p (W^p).
  \]
- Solve the coarse level problem: $R^{p-1} (W_0^{p-1}) = f^{p-1}$.
- Prolongate the coarse level error and correct the fine level state:
  
  \[
  W^p \leftarrow W^p + I_{p-1}^{p} (W_0^{p-1} - W_0^{p-1}).
  \]
- Perform $\nu_2$ smoothing sweeps on the fine level to improve the solution $W^p$: $W^p \leftarrow G^p W^p$.

In this algorithm, $G^p$ represents an arbitrary smoothing operator on the fine level and $f^{p-1}$ is the so-called forcing function. When the fine level residual is zero, the coarse level correction is zero since $W_0^{p-1} = W^{p-1}$.

The coarse level problem can again be solved using a FAS algorithm, and so on. In this way, one arrives at a V-cycle. A further increase in efficiency can be achieved by initializing the solution on coarser levels, i.e. using the full multigrid approach (FMG). In this way, a better initial solution is provided for the fine levels, which will also improve the robustness of the method. To determine when to switch approximation orders in the FMG process, a residual-based approach can be used.
B.2 Transfer operators

The prolongation and restriction operators between orders are local to the elements. Their definitions are given for the SV method, with omission of the cell index $i$. The definition for the SD method is analogous.

- **Prolongation state operator** $I_{p-1}^p$. For the SV method, the solution within a cell is represented by a polynomial $\sum_{j=1}^{N^s} \tilde{W}_j \tilde{L}_j$. The coarse level polynomials $\tilde{L}_j^{p-1}$ can be written as a function of the fine level polynomials $\tilde{L}_m^p$:

$$\tilde{L}_j^{p-1} = \sum_{m=1}^{N^s_p} \alpha_{jm}^{p-1} \tilde{L}_m^p, \quad j = 1, \ldots, N^s_{p-1}, \quad (B.1)$$

where $N^s_{p-1}$ and $N^s_p$ are the number of CVs within a cell on the coarse and fine level. By equating the fine level solution to the coarse level solution, the following expression for $I_{p-1}^p$ is found: $(I_{p-1}^p)_{mj} \equiv \alpha_{jm}^{p-1}$.

- **State restriction operator** $\tilde{I}_{p-1}^p$. This operator projects the fine level solution onto the coarse level polynomial basis. It is defined as $(\tilde{I}_{p-1}^p)_{jm} = (P^{-1}Q)_{jm}$, with the matrices $P$ and $Q$ defined by

$$P_{jm} = \int_{\Omega} \tilde{L}_j^{p-1} \tilde{L}_m^{p-1} d\Omega, \quad j, m = 1, \ldots, N^s_{p-1}, \quad (B.2)$$

$$Q_{jm} = \int_{\Omega} \tilde{L}_j^{p-1} \tilde{L}_m^p d\Omega, \quad j = 1, \ldots, N^s_{p-1}, m = 1, \ldots, N^s_p. \quad (B.3)$$

- **Residual restriction operator** $I_{p-1}^p$. The SV residuals are CV-averaged quantities, like the SV solution variables. It is defined as

$$(I_{p-1}^p)_{jm} \equiv \Omega_j^{p-1} (\tilde{I}_{p-1}^p)_{jm} \frac{1}{\Omega_m^p}. \quad (B.4)$$
Appendix C

Newton-Raphson GMRES solver

The need to solve non-linear systems of algebraic equations is ubiquitous throughout computational physics. Such systems typically arise from the discretization of partial differential equations (PDEs), whether scalar (such as heat conduction) or a system of coupled equations (such as the Navier-Stokes equations). One may be interested in the steady-state solution of these equations (a boundary value problem) or in their dynamical evolution (an initial value problem). For boundary value problems, non-linear iterative methods are desirable. The same is true for multiple time-scale initial value problems, when discretized implicitly at each time step.

In this appendix, the Newton-Raphson’s method coupled with a generalized minimum residual (GMRES) method, developed by Saad and Schultz [151], is presented. This solver is available in the COOLFluiD collaborative simulation environment, developed at the von Karman Institute for Fluid Dynamics [95, 136, 137]. In Chapter 8, the Newton-Raphson GMRES algorithm is used as reference efficient algebraic solver to assess the convergence properties of the non-linear LU-SGS algorithm for steady-state flow simulations.
C.1 Newton-Raphson algorithm

The overall goal of a non-linear algebraic solver is obtaining a solution to the following problem:

Given \( \mathbf{F} : \mathbb{R}^N \rightarrow \mathbb{R}^N \),

find \( \mathbf{x}^* \) such that \( \mathbf{F}(\mathbf{x}^*) = 0 \),

where \( N \) is the length of the unknown column vector \( \mathbf{x}^* \). For example, considering the non-linear algebraic system associated to a general spatial discretization and the backward Euler time marching scheme, this translates in: find the solution vector at the new time level \( \mathbf{W}^{n+1} \), starting from the solution vector at the previous time level \( \mathbf{W}^n \), of the following non-linear algebraic system

\[
\frac{\mathbf{W}^{n+1} - \mathbf{W}^n}{\Delta t} = \mathbf{R}(\mathbf{W}^{n+1}) = \mathbf{R}^{n+1}.
\]

(C.1)

In the present work, the column vector \( \mathbf{W} \) contains all the solution variables on the grids. The non-linear system (C.1) can be solved with any method for such a system, e.g. the Newton-Raphson algorithm. Writing the \( m \)-th approximation of \( \mathbf{W}^{n+1} \) as \( \mathbf{W}^{n+1,m} \), and the update to the \( (m+1) \)-th approximation as \( \Delta \mathbf{W}^{n+1,m+1} = \mathbf{W}^{n+1,m+1} - \mathbf{W}^{n+1,m} \), the Newton-Raphson algorithm is given by

\[
\left[ -\frac{\partial \mathbf{R}}{\partial \mathbf{W}} \bigg|_{m} + \frac{\mathbf{I}}{\Delta t} \right] \Delta \mathbf{W}^{m+1} = \mathbf{R}^{m} - \frac{\mathbf{W}^{m} - \mathbf{W}^{n}}{\Delta t},
\]

(C.2)

where \( \mathbf{I} \) is the unity matrix and superscripts \( n \) and \( n+1 \) denote the time levels \( t^n \) and \( t^{n+1} \). Time step is given by \( \Delta t = t^{n+1} - t^n \). This expression is a linear algebraic system, which should be inverted at each inner iteration \( m \) of the Newton-Raphson algorithm. This method is an extremely powerful technique- in general the convergence is quadratic: the error is essentially squared at each step.

Solving system (C.2) may be computationally expensive, on the order of \( \mathcal{O}(N^3) \) arithmetic operations for direct linear algebra methods, where \( N \) is the number of unknown of the system. In large-scale problems, where \( N \) is \( 10^4 \), \( 10^5 \) or greater, direct solutions are often infeasible. An alternative is to use iterative linear algebra methods to form an approximation of the solution. Using an iterative linear solver to obtain approximate Newton steps results in a Newton iterative method, or a truncated Newton method. Candidates for this are for instance the Jacobi method or the (symmetric) Gauss-Seidel method, with or without relaxation [135]. Here, the GMRES algorithm is discussed for the inversion of the linear systems.
C.2 GMRES

GMRES is a particular Krylov subspace method and a key to derive it is the Arnoldi process [7]. This method constitutes a class of algorithms designed to solve a linear algebraic problem:

\[
\text{Given } A \in \mathbb{R}^{N \times N} \text{ and } b \in \mathbb{R}^N, \\
\text{find } x \text{ such that } Ax = b.
\]

A Krylov subspace method begins with an initial \(x^{(0)}\) and at the \(l\)-th step, determines an iterate \(x^{(l)}\) through a correction in the \(l\)-th Krylov subspace, which is defined by the following set of basis vectors

\[
r^{(0)}, Ar^{(0)}, A^2r^{(0)}, \ldots, A^{l-1}r^{(0)},
\]

where \(r^{(0)} = b - Ax^{(0)}\) is the initial residual\(^1\). In GMRES, each iterate \(x^{(l)}\) is chosen to minimize the residual norm

\[
\|Ax^{(l)} - b\|.
\]

This is a linear least squares problem, which should be solved at each iteration \(l\). The convergence of the GMRES algorithm is monotonous, since the Krylov subspace \(m\) contains the entire subspace \(l - 1\). It is guaranteed to converge when \(l\) is equal to the size of the matrix \(A\). However, it often performs much better and a good approximation of the exact solution \(x\) is obtained after a few iterations, for certain classes of matrices [171]. Preconditioning methods, like ILU-preconditioners or the additive Schwarz method, are often used to accelerate the convergence of the GMRES algorithm. Like the full orthogonalization method or Arnoldi’s method for linear system, the GMRES requires a big computational cost. For this reason, two variants of the classical GMRES have been developed, namely the \(l\)-step restart GMRES and the Quasi-GMRES. More details on these two variants can be found in Saad and Schultz [151] and Saad and Wu [152].

With methods that are based on the solution of a sparse linear system like (C.2), the amount of memory required to store the sparse matrix may be a serious problem. For instance, on a grid with \(N\) tetrahedral cells with solution polynomial degree \(p\), and with classical diffusive treatment approach, i.e. the LSV/LSD approach (see for instance Section 4.1.3 and 4.2.3), the residuals in one cell depend on the solution in the current cell,

---

\(^1\)Here, the iteration index is enclosed between brackets to avoid confusion with the exponent of the matrix \(A\) in Equation (C.3).
the neighbor cells, and the neighbors of the neighbor cells. Therefore, the total number of non-zero entries in the sparse matrix is then about

\[
\sim 17N \left[ \frac{(p+1)(p+2)(p+3)}{6} \times \# \text{physical variables} \right]^2. \tag{C.5}
\]

With fully compact diffusive approaches, like the BR2 approach described in Section 4.2.3, only the immediate neighbors are required for the computation of the residuals in a cell. In this case, the number of non-zero entries is smaller in comparison with the LSV/LSD approach and it is about

\[
\sim 5N \left[ \frac{(p+1)(p+2)(p+3)}{6} \times \# \text{physical variables} \right]^2. \tag{C.6}
\]

These numbers rapidly increase with \( p \), to the sixth power. Upon comparison with the number of non-zero elements in the Jacobian matrix of the non-linear LU-SG method, (5.9), it is clear that the GMRES method requires significantly more memory than the non-linear LU-SG algorithm.
Appendix D

ESDIRK schemes

Implicit multi-step backward difference formulae (BDF) compute each solution vector update to design order of accuracy using one non-linear equation solve per time step. Unfortunately, they are not $A$-stable above second-order (BDF2) and self-starting. Practical experience indicates that large scale engineering computations are seldom stable if run with the fourth-order backward difference formula (BDF4) [112]. The third-order backward difference formula (BDF3), with its smaller region of instability, is often stable but diverges for certain problems and some spatial operators.

The focus of the studies of Bijl et al. [20, 21], Carpenter et al. [31] and Isono and Zingg [84] is on high-order implicit Runge-Kutta (I-RK) methods, in particular explicit-first-stage, single-diagonal-coefficient, diagonally-implicit Runge-Kutta (ESDIRK) methods of various orders. Implicit multi-stage schemes do not face the $A$-stability restriction of the BDF, and ESDIRK schemes can be of arbitrarily high-order while retaining $A$-stability [21]. In general, higher-order schemes in both space and time become more attractive as the need for accuracy increases, i.e. error tolerances decrease. Consequently, ESDIRK family of schemes, especially the fourth-order variant, could be an efficient alternative to second-order methods (of which the BDF2 which is $A$-stable is the most popular) for problems requiring high accuracy, such as turbulent numerical simulations and long-range wave propagation in computational aeroacoustics.

A general ESDIRK scheme with $s$ stages is given by the following [20]:

\[ \]
APPENDIX D. ESDIRK SCHEMES

\[
\frac{W^k - W^n}{\Delta t} - \sum_{j=1}^{k} a_{kj} R(W^j) = 0, \quad \text{(D.1a)}
\]
\[
W^{n+1} = W^n + \Delta t \sum_{j=1}^{k} b_j R(W^j), \quad \text{(D.1b)}
\]

where superscripts \(n\) and \(n + 1\) denote the time levels \(t^n\) and \(t^{n+1}\), and \(a_{ij}\) and \(b_j\) are the stage and the main scheme weights. Table D.1 shows a Butcher table of the coefficients for a general six-stage ESDIRK scheme, where \(c_i\) are the abscissae that denote the point in the time, \(t + c_i \Delta t\), where the stage is evaluated. ESDIRK schemes differ from traditional SDIRK (see Hairer and Wanner [64]) methods by the choice \(a_{11} = 0\). This means that the first stage is explicit, i.e \(W^{k=1} = W^n\).

Table D.1: Sample Butcher table for a general six-stage ESDIRK scheme.

<table>
<thead>
<tr>
<th></th>
<th>c_2</th>
<th>a_{21}</th>
<th>a_{22}</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>c_3</td>
<td>a_{31}</td>
<td>a_{32}</td>
<td>a_{33}</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>c_4</td>
<td>a_{41}</td>
<td>a_{42}</td>
<td>a_{43}</td>
<td>a_{44}</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>c_5</td>
<td>a_{51}</td>
<td>a_{52}</td>
<td>a_{53}</td>
<td>a_{54}</td>
<td>a_{55}</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>c_6</td>
<td>a_{61}</td>
<td>a_{62}</td>
<td>a_{63}</td>
<td>a_{64}</td>
<td>a_{65}</td>
<td>a_{66}</td>
<td>0</td>
</tr>
</tbody>
</table>

Expression (D.1a) is a non-linear system of algebraic equations which must be solved at each stage. Applying the non-linear LU-SGS algorithm to (D.1a), results in the following general equation for the stage \(k\):

\[
\left[ -a_{kk} \frac{\partial R_{cc}}{\partial W_{cc}} \right]^n + \frac{\mathbf{I}}{\Delta t} \Delta W_{cc}^{k,m+1} = a_{kk} R_{cc}(W^*) + \sum_{j=1}^{k} a_{kj} R_{cc}(W^j) - \frac{\Delta W_{cc}^{k,m}}{\Delta t}, \quad \text{(D.2)}
\]

where the subscripts \(cc\) and \(*\) denote the current cell\(^1\) and the most recent solution when doing forward and backward sweeps respectively, and \(\Delta W_{cc}^{k,m+1} = W_{cc}^{k,m+1} - W_{cc}^{k,m}, \Delta W_{cc}^{k,m} = W_{cc}^{k,m} - W_{cc}^n\).

\(^1\)The non-linear LU-SGS algorithm solves the non-linear system of equations with multiple cell-wise symmetric forward and backward sweeps.

206
The coupling between ESDIRK schemes and the non-linear LU-SGS algorithm was presented for the first time in 2007 by Parsani et al. [128].
List of publications

Journal articles

Accepted


Submitted


LIST OF PUBLICATIONS

Lecture notes


Conference proceedings


Bibliography


BIBLIOGRAPHY


227
BIBLIOGRAPHY


