

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

Institut National Polytechnique de Toulouse (INP Toulouse)

Discipline ou spécialité :

Dynamique des fluides

Présentée et soutenue par :

M. JULIEN VANHAREN

le mardi 16 mai 2017

Titre :

High-order numerical methods for unsteady flows around complex geometries

---

École doctorale :

Mécanique, Énergétique, Génie civil, Procédés (MEGeP)

Unité de recherche :

Centre Européen de Recherche et Formation Avancées en Calcul Scientifique (CERFACS)

Directeur de Thèse :

M. GUILLAUME PUIGT

Rapporteurs :

Mme PAOLA CINNELLA, ENS ARTS METIERS PARIS

M. RODOLPHE TURPAULT, UNIVERSITE DE BORDEAUX

Membres du jury :

M. BRUNO KOOBUS, UNIVERSITE DE MONTPELLIER, Président

M. GUILLAUME PUIGT, CERFACS, Membre

Mme MARTA DE LA LLAVE PLATA, ONERA, Membre

M. PIERRE-HENRI MAIRE, CEA - Cesta LE BARP, Membre

Remerciements

"Si tu sais conclure, alors tu sauras introduire."

— François Pechereau (1985 -)

A l'heure où j'écris ces lignes, j'ai déjà commencé mon post-doc à l'INRIA. Il est maintenant temps pour moi de remercier toutes les personnes qui ont rendu possible cette thèse. Il me faut ainsi tourner la page d'une période de trois ans de ma vie qui m'a tant apporté aussi bien sur le plan scientifique que personnel.

Tout d'abord, j'aimerais remercier les personnes qui ont rendu possible le déroulement de cette thèse, à savoir, Jean-Baptiste Leterrier, Marcus Barth, Marie-Josèphe Estève et Pascal Larrieu du côté Airbus et Thierry Poinsot et Jean-François Boussuge du côté CERFACS puisque cette thèse est le fruit de la collaboration entre ces deux entités. J'aimerais également remercier les rapporteurs de cette thèse ainsi que les membres du jury pour avoir bien voulu assister à ma soutenance de thèse et pour leurs critiques et remarques constructives.

Avec une énorme mention spéciale, j'aimerais remercier du fond du cœur mon directeur de thèse, Guillaume Puigt. Guillaume, merci pour ta patience, tes qualités de pédagogue ton humanité, ta sincérité, ... Impossible d'énumérer toutes tes qualités tant elles sont nombreuses. Approchant de la fin de thèse et étant légèrement fatigué, je me souviendrais toujours d'un moment où tu essayais de m'expliquer une notion. Une fois terminé, je t'ai répondu que je n'avais rien compris et toi, tu as patiemment repris une énième fois ton explication comme si de rien n'était. Merci pour ces années fructueuses où nous avons travaillé ensemble. J'espère bien que cela continuera ! Toujours avec une énorme mention spéciale, je remercie également Marc Montagnac pour son extrême patience avec moi et pour de nombreuses heures passées à débugger des codes. J'aimerais également remercier Xavier Vasseur, Aloïs Sengissen, Jean-François Boussuge, Pierre Sagaut, Thierry Poinsot, Cyril Gacherieu, Loïc Tourrette et Marcus Barth pour toutes les discussions scientifiques que nous avons pu avoir et qui m'ont permis d'avancer à grand pas dans mon travail de thèse quand je butais sur un problème. Enfin, merci à Sophie Le Bras pour nos discussions sur les fameux raccords non conformes, à Luis Miguel Segui Troth pour nos nombreuses discussions sur les schémas numériques, à Majd Daroukh pour nos discussions sur le pre-, co- et post-processing et à Frédéric Alauzet pour ces précieux conseils sur la rédaction du manuscrit et la gestion de la fin de thèse.

Un grand merci également à toute l'équipe CSG et en particulier à Fabrice Fleury, Gérard Dejean, Isabelle D'Ast et Patrick Laporte pour tout leur support technique informatique. Comme je dis toujours, quand tu as un problème informatique, va voir CSG, le problème sera réglé dans la minute et cela t'évitera de perdre de précieuses heures !

Je remercie Chantal Nasri pour son extrême gentillesse et sa légendaire bonne humeur ainsi que tous les membres de l'administration. Merci à Dominique Roffi pour nos discussions sur le si attachant peuple du Nord et de la Picardie, ma région natale. Merci à Marie Labadens et

Nicole Boutet, nos secrétaires, qui font un travail remarquable pour nous aider dans notre vie de tous les jours, nous scientifiques qui avons parfois un peu la tête dans les nuages.

Merci à Nadège Villedieu, Thomas Guédeney et Bastien Rochette qui ont partagé mon bureau et qui ont surtout su me supporter !

Plus largement, je remercie toute la fine équipe des doctorants du CERFACS qui ont soutenu ou soutiendront cette année, l'ensemble des permanents du CERFACS pour tous les bons moments passés durant ces trois années et l'ensemble de mon équipe à Airbus Toulouse. Grâce à eux, une maquette du 350 trône fièrement dans mon bureau à l'INRIA, prête à décoller !

Je m'approche de la fin de ces remerciements et comment ne pourrais-je pas remercier la dream team composée de Thibault Bridel-Bertomeu, François Pechereau, Christophe Coreixas et Gabriel Staffelbach. Merci pour ces innombrables moments passés ensemble à faire une pause ou à travailler sur des sujets sensibles ou à mettre une super ambiance de travail qui rend le CERFACS un endroit où il fait bon travailler ! J'en veux pour preuve cette légendaire citation énoncée ci-dessus ! Je précise qu'il parlait du manuscrit de cette thèse. En effet, selon lui, si l'on sait écrire la conclusion de son manuscrit de thèse alors on saura quoi qu'il arrive en écrire l'introduction.

Merci à mes amis de Suapéro, Clément Pfiffer, mon cher binôme, Quentin Barbès, Nicolas Taglang, Arthur Gauthier et Jonathan Féral pour tous les bons moments que nous avons passés ensemble.

Enfin, je voudrais remercier mes parents sans qui rien de tout cela n'aurait été possible et ma chère et tendre qui m'a supporté durant les derniers dix-huit mois de ma thèse. Tout le monde sait qu'il s'agit de la période la plus tendue et je te remercie pour ta patience, ton soutien et ta tendresse à mon égard dans les moments les plus difficiles. Je suis sûr que j'ai oublié de nombreuses personnes mais qu'elles soient rassurées, je les remercie également.

Voilà, maintenant, il faut que je m'arrête avant de tomber dans le ridicule des effusions sentimentales. Une dernière fois, merci à toutes et tous ! Enfin, vous remarquerez sans doute le petit clin d'oeil dans la signature ci-dessous.

Dr. Julien Vanharen
A Palaiseau, le 7 juin 2017.

Glossary

AIAA	**A**merican **I**nstitute of **A**eronautics and **A**stronautics
ALESIA	**A**ppropriate **L**arge **E**ddy **S**imulation for **A**eroacoustics
ATAG	**A**ir **T**ransport **A**ction **G**roup
CFD	**C**omputational **F**luid **D**ynamics
CFL	**C**ourant-**F**riedrichs-**L**ewy
CGG	**C**orrected **G**reen-**G**auss
COVO	**CO**nvection of a **VO**rtex
CPU	**C**entral **P**rocessing **U**nit
DG	**D**iscontinuous **G**alerkin
DMP	**D**iscrete **M**aximum **P**rinciple
DNS	**D**irect **N**umerical **S**imulation
DRP	**D**ispersion-**R**elation-**P**reserving
elsA	**e**nsemble **l**ogiciel de **s**imulation en **A**érodynamique
ENO	**E**ssentially **N**on-**O**scillatory
FD	**F**inite **D**ifference
FR	**F**lux **R**econstruction
FV	**F**inite **V**olume
HPC	**H**igh **P**erformance **C**omputing
HTP	**H**orizontal **T**ail **P**lane
GG	**G**reen-**G**auss
IMEX	**I**mplicit-**E**xplicit
JAGUAR	pro**J**ect of an **A**erodynamic solver using **G**eneral **U**nstructured grids **A**nd high orde**R** schemes
LES	**L**arge **E**ddy **S**imulation
LMP	**L**ocal **M**aximum **P**roperty
MOOD	**M**ulti-dimensional **O**ptimal **O**rder **D**ectection
MPM	**M**atrix **P**ower **M**ethod
MUSCL	**M**onotone **U**pwind **S**chemes for **S**calar **C**onservation **L**aws
NGI	**N**onconforming **G**rid **I**nterface
NSCBC	**N**avier-**S**tokes **C**haracteristic **B**oundary **C**onditions
PDE	**P**artial **D**ifferential **E**quation
PPW	**P**oints **P**er **W**avelength
RANS	**R**eynolds **A**veraged **N**avier-**S**tokes
RK	**R**unge-**K**utta
SBP	**S**ummation **B**y **P**arts
SCM	**S**uccessive **C**orrections **M**ethod
SD	**S**pectral **D**ifference
SFC	**S**pecific **F**uel **C**onsumption
SV	**S**pectral **V**olume

TGV	**T**aylor-**G**reen **V**ortex
TILDA	**T**owards **I**ndustrial **L**ES/**D**NS in **A**eronautics
TVD	**T**otal **V**ariation-**D**iminishing
UCL	**U**niversité **C**atholique de **L**ouvain
UIG	**U**nstructured **I**nterface **G**radient
URANS	**U**nsteady **R**eynolds **A**veraged **N**avier-**S**tokes
WENO	**W**eighted **E**ssentially **N**on-**O**scillatory

Contents

Remerciements i

Glossary iii

Part I: Introduction 1

1 Introduction 3
- 1.1 Industrial context . . . 3
- 1.2 Industrial needs . . . 4
  - 1.2.1 Reproducibility . . . 5
  - 1.2.2 CFD challenges . . . 5
- 1.3 Ph.D. objective . . . 6
  - 1.3.1 Hybrid approach for the low-order industrial Finite Volume Method . . . 7
  - 1.3.2 Towards high-order hybrid simulations . . . 7
  - 1.3.3 The Spectral Differences Method on unstructured grids . . . 8
  - 1.3.4 The Lattice-Boltzmann Method . . . 8
- 1.4 Publications . . . 9
  - 1.4.1 Papers . . . 9
  - 1.4.2 Conferences . . . 9

Part II: Nonconforming grid interface analysis 11

2 Nonconforming grid interface 13
- 2.1 Introduction and problem description . . . 13
- 2.2 Theoretical aspects . . . 16
  - 2.2.1 Model problem . . . 16
  - 2.2.2 Spatial discretization . . . 16
  - 2.2.3 Basic concepts of spectral analysis . . . 19
  - 2.2.4 Specific case: uniform grid . . . 19
  - 2.2.5 Extension of spectral analysis in two dimensions . . . 20
- 2.3 On stability and accuracy of nonconforming interface . . . 22
  - 2.3.1 Convection of a vortex . . . 22

2.3.2 Coarsening along the x-axis . . . 24
2.3.3 Coarsening along the z-axis . . . 27
2.3.4 Specific case of uniform grids . . . 32
2.3.5 Reflection of high-frequency waves . . . 34
2.4 High-order metric-dependent interpolation for Riemann solver . . . 35
2.5 Conclusion . . . 39

3 Vichnevetsky's framework **41**
3.1 Introductory numerical experiment . . . 41
3.2 Spectral analysis . . . 45
3.3 Stability analysis . . . 46
3.4 Necessity of Vichnevetsky's framework . . . 47
3.4.1 Inside the domain . . . 47
3.4.2 Reflection coefficient . . . 49
3.4.3 Reflection coefficient measurement . . . 50
3.5 Conclusion . . . 50

Part III: Hybrid approach validation **53**

4 Hybrid convective fluxes **55**
4.1 Preliminary remarks . . . 55
4.2 The convection of a vortex: the COVO test case . . . 60
4.2.1 Objectives . . . 60
4.2.2 Description . . . 60
4.2.3 Meshes . . . 61
4.2.4 Numerics . . . 66
4.2.5 Error computation . . . 66
4.3 Results for the first-order Roe scheme . . . 66
4.4 Results for the first-order Roe scheme on a short convection distance . . . 70
4.5 Results for the first-order Roe scheme with NGI on a short convection distance 76
4.6 Results for the second-order Roe scheme . . . 77
4.7 Results for the second-order Roe scheme with NGI . . . 81

5 Hybrid diffusive fluxes **85**
5.1 The Taylor-Green Vortex: the TGV test case . . . 85
5.1.1 Objectives . . . 85
5.1.2 Description . . . 85
5.1.3 Meshes . . . 86
5.1.4 Numerics . . . 87
5.1.5 Error computation . . . 87
5.2 Results . . . 87
5.3 Conclusion . . . 88

Part IV: Extension to high-order unstructured FV schemes 91

6 High-order reconstruction on unstructured grids 93
6.1 The Roe solver . . . 93
6.2 First-order extrapolation . . . 93
6.3 Second-order extrapolation . . . 94
6.4 Third-order extrapolation . . . 95
6.5 Results for the third-order Roe scheme . . . 95

7 The unstructured interface gradient 99
7.1 Finite Volume gradient review . . . 99
7.1.1 Techniques based on Green-Gauss formulation on mesh elements . . . 100
7.1.2 Least-Squares Method for cell-centered gradient . . . 102
7.1.3 Quasi-Green and Quasi-Least-Squares methods . . . 103
7.1.4 Diamond cell approaches . . . 103
7.1.5 Multi Flux Point Approximation . . . 105
7.1.6 Diffusion on curvilinear meshes . . . 105
7.1.7 Other diffusion schemes . . . 107
7.1.8 Partial conclusion . . . 109
7.2 The new interface gradient scheme . . . 109
7.2.1 Building a new dual cell including the interface . . . 110
7.2.2 Computation of the new gradient for the diffusive flux . . . 112

8 *k*-exact schemes 115
8.1 Bibliography . . . 115
8.2 One-dimensional Finite Difference example on a non-uniform mesh . . . 116
8.3 One-dimensional Finite Volume example on a non-uniform mesh . . . 117
8.3.1 Link between nodal value and averaged value . . . 118
8.3.2 *k*-exact principle . . . 118
8.3.3 First-order operator for the first derivative . . . 119
8.3.4 First-order operator for the second derivative . . . 120
8.3.5 Correction procedure: second-order operator for the first derivative . . . 122
8.3.6 Extrapolation formula . . . 122
8.3.7 Truncation error . . . 123
8.3.8 Specific case: uniform mesh . . . 124
8.4 HPC efficiency . . . 125

Part V: The Spectral Difference Method 127

9 Spectral analysis 129
9.1 Introduction . . . 129
9.2 Spectral analysis for high-order Finite Difference methods . . . 131
9.2.1 Spatial discretization . . . 131
9.2.2 Spectral analysis . . . 133
9.2.3 Points Per Wavelength (PPW) . . . 136
9.3 Effect of time integration on the spectral analysis . . . 136

9.3.1 Time discretization . . . 137
9.3.2 Stability analysis based on CFL criterion . . . 138
9.3.3 Spectral analysis . . . 138
9.3.4 Number of Points Per Wavelength (PPW) . . . 138
9.3.5 Summary . . . 138
9.4 Spectral analysis of the Spectral Difference method . . . 140
9.4.1 Description of the Spectral Difference method . . . 140
9.4.2 Matrix form of the SD procedure . . . 141
9.4.3 Matrix form for the spectral analysis . . . 142
9.4.4 Time discretization . . . 143
9.4.5 Stability analysis based on CFL criterion . . . 143
9.4.6 Application of the Nyquist-Shannon sampling theorem . . . 143
9.4.7 A naive approach for the spectral analysis . . . 144
9.4.8 Matrix Power Method for the spectral analysis . . . 145
9.4.9 Comparison with numerical results . . . 147
9.5 Extension to high wavenumbers for the Spectral Difference method . . . 148
9.5.1 Aliasing and initial solution projection . . . 148
9.5.2 Mathematical consideration . . . 150
9.5.3 Energy loss estimation . . . 151
9.5.4 Phase shift estimation . . . 151
9.5.5 Summary . . . 152
9.6 Comparison with standard high-order schemes . . . 153
9.6.1 Naive comparison . . . 153
9.6.2 Rescaling by the number of degrees of freedom . . . 153
9.6.3 New criteria extension to standard aeroacoustic schemes . . . 154
9.6.4 Points Per Wavelength (PPW) - Dissipation-based criterion . . . 155
9.6.5 Points Per Wavelength (PPW) - Dispersion-based criterion . . . 158
9.6.6 Computational cost . . . 161
9.7 Conclusion . . . 161

Part VI: Conclusion & Perspectives **163**

10 Conclusion **165**

11 Perspectives **167**

Part VII: Appendices **169**

A The *elsA* workflow **171**
A.1 Hybrid mesh generation . . . 171
A.2 Hybrid splitter . . . 171
A.3 Face-based connectivity . . . 171
A.4 Coprocessing . . . 172

B Analytical order of accuracy **173**
B.1 Preliminaries . . . 173
B.1.1 Fourier transform . . . 173
B.1.2 The Gaussian integral . . . 173
B.2 Heat equation . . . 174
B.3 Advection-diffusion equation . . . 174
B.4 Error computation . . . 175

C The COVO analytic solution **177**

D Order analysis for unsteady equations **183**
D.1 Truncation error . . . 183
D.2 Discretization error . . . 184

E About the work unit evaluation **187**

Part VIII: Bibliography **189**

I Introduction

1 Introduction **3**
1.1 Industrial context
1.2 Industrial needs
1.3 Ph.D. objective
1.4 Publications

1 Introduction

"Sooner or later, we will have to recognize that the Earth has rights, too, to live without pollution. What mankind must know is that human beings cannot live without Mother Earth, but the planet can live without humans."

— Juan Evo Morales Ayma (1959 -)

1.1 Industrial context

In 2011, the High Level Group on Aviation and Aeronautics Research published the Flightpath 2050 Europe's Vision for Aviation [1]. The main identified objectives are to meet societal and market needs, to maintain and extend industrial leadership, to ensure safety and security, to prioritize research and testing capabilities and to protect the environment and the energy supply. To achieve their last objective, five goals have been defined:

1. In 2050, technologies and procedures available allow a 75% reduction in CO_2 emissions per passenger kilometer to support the ATAG target (Carbon-neutral growth starting 2020 and a 50% overall CO_2 emission reduction by 2050) and a 90% reduction in NO_x emissions. The perceived noise emission of flying aircraft is reduced by 65%. These are relative to the capabilities of typical new aircraft in 2000.
2. Aircraft movements are emission-free when taxiing.
3. Air vehicles are designed and manufactured to be recyclable.
4. Europe is established as a center of excellence on sustainable alternative fuels, including those for aviation, based on a strong European energy policy.
5. Europe is at the forefront of atmospheric research and takes the lead in the formulation of a prioritized environmental action plan and establishment of global environmental standards.

To reduce their environmental footprint, the manufacturers have to improve the efficiency of their aircraft. Basically, there are three ways to improve the efficiency of an aircraft following the famous Breguet range equation:

$$R = V \times \underbrace{\frac{L}{D}}_{\substack{\text{Aerodynamic} \\ \text{efficiency}}} \times \underbrace{\frac{1}{g \cdot SFC}}_{\substack{\text{Propulsion} \\ \text{efficiency}}} \times \underbrace{\ln\left[\frac{W_i}{W_f}\right]}_{\substack{\text{Weight} \\ \text{efficiency}}}, \tag{1.1}$$

where R is the aircraft range, V is the flight speed, L/D is the lift-to-drag ratio, g is the gravity, SFC is the Specific Fuel Consumption, W_i is the initial weight and W_f is the final weight. The first and most obvious way is to improve engines by decreasing the specific fuel consumption. The second technique consists in increasing the lift-to-drag ratio C_L/C_D, playing on aerodynamic shape: one can try to increase the lift, to decrease the drag or to do both. The

last but the not the least way is based on structural analysis: it consists in reducing the aircraft mass. The latter two ways of dealing with the Breguet range equation are analyzed in the context of the European project called AFLoNext ("2nd Generation Active Wing" - Active Flow Loads & Noise control on next generation wing). Even if manufacturers still work on fuel consumption and lift-to-drag ratio, one easy way to get a better aircraft consists in reducing its weight. The A350 is a perfect example of a mass reduction since most of the aircraft is made of composite materials. However, even if they enable mass saving, composite materials have structural characteristics that strongly differ from aluminum. This is important since this structural properties modification can lead to vibrations due to an aeroelastic coupling: a frequency of the unsteady flow could excite an eigenfrequency of the structure. Indeed, generally speaking, dealing with the vibrations tends to be a crucial challenge for aircraft manufacturers.



Fig. 1.1. *Picture of A350-900 during landing.The nose landing creates wake vortices. They are convected under the fuselage and interact with the main landing gear doors. This induces vibrations leading to structure fatigue or even to rupture.*

1.2 Industrial needs

This work focuses on the case where the source of the flow unsteadiness is far from the region where the structure vibrates for which several examples can be highlighted.

- **Emergency descent.** In case of cabin altitude warning or rapid depressurization, the crew should initiate an emergency descent. The airbrakes are then deployed and huge wake vortices are created. They are convected behind the wings and they can interact with the Horizontal Tail Plane (HTP). This interaction can lead to a lot of vibrations which are very uncomfortable for the crew and passengers.
- **Jet-flap interaction.** During low-speed phases, the high-lift devices are deployed to increase lift which is necessary for take-off and landing. The engines also operate at full power. In certain cases, the exhaust gas jet can interact with the deployed flaps leading to structural vibrations.
- **Aeroacoustics.** Sound emitted by aircraft tends to be reduced following the certification

and environmental recommendations.

– **Landing gears.** During landing or take-off, the nose landing gear creates wake vortices which are convected under the fuselage. They could impact and interact with the main landing gear doors thus inducing vibrations leading to structural fatigue and even to rupture. In this context, aircraft manufacturers pay special attention to the interaction between the nose landing gear wake and the main landing gear doors. It is therefore of great importance to be able to compute, capture, transport and analyze the interaction of the nose landing gear with the main landing system, including doors.

Focusing specifically on Computational Fluid Dynamics, three zones can always be identified to analyze the flow physics. The first zone is the zone around the source (airbrakes, jets or nose landing gear) in which the unsteady flow and the wake vortices have to be accurately computed. The second zone is the convection zone (between the wing and the HTP or between the landing gears under the fuselage) in which the created wake vortices need to be convected without being dissipated by numerics. Wake vortices have to be tracked in this propagation zone. Finally, the third zone is the interaction zone (around the HTP, the flaps or the main landing gear). In the specific case of the main landing gear doors, the effects of the unsteady flow have to be analyzed with a special emphasis on the flow frequencies from 10 Hz up to 200 Hz that could excite the door structure. In addition to the complex flow physics, the phenomena always occur near complex geometry.

1.2.1 Reproducibility

These vibrations induced by complex flow physics around complex geometries must obviously be avoided but the analysis depends entirely on the ability to reproduce the phenomenon. Three kinds of prediction are possible:

1. flight tests,
2. wind tunnel tests,
3. numerical simulations.

The first and second methods are quite efficient but are very expensive and come late in the design process. Moreover, it is difficult to reproduce in wind tunnel the true range of an aircraft in both Mach and Reynolds numbers regimes. Furthermore, the aircraft manufacturers tend to reduce the costs. The third method seems to be attractive but requires to overcome several difficulties.

1.2.2 CFD challenges

The first challenge is to handle **complex geometries** such as the landing gear systems. In fact, the full aircraft at landing or take-off condition with deployed high-lift devices and landing gears is one of the most complex configurations to deal with: they include many arms, bolts and cavities. All in all, the definition of the discrete representation of the physical domain is the first difficulty to address.

The second challenge is to convect **unsteady flow physics** over a long distance, for example between the nose and main landing gear systems. Specific numerical methods have to be developed because standard CFD methods are not able to transport such unsteady aerodynamics over a long distance without dissipation. Standard industrial computations consider a Reynolds Averaged Navier-Stokes (RANS) model in which turbulence effects are averaged in time and space. A recent industrial extension concerns Unsteady Reynolds Averaged Navier-Stokes (URANS) simulations for which unsteadiness is related to the geometry and not really to turbulence. Since pressure fluctuations need to be captured, switching to Large Eddy Simulations (LES) seems to be attractive. The key point is to capture numerically the large structures of the turbulence and to model the smallest ones. This approach is possible since large turbulent structures are linked with aircraft geometry while the smallest structures have a more general behavior and can be modeled. Note that industrialization of LES is still

today in its early stages.

The last challenge deals with the **High Performance Computing** (HPC) **capability**. Large Eddy Simulation is very intensive in terms of computation power and storage since the unsteady flow needs to be accurately computed in space and time.

These three challenges can be summarized by Fig. 1.2 and a numerical method needs to meet these three requirements at the same time to deal with such complex configurations.

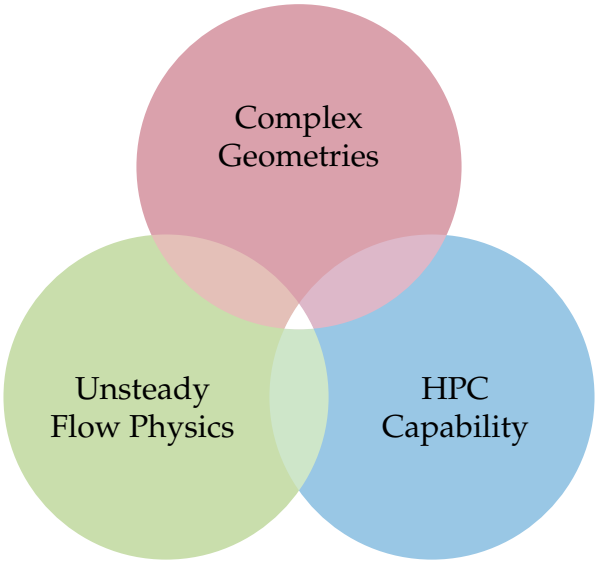


Fig. 1.2. *The three building blocks necessary for LES code. The extreme difficulty in designing a LES code meeting all three requirements at once is represented by the small white area.*

1.3 Ph.D. objective

The Ph.D. objective is more general than a simple focus on the landing gear configuration. The main objective is to find an **efficient** (in terms of HPC) and **accurate** (to capture the flow physics) **CFD method** to deal with **complex geometries** and **convection of unsteady flow over long distances**.

Before going into the details, it appears to be mandatory to be convinced by the necessity of low dispersive and dissipative methods. A quick numerical experiment was performed to compute the CPU time to obtain a given accuracy. The test case used here deals with the convection of an isentropic compressible vortex described in Sec. 4.2. The spatial discretization is achieved with the Spectral Difference Method from the second- to the sixth-order polynomial reconstruction. A second-order polynomial reconstruction corresponds to a third-order accurate scheme whereas a sixth-order polynomial reconstruction corresponds to a seventh-order accurate scheme. The CPU time is expressed in workunits. The details about the workunit determination are given in App. E. For instance, to obtain a level of accuracy of 10^{-3} on the x-component of the velocity fluctuation, it is much faster in terms of CPU time to discretize the equations

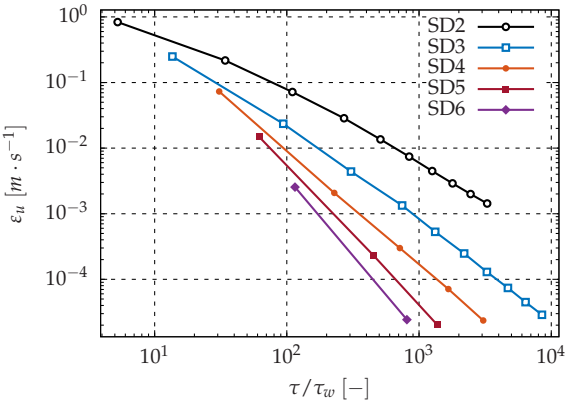


Fig. 1.3. *ℓ^2-norm of the error ε_u of x-component of the velocity fluctuation versus the computational time. The test case deals with the convection of an isentropic compressible vortex (COVO).*

with the Spectral Difference Method with a sixth-order polynomial. It perfectly illustrates the comments of Wang *et al.* [2] summarized by two beliefs. "[First belief:] High-order methods are expensive". Indeed, "[the] method efficiency based on the cost [cannot be evaluated] on the same mesh". It should be based on the cost to achieve the same error, as shown in Fig. 1.3. "[Based on this criterion], high-order methods are not necessarily expensive." "[Second belief:] High-methods are not needed for engineering accuracy". The author gives the example of the flow over a helicopter. As stated in [2], "the aerodynamic loading on the helicopter body is strongly influenced by the tip vortices generated by the rotor. These vortices travel many revolutions before hitting the body. It is critical that these vortices be resolved for a long distance in order to obtain even an engineering accuracy level prediction of the aerodynamic forces on the helicopter body." Low-order (< 3) schemes dissipate these vortices when they are convected over a long distance. This example can be of course related to the emergency descent, the jet-flap interaction or the nose landing gear/main landing gear interaction, which highlight and demonstrate the necessity of low dispersion and low dissipation schemes obtained by high-order methods. The resolution of all these challenges has been split into several steps.

1.3.1 Hybrid approach for the low-order industrial Finite Volume Method

The first step relies on the extension of a standard industrial method for design: the Finite Volume Method. The ***elsA*** solver of ONERA [3, 4], co-developed by CERFACS, implements this method to solve the Reynolds Average Navier-Stokes (RANS) equations with standard second-order schemes on structured multiblock meshes. It was extended to deal with unstructured grids in the context of URANS simulations to bring flexibility for meshing. Near the landing gears, the geometry is too complex and only unstructured grids can be considered. The key point concerns the coupling of unstructured and structured blocks and dedicated schemes inside a single computational domain. The coupling between structured and unstructured zones can be performed through the use of a Nonconforming Grid Interface (NGI) treatment. In this work, the chimera technique using overlapping grids is not retained due to the lack of conservation introduced by the interpolation of solutions between grids. Generally, structured and unstructured grids are built using different tools and enabling the same discretization on the common interface is not always possible. This is the reason why a nonconforming grid interface is the preferred solution to couple both kind of grids and the different schemes.

The **first part** of the present work will focus on the **analysis of a nonconforming grid interface**. The use of a nonconforming grid interface for **unsteady flow** has never been studied in the past and the first part of this manuscript concerns the analysis of such a technique to deal with unsteady flows using structured grids. A special emphasis is done on the **Vichnevetsky's framework** [5] which highlights the fully unsteady behavior of this kind of interface.

The **second part** of this manuscript will exhaustively validate the **hybrid approach** for the **convective fluxes** for first- and second-order schemes as well as for the **diffusive fluxes**. The validation for the convective flux is achieved with the convected vortex test case, whereas the validation for the diffusive flux is performed with the Taylor-Green vortex.

1.3.2 Towards high-order hybrid simulations

Dealing with the convection of unsteady aerodynamics over a long distance requires high-order CFD methods or at least low dispersion and low dissipation numerical methods. In 2010, Fosso [6] extended high-order schemes dedicated to the transport of turbulent structures on structured meshes in a Finite Volume formalism. This sixth-order scheme is low dissipative and low dispersive and it is therefore a good solution to compute wake vortices convection. However, where the geometry is too complex like around the nose landing gear, it is planned to use unstructured meshes. Both structured and unstructured meshes are then coupled by a nonconforming grid interface which is studied in the first part of this manuscript. In addition, this approach is validated in the second part but only for first- and second-order schemes. It

would also be interesting to be able to couple through the nonconforming grid interface both the sixth-order structured scheme and a high-order unstructured scheme to be determined. However, **high-order unstructured schemes** are much more complicated than the structured ones. This is the topic of the **third part** of this manuscript. **High-order unstructured schemes** are derived from the **standard second-order reconstruction schemes**. A Riemann solver is used after a high-order extrapolation step. This extrapolation is compacted using the gradients in each cell. The accuracy is directly driven by the accuracy of this gradient. The standard gradient formulation suffers from a lack of accuracy for general unstructured grids. This is why a chapter is devoted to the development of an **Unstructured Interface Gradient** (UIG) which is second-order accurate on general unstructured grids. The aim is to improve actual **diffusion schemes** and to find a **high-order unstructured compact scheme for the convection term**. However, this kind of formulation recovers the ***k*-exact** formulation. Indeed, to increase the order of accuracy of any numerical schemes, one needs information on an extended stencil even if it can be informatically reduced. This kind of formulation can lead to a loss of HPC efficiency which has led to study more promising high-order methods such as the Spectral Difference Method.

1.3.3 The Spectral Differences Method on unstructured grids

The Spectral Difference Method is a promising approach to discretize the Navier-Stokes equations. It belongs to Spectral Discontinuous methods: quantities are represented in any grid cell as a polynomial and since no continuity is required at cell interface, a Riemann solver accounts for the discontinuity. CERFACS has been involved for four years in the development of a new solver called *JAGUAR* [7] that solves the Navier-Stokes equations using the Spectral Difference paradigm. The mesh is unstructured and composed of hexahedra which allows to mesh complex geometries. The user can choose the accuracy and so far, simulations from third to seventh order have been performed. In the **last part** of this manuscript, the **spectral analysis** for the **Spectral Difference Method** was revisited and this approach leads to low-dissipation and low-dispersion schemes which are suitable for Large Eddy Simulations. It is the prerequisite to define the number of grid points as a function of turbulent structures to capture. With the Spectral Difference Method, polynomials are built from polynomial interpolation using the data at degrees of freedom. As a consequence, several degrees of freedom are located inside any grid cell. Compared with standard Finite Volume approach with one degree of freedom per mesh cell, the Spectral Difference approach needs less refined meshes with larger cells. Moreover, the CPU time is shown to be only weakly sensitive to the polynomial degree leading to smaller CPU time to obtain the same accuracy compared with low-order methods.

1.3.4 The Lattice-Boltzmann Method

The last promising method is the Lattice-Boltzmann Method. Both previous approaches deal with the discretization of the Navier-Stokes equations. Formally, the Navier-Stokes equations can be asymptotically [8] recovered from the Boltzmann equation [9, 10]. The Boltzmann equation represents the behavior of atoms and molecules. Introduced by McNamara *et al.* [11], an alternative consists in solving the Lattice-Boltzmann equations and macroscopic quantities (density, velocity, temperature) are recovered using quadrature [12, 13]. This method is based on a Cartesian mesh generated on the fly. This mesh intersects a surface tessellation of the geometry and dedicated treatments account for the wall. More details about the method can be found in [14]. This numerical method is very promising since it leads to low dissipation schemes suitable for Large Eddy Simulation (LES) [15]. Furthermore, the non body-fitted Cartesian meshes usable by Lattice-Boltzmann solvers allow to easily mesh very complex geometries. However it suffers from many problems due to its relative youth which include the resolution transition [16] and a low-Mach number formulation [17]. Computations on confidential configurations have been carried out by the author acting as a mere user. The

work is still ongoing and will not be presented in this manuscript for confidentiality reasons.

1.4 Publications

1.4.1 Papers

This Ph.D. work led to several publications:

[18] J. Vanharen, G. Puigt, and M. Montagnac. Theoretical and numerical analysis of nonconforming grid interface for unsteady flows. *J. Comput. Phys.* 285 (2015), 111–132. (Cited on page 13). http://dx.doi.org/10.1016/j.jcp.2015.01.013.

[66] G. Puigt, P. Cayot, J. Vanharen, and P. Sagaut. Unstructured interface gradient for the diffusion operator. *Under preparation for J. Sci. Comput.* (2017). (Cited on pages 99, 114).

[134] J. Vanharen, G. Puigt, X. Vasseur, J.-F. Boussuge, and P. Sagaut. Revisiting the spectral analysis for high-order spectral discontinuous methods. *J. Comput. Phys.* 337 (2017), 379–402. (Cited on page 129). http://dx.doi.org/10.1016/j.jcp.2017.02.043.

[177] J. Vanharen and G. Puigt. Optimized Runge-Kutta time integration for Spectral Difference Method. *Under preparation for Journal of Computational Physics* (2017). (Cited on page 167).

[179] V. Joncquières, G. Puigt, and J. Vanharen. About the solution initialization for the Spectral Difference Method. *Under preparation for Journal of Computational Physics* (2017). (Cited on page 167).

1.4.2 Conferences

This Ph.D. work led to several conference papers:

[19] J. Vanharen, G. Puigt, and M. Montagnac. Two-dimensional spectral analysis of nonconforming grid interface. Emphasis on unsteady flows. 50th 3AF International Conference on Applied Aerodynamics. (FP36-2015-vanharen). Toulouse, France. (2015). (Cited on page 13). http://www.cerfacs.fr/~cfdbib/repository/TR_CFD_15_23.pdf.

[135] J. Vanharen, G. Puigt, X. Vasseur, J.-F. Boussuge, and P. Sagaut. Revisiting the spectral analysis for high-order spectral discontinuous methods. TILDA - Symposium & Workshop on Industrial LES & DNS. Toulouse, France. (2016). (Cited on page 129). https://www.researchgate.net/publication/310606899_Revisiting_the_spectral_analysis_for_high-order_spectral_discontinuous_methods.

[136] J. Vanharen, G. Puigt, X. Vasseur, J.-F. Boussuge, and P. Sagaut. Comparing the resolution power of standard finite difference and spectral difference schemes. TILDA - Symposium & Workshop on Industrial LES & DNS. Toulouse, France. (2016). (Cited on page 129). https://www.researchgate.net/publication/310606902_Comparing_the_resolution_power_of_standard_finite_difference_and_spectral_difference_schemes.

[178] J. Vanharen, G. Puigt, J.-F. Boussuge, and A. Balan. Optimized Runge-Kutta time integration for Spectral Difference Method. 23rd AIAA Computational Fluid Dynamics Conference, 2017 AIAA Aviation and Aeronautics Forum and Exposition. Denver, Colorado, USA. (2017). (Cited on page 167).

II Nonconforming grid interface analysis

2 Nonconforming grid interface 13
2.1 Introduction and problem description
2.2 Theoretical aspects
2.3 On stability and accuracy of nonconforming interface
2.4 High-order metric-dependent interpolation for Riemann solver
2.5 Conclusion

3 Vichnevetsky's framework 41
3.1 Introductory numerical experiment
3.2 Spectral analysis
3.3 Stability analysis
3.4 Necessity of Vichnevetsky's framework
3.5 Conclusion

2 Nonconforming grid interface

"It is paradoxical, yet true, to say, that the more we know, the more ignorant we become in the absolute sense, for it is only through enlightenment that we become conscious of our limitations. Precisely one of the most gratifying results of intellectual evolution is the continuous opening up of new and greater prospects."

— Nikola Tesla (1856 - 1943)

Abstract. *In this chapter, a framework is proposed for the spectral analysis of the numerical scheme applied on a nonconforming grid interface between two structured blocks for the two-dimensional advection equation, in a cell-centered finite-volume formalism. The conservative flux computation on the nonconforming grid interface is based on the sum of partial fluxes computed with the same numerical scheme used for a standard cell interface. This framework is used to analyze the effect of grid refinement or coarsening on the stability of the second-order centered scheme. New theoretical results are given and compared to numerical results. Considering the convection of a two-dimensional isentropic compressible vortex, the refinement/coarsening are shown to be the cause of instabilities, poor accuracy and reflection of high-frequency waves. A new approach to compute partial fluxes, which is based both on a high-order extrapolation that accounts for the local metric and on a Riemann solver, is then proposed to reduce spurious modes. This chapter led to a paper published in the Journal of Computational Physics [18] and to a conference paper presented at the 3AF conference [19].*

2.1 Introduction and problem description

Computational Fluid Dynamics (CFD) based on structured grids (multiblock structured mesh and dedicated schemes) has entered the aerospace industry for 2D flows and then for 3D flows some decades ago. People were looking for predictive tools for the analysis of flows around quite simple geometry. In this context, many efforts were devoted to the Reynolds-Averaged Navier-Stokes (RANS) equations. During the last 10 years, the focus was made on increasing solver robustness and on accounting for a more complex geometry. These two points must be analyzed together. On the one hand, complex multiblock topology is mandatory to account for complex geometry, playing with the basic grid topologies (H-, O-, or C-grids). On the other hand, as the mesh is not expected to have a high quality in the whole computational domain, the solver robustness plays an important role to guarantee a fast convergence of the computation. Nowadays, experience and best practices give a high confidence in RANS CFD solutions. For example, the main part of the aircraft flight envelope is accurately computed with RANS codes, and CFD simulations can help optimize design.

Two main drawbacks characterize the structured approach. The first one is the time spent to generate a mesh. Except for very simple geometry for which the blocking (division of the computational domain in structured zones) can be predefined, this can take several weeks as in the case of a high-lift configuration. Many techniques exist to ease the mesh generation process

such as the unstructured approach [20], the overset method or Chimera method [21, 22], the hybrid [4, 23] or DRAGON grid [24] approaches, and the nonconforming grid interface [25, 26, 27, 28]. The second drawback is the following: structured grids are well adapted to cope with flow anisotropy, and provide a way to align the flow physics along some particular mesh lines. But for many geometries, the grid refinement near boundary layers extends to downstream areas as shown in Fig. 2.1. Then, many grid points are located in regions that are sometimes of low interest for the computation of the flow.

The meshing technique based on nonconforming grid interface (NGI) addresses both points. The mesh sizes of two blocks separated by a NGI can be different. Mesh lines can therefore be discontinuous at the interface. On the one hand, the number of useless cells can be limited, and the time to perform the computation reduced. For aerospace applications, a typical difference in order of magnitude between cell aspect ratios on both sides of the interface may reach 100. A typical industrial mesh for the computation of steady lift and drag coefficients is shown in Fig. 2.1. The key point for the computation of these coefficients is a good capture of the wing boundary layer. The steady wake from the wing only has a very small influence on the overall coefficients. The mesh is composed of a well refined region of high interest (boundary layer around the wing in Fig. 2.1) and a coarsen region of low interest (wake of the wing in Fig. 2.1). It is therefore acceptable that the low-interest region suffers from a lack of accuracy. Nonetheless, the high-interest region must not suffer from a lack of accuracy because of the adjacent coarsen block and/or the nonconforming grid interface. On the other hand, each block on both sides of the interface can be meshed separately, and therefore the meshing process is far more easier. The two previous points can be extended to unsteady simulations. In particular, meshes used for steady flows can also be used for unsteady flows, especially in industry.

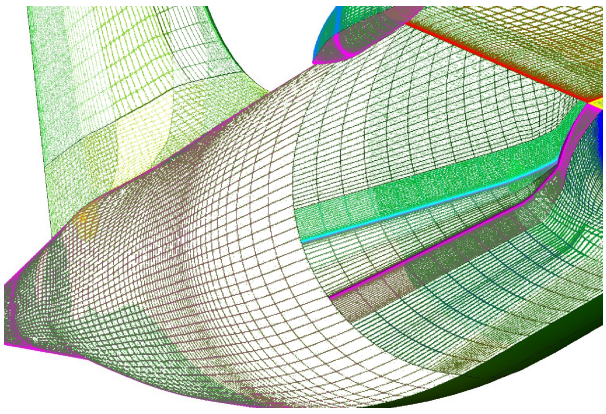


Fig. 2.1. *Nonconforming grid interface after the C-grid around the wing.*

The nonconforming grid interface was firstly introduced by Rai [25, 29, 30], with a conservative treatment of fluxes, for Euler equations in a finite-difference context. The same author extended this technique to the unsteady thin-layer Navier-Stokes equations, but with a nonconservative treatment, to compute the flow within an axial turbine stage [31, 32]. Afterwards, many authors extended this technique to different numerical schemes and other configurations, either with a conservative or a nonconservative approach. As a few examples, it was used to compute the steady flow around the F-18 forebody with actuated control strake [33], and around a generic wing [34]. Rumsey [35] used the nonconforming grid interface to compute acoustic waves through sliding-zone interfaces. This technique was also applied to study wing control surfaces using three-dimensional RANS equations [36]. Stability of nonconforming grid interface for the steady compressible Euler equations was studied by Lerat and Wu [37]. However, the influence of the nonconforming grid interface on an unsteady flow has never been finely studied in the past.

In practice, the coarsening of a block relative to another block can occur in the tangential and/or normal directions to the zonal boundary. It will be shown that a large coarsening ratio will cause spurious mode reflection. These modes can have an amplitude as high as the initial wave. A lot of solutions are considered in the literature to lower or remove spurious waves in general.

The first type of methods improves the spectral properties of the basic scheme. Berland *et al.*[38] proposed explicit high-order numerical schemes based on Taylor series expansions for

the accurate computation of multiple-scale problems and for the implementation of boundary conditions. They showed that, applied on a non-uniform grid, a scheme designed with metric information gives much better results than a scheme designed without metric information. Lele [39], Kim and Lee [40], Ashcroft and Zhang [41], Nance *et al.* [42], Sengupta *et al.* [43], and Fosso *et al.* [6] developed compact schemes to improve the spectral behavior of numerical schemes. They compare compact schemes with well-known schemes based on Taylor series expansions. Fosso *et al.* [6] take into account the local metric with an isoparametric formulation. Bogey and Bailly [44], Tam and Webb [45], and Popescu *et al.* [46] propose explicit numerical methods by minimizing the dispersion and the dissipation errors in the wavenumber space to obtain the so-called Dispersion-Relation-Preserving (DRP) schemes for computing flow and noise with a high accuracy.

The second type of methods concerned by spurious waves in general is related to boundary conditions. When solving a partial differential equation numerically, the energy must remain constant or decrease, but it must not increase. Otherwise, the computation errors could grow arbitrarily fast. The fulfillment of this principle can be verified by the energy method which consists in an integration by parts of partial differential equation. Once this equation is discretized, it is replaced by a Summation By Parts (SBP), first introduced by Kreiss and Scherer [47]. This kind of stability analysis takes into account boundary conditions thanks to the integration by parts. Then, this method was improved by Strand [48] and Olsson [49], and more recently by Bodony *et al.* [50] and Carpenter *et al.* [51]. Based on the work of Thompson [52], Poinsot and Lele [53] introduced the Navier-Stokes Characteristic Boundary Conditions (NSCBC) to reduce instabilities and to avoid high-frequency wave reflection. This boundary condition is deduced from characteristic lines of hyperbolic systems, and is based on the analysis of the different waves crossing the boundary. This method was also studied by Kim and Lee [54, 55].

The goal of this chapter is to provide new theoretical and numerical results regarding the behavior of the nonconforming grid interface between two structured blocks for a two-dimensional unsteady flow in a cell-centered finite-volume formalism. A theoretical stability analysis is difficult to perform on the full Navier-Stokes equations. So, the focus is put on two hyperbolic systems, namely the advection equation and Euler equations. This chapter gives a general framework for the analysis of refinement/coarsening effects when using nonconforming grid interfaces and classical low-order industrial schemes. A second-order central scheme is considered, as in [56] for Large Eddy Simulation. Naturally, Navier-Stokes equations would be considered for industrial flows, but the treatment of the non-linear diffusion scheme is not addressed here.

The chapter is organized as follows. A classical 1-D Fourier analysis is performed in Sec. 2.2 after the introduction of the linear convection problem, notations, and the toy problem. Next, it will also explain how to analyze the numerical treatment of the nonconforming grid interface in two dimensions. Sec. 2.3 describes the test case used to perform numerical simulations, and describes in detail the stability and accuracy analysis. The coarsening of one block on the one side of the nonconforming grid interface is particularly investigated. Inspired by the NSCBC theory, an approximate Riemann solver is proposed in Sec. 2.4 to make disappear or, at least, to limit wave reflection. The solution is based on two extrapolations of data on both sides of the interface and on an approximate Riemann solver to account for the discontinuity. The Roe's solver [57, 58] is chosen since it is used for industrial computations. Conclusions are given in Sec. 2.5.

2.2 Theoretical aspects

2.2.1 Model problem

Let us consider the advection equation (2.1) for the one-dimensional problem where c is a constant advective velocity, f is the unknown, and $\boldsymbol{F}$ is the flux vector defined by $\boldsymbol{F}(x,y,z) = (cf,0,0)^\top$.

$$\frac{\partial f}{\partial t} + \nabla \cdot \boldsymbol{F} = 0 \tag{2.1}$$

The linear hyperbolic partial derivative equation (2.1) is used to study the behavior of numerical schemes. Before analyzing the interface treatment, let us recall a classical property of the finite-volume discretization.

2.2.2 Spatial discretization

Let $\boldsymbol{F}(x,y,z)$ be a continuously differentiable vector field. Let Ω be an open set, and let $\partial\Omega$ be its boundary. Let $\boldsymbol{n}$ be the unit outward normal vector of $\partial\Omega$. Integrating (2.1) on Ω, it comes:

$$\int_\Omega \frac{\partial f}{\partial t}\,\mathrm{d}\mathcal{V} + \int_{\partial\Omega} \boldsymbol{F}\cdot\boldsymbol{n}\,\mathrm{d}\mathcal{S} = 0. \tag{2.2}$$

Just for sake of clarity, the stencil for computing the flux is presented for a Cartesian grid in Fig. 2.2.

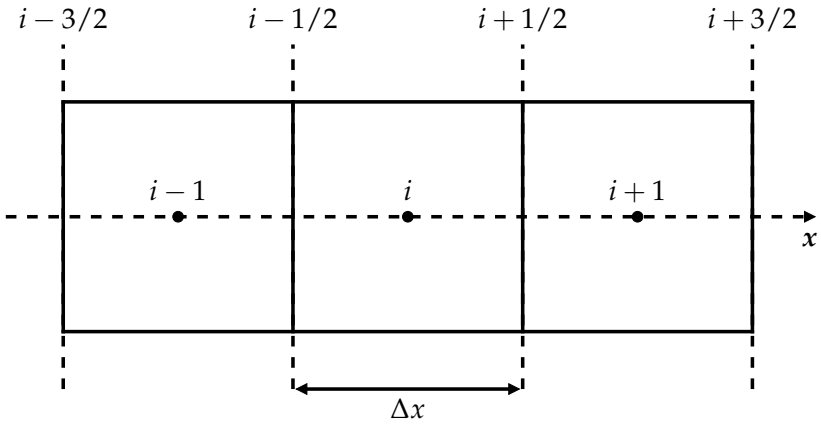


Fig. 2.2. *Stencil for the Finite Volume Method.*

Ω represents the cell i. Both the cell volume $\mathcal{V}$ and the interface area $\mathcal{S}$ are assumed to be constant. Δx is the length of any cell in the x-direction. Discretizing (D.2) over a grid of square cells, and introducing the mean value $\overline{f}$ of f over Ω, one obtains

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\mathcal{V}\overline{f}\right)_i + (c\mathcal{S}f)_{i+1/2} - (c\mathcal{S}f)_{i-1/2} = 0, \tag{2.3}$$

and

$$\frac{\mathrm{d}}{\mathrm{d}t}\overline{f}_i + c\cdot\frac{f_{i+1/2}-f_{i-1/2}}{\Delta x} = 0, \tag{2.4}$$

where interface values $f_{i+1/2}$ and $f_{i-1/2}$ are unknown. To evaluate them from the mean value $\overline{f}_i$, a numerical scheme links interface values with cell-centered data. In the following, a second-order centered scheme is considered. The numerical mean value of the partial derivative of f with respect to x is consequently evaluated by

$$\left.\overline{\frac{\partial f}{\partial x}}\right|_i = \frac{f_{i+1/2}-f_{i-1/2}}{\Delta x} = \frac{\overline{f}_{i+1}-\overline{f}_{i-1}}{2\Delta x}. \tag{2.5}$$

Formally, (2.5) can be recovered using a simple finite-difference scheme when replacing mean quantities over each control volume by local values. In the following and for the sake of clarity, the sign (-) representing a mean value is omitted.

Now, we consider the two-dimensional toy mesh represented in Fig. 2.3. It is made of two blocks, the left one and the right one, which are separated by a block interface. This interface can be made of conforming or nonconforming nodes. Spacings are fixed in each block. Focusing on the colored cells, two parameters, h and Δz, will play an important role. h measures the influence of the opposite cells on the flux balance, while Δz is a parameter for mesh sizes. The L (resp. R) subscript refers to a quantity whose value is defined on left (resp. right) side of the interface, following Fig. 2.3. Of course, a classical conforming interface is recovered if $h = 0$ and $\Delta z_R = \Delta z_L$. Note also that $h \in [0, \Delta z_L/2]$.

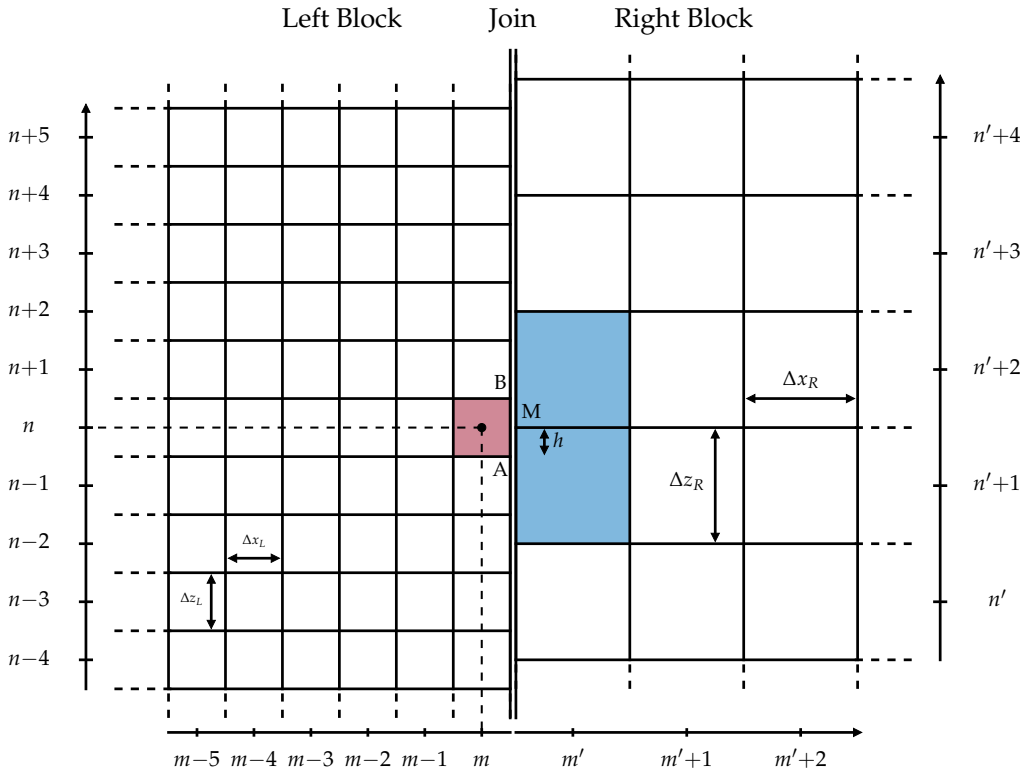


Fig. 2.3. *Toy mesh composed of two blocks: the left one ($\Delta x_L, \Delta z_L$) and the right one ($\Delta x_R, \Delta z_R$). The right block is shifted by h along the vertical axis.*

In the following, a local Fourier, or Von Neumann, analysis is performed to characterize the effect of the nonconforming grid interface. But before entering into details, many aspects regarding the nonconforming grid interface and its numerical discretization have to be explained.

In a general framework, the nonconforming grid interface only exists in two or three dimensions, and it is then mandatory to analyze the grid interface effects following the tangential and normal directions of the NGI. Here, the analytical expressions of the partial derivatives of f are defined in two dimensions, and the Fourier analysis along the x- and z-axis concerns the cell (m, n).

The procedure is established for the left block, but the same approach leads to similar relations for the right block. The analytical expressions are given in the general case for a nonconforming interface. In the following, $\boldsymbol{k}$ is the wave vector, and $k_x = \|\boldsymbol{k}\| \cdot \cos(\alpha)$ (resp. $k_z = \|\boldsymbol{k}\| \cdot \sin(\alpha)$) is the projection of $\boldsymbol{k}$ on the x- (resp. z-) axis, and α represents

the angle between $\boldsymbol{k}$ and the x-axis. In order to perform a spectral analysis, the harmonic function $f = \exp\left[j\left(k_x \cdot x + k_z \cdot z\right)\right]$, where j is the imaginary unit ($j^2 = -1$), is assumed to be solution of (2.1). The discretized value of f for the cell (m, n) of the left block is $f_{m,n} = \exp\left[j\left(k_x \cdot m\Delta x_L + k_z \cdot n\Delta z_L\right)\right]$. According to (2.5), the corresponding partial derivative of f with respect to x is evaluated by:

$$\left.\frac{\partial f}{\partial x}\right|_{m,n} = \frac{f_{m+1/2,n} - f_{m-1/2,n}}{\Delta x_L}, \tag{2.6}$$

where the interface values $f_{m+1/2,n}$ and $f_{m-1/2,n}$ are still unknown. By applying the second-order centered scheme, one obtains:

$$f_{m-1/2,n} = \frac{f_{m,n} + f_{m-1,n}}{2}. \tag{2.7}$$

The other term $f_{m+1/2,n}$ is more complex to assess because of the nonconforming grid interface. Following notations defined in Fig. 2.3, $f_{m+1/2,n}$ is defined by (2.8) to ensure the conservation of fluxes. The interface flux accounts for two integrals, on AM and MB respectively, and the flux through AB is by linearity of integration the sum of both contributions:

$$f_{m+1/2,n} = \frac{AM}{AB} \cdot f_{AM} + \frac{MB}{AB} \cdot f_{MB}. \tag{2.8}$$

Then, by applying the centered scheme, f_{AM} and f_{MB} are evaluated by:

$$f_{AM} = \frac{f_{m,n} + f_{m',n'+1}}{2}, \tag{2.9}$$

$$f_{MB} = \frac{f_{m,n} + f_{m',n'+2}}{2}. \tag{2.10}$$

By injecting (2.7), (2.8), (2.9) and (2.10) into (2.6), one obtains:

$$f'_{m,n} := \left.\frac{\partial f}{\partial x}\right|_{m,n} = \frac{f^*_{m+1,n} - f_{m-1,n}}{2\Delta x_L}, \tag{2.11}$$

where

$$f^*_{m+1,n} = \frac{AM}{AB} \cdot f_{m',n'+1} + \frac{MB}{AB} \cdot f_{m',n'+2}. \tag{2.12}$$

By injecting the analytical expression of $f_{m,n}$ into (2.12), one obtains:

$$\begin{aligned} f^*_{m+1,n} = \left(1 - \frac{h}{\Delta z_L}\right) \exp\left[jk_x\left((m+\frac{1}{2})\Delta x_L + \frac{\Delta x_R}{2}\right) + jk_z\left(n\Delta z_L + \frac{\Delta z_R}{2} - \frac{\Delta z_L}{2} + h\right)\right] \\ + \frac{h}{\Delta z_L} \exp\left[jk_x\left((m+\frac{1}{2})\Delta x_L + \frac{\Delta x_R}{2}\right) + jk_z\left(n\Delta z_L - \frac{\Delta z_R}{2} - \frac{\Delta z_L}{2} + h\right)\right]. \end{aligned} \tag{2.13}$$

The partial derivative of f with respect to z can be easily computed since on the Cartesian grid, the z-derivative only needs information from one side of the nonconforming grid interface. We note $\mathrm{sinc}(x) := \sin(x)/x$. Considering again the toy mesh in Fig. 2.3 and the harmonic function f previously introduced,

$$\left.\frac{\partial f}{\partial z}\right|_{m,n} = \frac{f_{m,n+1} - f_{m,n-1}}{2\Delta z_L} = jf_{m,n} \cdot k_z \mathrm{sinc}\left(k_z \Delta z_L\right). \tag{2.14}$$

2.2.3 Basic concepts of spectral analysis

The modified wavenumber $\tilde{k}_x$ is defined by $\tilde{k}_x := f'_{m,n}/(jf_{m,n})$. $f'_{m,n}$ and $f_{m,n}$ are complex *a priori*, but $|f_{m,n}| = 1$. $\mathcal{R}e$ (resp. $\mathcal{I}m$) representing the real (resp. imaginary) part of complex number,

$$\tilde{k}_x = \frac{\mathcal{R}e\left(f'_{m,n}\right) + j\mathcal{I}m\left(f'_{m,n}\right)}{j\mathcal{R}e\left(f_{m,n}\right) - \mathcal{I}m\left(f_{m,n}\right)} = -\mathcal{R}e\left(f'_{m,n}\right)\mathcal{I}m\left(f_{m,n}\right) + \mathcal{I}m\left(f'_{m,n}\right)\mathcal{R}e\left(f_{m,n}\right) - j\left[\mathcal{R}e\left(f'_{m,n}\right)\mathcal{R}e\left(f_{m,n}\right) + \mathcal{I}m\left(f'_{m,n}\right)\mathcal{I}m\left(f_{m,n}\right)\right]. \quad (2.15)$$

Once (2.15) is implemented, it allows to study the effect of the parameters α, h, Δx_L, Δx_R, Δz_L and Δz_R on the real and imaginary parts of the modified wavenumber.

If $k_z = 0$, then (2.13) is independent from h and Δz_R, the modified wavenumber is only a function of Δx_L and Δx_R.

According to the Nyquist-Shannon sampling theorem, *If a function $x(t)$ contains no frequencies higher than B hertz, it is completely determined by giving its ordinates at a series of points spaced $1/(2B)$ seconds apart*. This theorem leads to (2.16) where f_s is the sampling frequency and f the frequency of the signal:

$$f_s \geqslant 2f. \quad (2.16)$$

For a one-dimensional wave of wavenumber k, one obtains:

$$\frac{1}{\Delta x} \geqslant 2 \cdot \frac{k}{2\pi} \Leftrightarrow k \cdot \Delta x \leqslant \pi.$$

Indeed, $k \cdot \Delta x$ should belong to 0 and π. If $\Delta x = 1$, higher wavenumbers than π are not resolved. More specifically, since the nonconforming grid interface only exists at least in two dimensions, both following equations must be satisfied: $k\cos\alpha \cdot \Delta x \leqslant \pi$ and $k\sin\alpha \cdot \Delta z \leqslant \pi$. But for sake of clarity, we want to satisfy these equations whatever the value of α. These equations become: $k \cdot \Delta x \leqslant \pi$ and $k \cdot \Delta z \leqslant \pi$.

2.2.4 Specific case: uniform grid

For a uniform grid where $\Delta x_L = \Delta x_R = \Delta x$, $\Delta z_L = \Delta z_R = \Delta z$, and when α and h are different from 0, one obtains with (2.11), (2.12), and (2.13):

$$\left.\frac{\partial f}{\partial x}\right|_{m,n} = \frac{f_{m,n}}{2\Delta x}\left[\left(1 - \frac{h}{\Delta z}\right)\exp\left[jk_x\Delta x + jk_z h\right] + \frac{h}{\Delta z}\exp\left[jk_x\Delta x - jk_z\Delta z + jk_z h\right] - \exp\left[-jk_x\Delta x\right]\right]. \quad (2.17)$$

Using (2.17), the modified wavenumber (2.18) can be calculated, and its real and imaginary parts are respectively given by (2.19) and (2.20).

$$\tilde{k}_x = \frac{1}{2j\Delta x}\left[(1 - \frac{h}{\Delta z})\exp\left[jk_x\Delta x + jk_z h\right] + \frac{h}{\Delta z}\exp\left[jk_x\Delta x - jk_z\Delta z + jk_z h\right] - \exp\left[-jk_x\Delta x\right]\right], \quad (2.18)$$

$$\mathcal{R}e\left(\tilde{k}_x\right) = \frac{1}{2\Delta x}\left[(1 - \frac{h}{\Delta z})\sin\left(k_x\Delta x + k_z h\right) + \frac{h}{\Delta z}\sin\left(k_x\Delta x - k_z(\Delta z - h)\right) + \sin\left(k_x\Delta x\right)\right], \quad (2.19)$$

$$\mathcal{I}m\left(\tilde{k}_x\right) = -\frac{1}{2\Delta x}\left[(1 - \frac{h}{\Delta z})\cos\left(k_x\Delta x + k_z h\right) + \frac{h}{\Delta z}\cos\left(k_x\Delta x - k_z(\Delta z - h)\right) - \cos\left(k_x\Delta x\right)\right]. \quad (2.20)$$

For $k_z = 0$, the standard expression for a one-dimensional equation, $\tilde{k}_x = k_x \cdot \mathrm{sinc}(k_x \Delta x)$, is recovered. The spectral analysis is classically performed for a one-dimensional problem. However, we need to extend it in two dimensions since the nonconforming grid interface only exists at least in two dimensions.

2.2.5 Extension of spectral analysis in two dimensions

Let us consider the two-dimensional advection equation (2.21), where $\mathbf{c} = (c_x, 0, c_z)^\top$ is a constant advective velocity, f is the unknown, and $\boldsymbol{F}$ is the flux vector defined by $\boldsymbol{F}(x, y, z) = (c_x f, 0, c_z f)^\top$:

$$\frac{\partial f}{\partial t} + \nabla \cdot \boldsymbol{F} = 0. \tag{2.21}$$

Let us inject a space and time harmonic wave $f(x, y, z, t) = \exp\left[j\left(k_x x + k_z z - \omega t\right)\right]$ in (2.21), where $\boldsymbol{k} = (k_x, 0, k_z)^\top$ is the wave vector, and ω the angular frequency. The exact dispersion relation (2.22) is obtained.

$$\omega = c_x k_x + c_z k_z. \tag{2.22}$$

The same kind of relation can be computed using approximated derivatives previously introduced. If the complex numbers $\tilde{k}_x$ and $\tilde{k}_z$ are defined as before, one obtains:

$$\tilde{\omega} = c_x \tilde{k}_x + c_z \tilde{k}_z. \tag{2.23}$$

General case.
This relation depends on the modified wavenumber which is a function of local metric. The method that follows is directly inspired from Berland *et al.* [38]. Take a harmonic wave $f(x, y, z, t = 0) = \exp\left[j\left(k_x x + k_z z\right)\right]$ as the initial disturbance. The analytical solution of (2.21) is given by:

$$f(x, y, z, t) = \exp\left[j\left(k_x x + k_z z - \omega t\right)\right].$$

The solution of (2.21) computed using approximated derivatives is given by:

$$\tilde{f}(x, y, z, t) = \exp\left[j\left(k_x x + k_z z - \tilde{\omega} t\right)\right]. \tag{2.24}$$

According to the dispersion relation (2.23), (2.24) becomes:

$$\begin{aligned}\tilde{f}(x, y, z, t) &= \exp\left[j\left(k_x x + k_z z - \left(c_x \tilde{k}_x + c_z \tilde{k}_z\right) t\right)\right] \\ &= \exp\left[j\left(k_x x + k_z z - \omega t\right)\right] \cdot \exp\left[jt\left(c_x k_x + c_z k_z - \mathcal{R}e\left[c_x \tilde{k}_x + c_z \tilde{k}_z\right]\right)\right] \cdot \\ &\qquad \exp\left[\mathcal{I}m\left[c_x \tilde{k}_x + c_z \tilde{k}_z\right] t\right].\end{aligned}$$

If the field is homogeneous and isotropic ($c_x = c_z = c$), one obtains:

$$\tilde{f}(x, y, z, t) = A(ct) \cdot \exp\left[j\pi\phi(ct)\right] \cdot \exp\left[j\left(k_x x + k_z z - \omega t\right)\right]. \tag{2.25}$$

$A(ct)$ and $\phi(ct)$ are given by:

$$\begin{aligned}A(ct) &= \exp\left[\mathcal{I}m\left[\tilde{k}_x + \tilde{k}_z\right] ct\right], \\ \phi(ct) &= \frac{k_x + k_z - \mathcal{R}e\left[\tilde{k}_x + \tilde{k}_z\right]}{\pi} ct.\end{aligned}$$

The phase error is given by $\phi(\Delta x)$, and the wave amplitude is multiplied by $A(\Delta x)$ for a right-running wave propagating over an arbitrary distance $\Delta x = ct$. $\phi(\Delta x)$ quantifies the dispersion error and $A(\Delta x)$ represents the amplitude evolution. There is dispersion if $\phi(\Delta x) \neq 0$, amplification if $A(\Delta x) > 1$, and finally, dissipation if $A(\Delta x) < 1$. These quantities generalize the concept of the one-dimensional modified wavenumber in two dimensions.

Application to the left and right blocks.
At this point, only general analytical expressions of derivatives for a cell which belongs to a block have been provided, and this is not sufficient to characterize the nonconforming grid interface effects. The time integration has to be taken into account when the unsteady flow is going through the nonconforming grid interface. To highlight the characteristic variables, let us consider the dispersion relation (2.23). Its expression for the left block is:

$$\tilde{\omega} = c\left(\tilde{k}_x^L + \tilde{k}_z^L\right), \tag{2.26}$$

where $\tilde{k}_x^L$ and $\tilde{k}_z^L$ are the modified wavenumbers computed for the left block. Let us introduce notations $A^L(\Delta x)$ and $\phi^L(\Delta x)$:

$$A^L(\Delta x) = \exp\left[\mathcal{I}m\left[\tilde{k}_x^L + \tilde{k}_z^L\right]\Delta x\right],$$
$$\phi^L(\Delta x) = \frac{k_x^L + k_z^L - \mathcal{R}e\left[\tilde{k}_x^L + \tilde{k}_z^L\right]}{\pi}\Delta x.$$

In the same manner, the expression for the right block is:

$$\tilde{\omega} = c\left(\tilde{k}_x^R + \tilde{k}_z^R\right), \tag{2.27}$$

where $\tilde{k}_x^R$ and $\tilde{k}_z^R$ are the modified wavenumbers computed for the right block. Let us introduce notations $A^R(\Delta x)$ and $\phi^R(\Delta x)$:

$$A^R(\Delta x) = \exp\left[\mathcal{I}m\left[\tilde{k}_x^R + \tilde{k}_z^R\right]\Delta x\right],$$
$$\phi^R(\Delta x) = \frac{k_x^R + k_z^R - \mathcal{R}e\left[\tilde{k}_x^R + \tilde{k}_z^R\right]}{\pi}\Delta x.$$

Left and right block coupling.
Breakup the analytic time-integration into the left and right parts sequentially to show the combined effect of the wave crossing the interface, the overall effect of the nonconforming interface is composed of a contribution of both left and right blocks:

$$\tilde{f}(t,x,z) = \underbrace{A^L \exp\left[j\pi\phi^L\right]}_{\text{Left contribution}} \cdot \underbrace{A^R \exp\left[j\pi\phi^R\right]}_{\text{Right contribution}} \cdot \exp\left[j\left(k_x x + k_z z - \omega t\right)\right].$$

In the previous two paragraphs, Δx was an arbitrary distance. To couple left and right block, Δx is chosen to be equal to $\min(\Delta x_L, \Delta x_R)$.

Specific case: uniform grid.
If the grid is uniform, $\Delta x_L = \Delta x_R = \Delta x$ and $\Delta z_L = \Delta z_R = \Delta z$, the overall effect of the nonconforming interface is characterized by $A^{NGI}(\Delta x)$ and $\phi^{NGI}(\Delta x)$. They can be seen as transfer functions and they are written as:

$$A^{NGI}(\Delta x) = A^L(\Delta x) \cdot A^R(\Delta x),$$
$$\phi^{NGI}(\Delta x) = \phi^L(\Delta x) + \phi^R(\Delta x).$$

Specific case: nonuniform grid.
If the grid is not uniform, for instance $\Delta x_L = \Delta z_L = 1$ and $\Delta x_R = \Delta z_R = 2$, it is no longer possible to compute $A^{NGI}(\Delta x)$ and $\phi^{NGI}(\Delta x)$. Indeed, for the left block, $k \cdot \Delta x \in [0, \pi]$ and for the right block, $k \cdot \Delta x \in [0, \pi/2]$ to comply the Nyquist-Shannon sampling theorem.

2.3 On stability and accuracy of nonconforming interface for the second-order centered scheme

All numerical results presented in the following are obtained with the ***elsA*** software from ONERA [3, 4]. The effect of nonconforming grid interface is studied on non-uniform grids in the general case. Theoretical results from Sec. 2.2 are compared with ***elsA*** computations in terms of stability. First of all, the test case, which is used to compare theoretical analysis with numerical solutions, is presented. Secondly, the mesh is coarsened along the x-axis. Consequently, the coarsening does not imply a nonconforming interface. Thirdly, it is coarsened along the z-axis. Then, the specific case of nonconforming grid interface is investigated on uniform grids. Finally, the mesh is coarsened along both directions, and the reflection of high-frequency waves is underlined.

2.3.1 Convection of a vortex

The main aim of this work is to compare the theoretical analysis with numerical solutions computed with the ***elsA*** software. Our test case deals with the convection of a compressible and isentropic vortex on a mean and constant speed flow ($p_0 = 101325.0\ [Pa]$, $T_0 = 300.0\ [K]$, $M_0 = U_0/\sqrt{\gamma R_{gas} T_0} = 0.1$) inspired by the High-Order Workshop [2]. The characteristic vortex radius R_c is equal to $0.1\ [m]$. The vortex intensity β is computed to obtain the desired velocity fluctuation u_{max} due to the vortex by $\beta = u_{max}\sqrt{e}/U_0$ with $u_{max} = 1.5\ [m/s]$. The fluid is assumed to be a perfect gas. The ratio of specific heats γ is equal to 1.4, and the gas constant R_{gas} is equal to $287.058\ [J/kg/K]$. The vortex is initialized around the point of coordinates $(x_c = 0.5\ [m], z_c = 0.5\ [m])$. The flow is initialized by the velocity $\boldsymbol{V} = (u, 0, w)$ and the temperature T with:

$$u = U_0 - \frac{\beta U_0}{R_c}(z - z_c)\exp\left[-\frac{r^2}{2}\right], \tag{2.28}$$

$$w = \frac{\beta U_0}{R_c}(x - x_c)\exp\left[-\frac{r^2}{2}\right], \tag{2.29}$$

$$T = T_0 - \frac{\beta^2 U_0^2}{2C_p}\exp\left[-r^2\right], \tag{2.30}$$

where $C_p = \frac{\gamma R_{gas}}{\gamma - 1}$ and $r^2 = \frac{(x-x_c)^2 + (z-z_c)^2}{R_c^2}$. Since the vortex is isentropic, the density is computed using:

$$\rho = \rho_0 \cdot \left(\frac{T}{T_0}\right)^{\frac{1}{\gamma - 1}}.$$

This flow is an analytical solution of unsteady compressible Euler equations. The expected solution is the initial vortex convected without deformation. A reference solution with a conforming interface was computed with the ***elsA*** software. The vortex is convected over one meter between two blocks. The reference computation was made on a mesh similar to that in Fig. 2.4, but with both block sizes of $(N_x = 200) \times (N_z = 400)$ where N_x is the number of nodes along the x-axis, and N_z along the z-axis. The boundary conditions are described in Fig. 2.4. Unsteady compressible Euler equations are solved on a single processor. The explicit time integration scheme is the 4th-order Runge-Kutta where the Courant-Friedrichs-Lewy number is equal to 0.1. Time steps are chosen sufficiently small in order to measure dissipation and dispersion due to the spatial scheme and not due to the temporal integration. The flux computation is performed with a second-order centered finite-volume scheme.

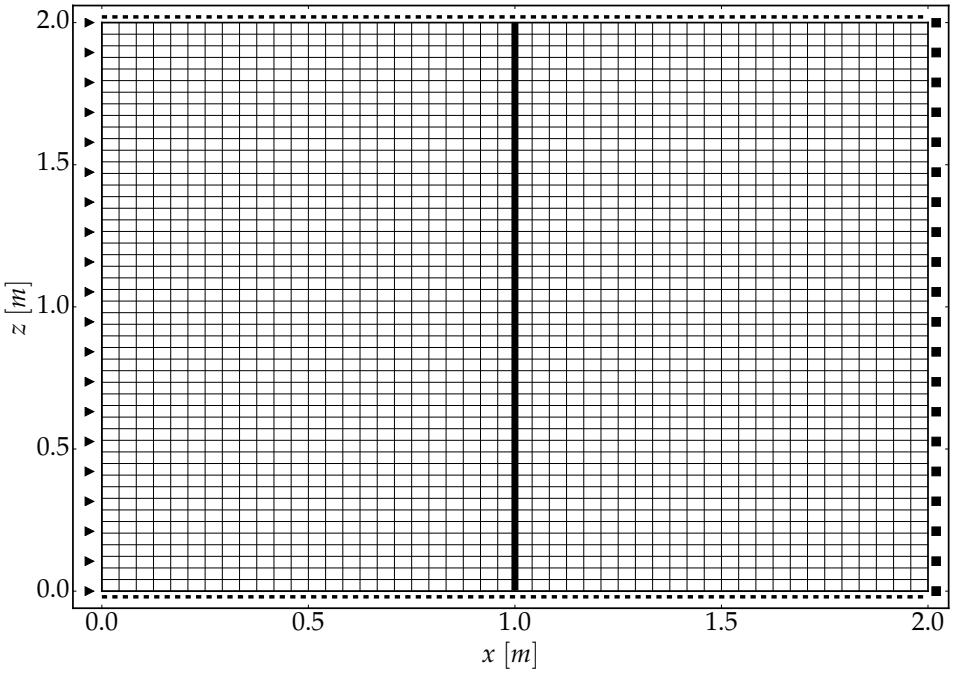


Fig. 2.4. *Coarse reference mesh and its boundary conditions:* ▶ *subsonic inlet,* ■ *subsonic outlet and* - - *periodic. The left block* (25×50) *and the right block* (25×50) *are separated by a grid interface represented by the bold line.*

An objective criterion is required in order to quantify the error between analytical solutions and ***elsA*** computations. Let f be a function. In the present work, the uniform norm will be preferred to the ℓ^2-norm. It is defined by $\|f\|_\infty = \sup\{|f(x)| : x \in S\}$, where S is a closed set. Indeed, the uniform norm gives a supremum of ℓ^2-norm, such that $\|f\|_2 \leqslant K \cdot \|f\|_\infty$ with K a constant (that depends on the space measure). It is clear that any function tending to 0 for the uniform norm tends to 0 for the ℓ^2-norm over a bounded space. Thereafter, to compare analytical solution with ***elsA*** computation, we define the error ξ and the relative error η of a quantity f by:

$$\xi = \left\| f_{analytical} - f_{computed} \right\|_\infty,$$
$$\eta = \frac{\left\| f_{analytical} - f_{computed} \right\|_\infty}{\left\| f_{analytical} \right\|_\infty}.$$

in Fig. 2.5, the x-component of the velocity fluctuation $u - U_0$ is plotted. The vortex is perfectly convected as shown in Tab. 2.1 and in Fig. 2.6.

Tab. 2.1. *Error analysis of reference test case* $\Delta x_L = \Delta x_R = \Delta x$ *and* $\Delta z_L = \Delta z_R = \Delta z$ *with* $\alpha = 0°$ *and* $h = 0.0$.

Error	ξ [−]	η [−]
$u - U_0$ [m/s]	$4.232 \cdot 10^{-3}$	0.282%
w [m/s]	$1.857 \cdot 10^{-2}$	1.238%

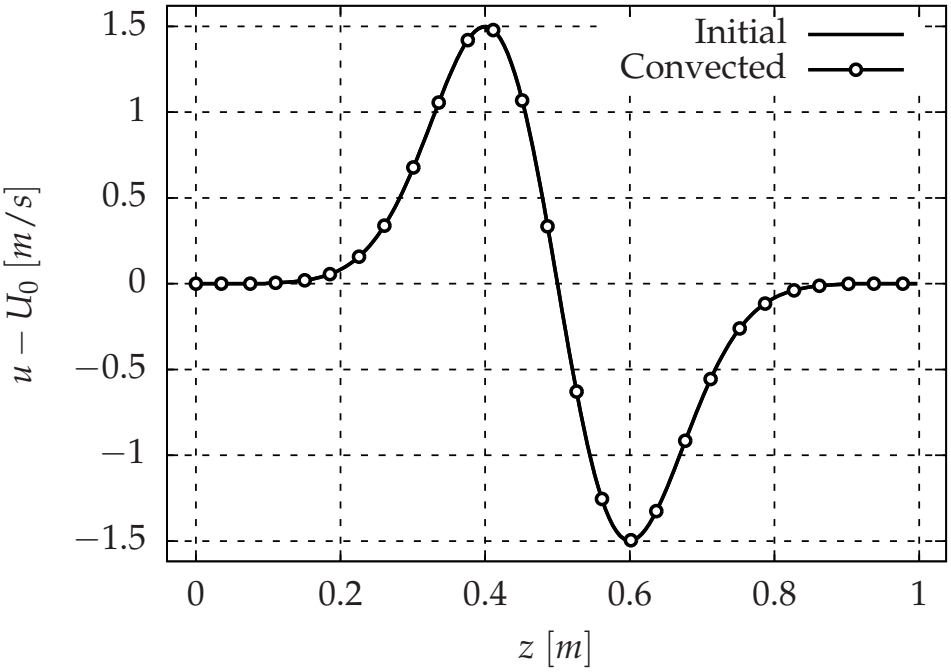
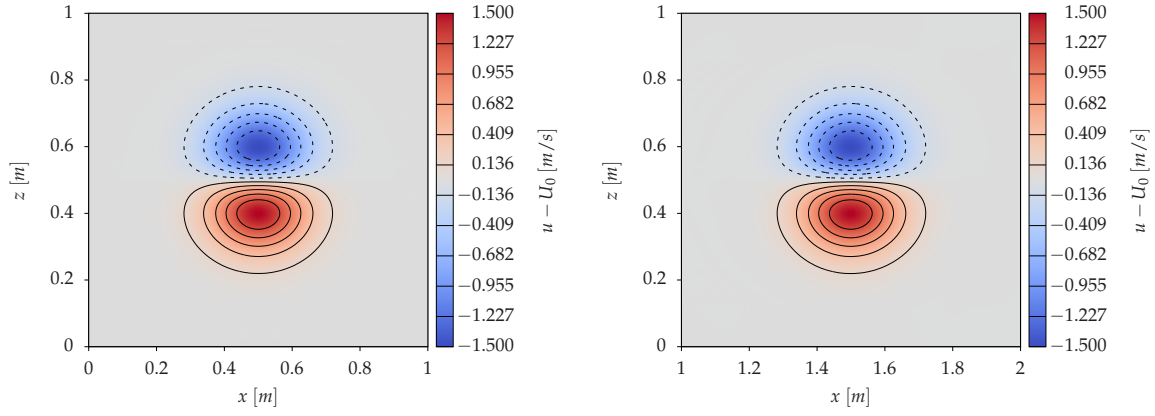


Fig. 2.5. *Reference test case on uniform grids, $\Delta x_L = \Delta x_R = \Delta x$ and $\Delta z_L = \Delta z_R = \Delta z$ with $\alpha = 0°$ and $h = 0.0$. x-component of the velocity fluctuation along the z-axis. Cut at $x = 0.5\ [m]$ and $x = 1.5\ [m]$.*



(a) *Initial vortex.* **(b)** *Convected vortex.*

Fig. 2.6. *Reference test case on uniform grids, $\Delta x_L = \Delta x_R = \Delta x$ and $\Delta z_L = \Delta z_R = \Delta z$ with $\alpha = 0°$ and $h = 0.0$. Isolines of x-component of the velocity fluctuation. Dash line for negative values. Solid line for positive values.*

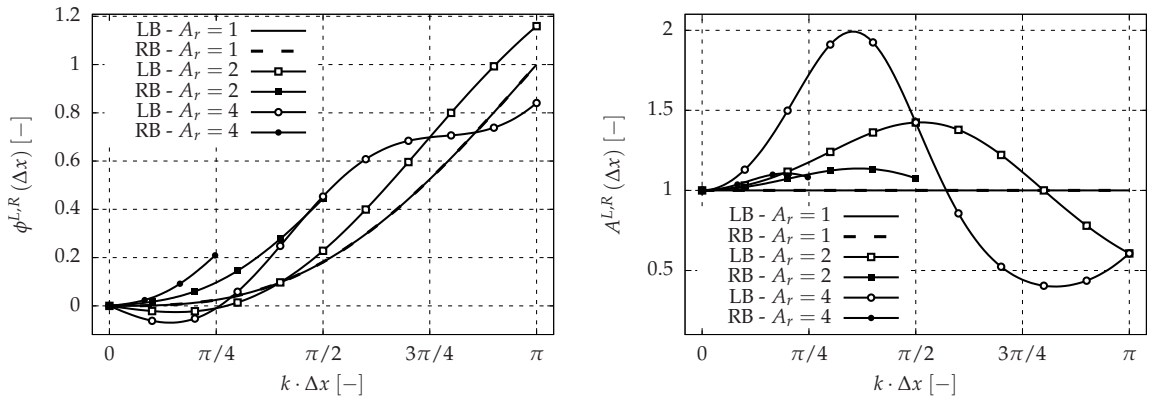
2.3.2 Coarsening along the x-axis

A coarsening ratio A_r is introduced to coarsen the right block. The left block spacings are $\Delta x_L = 1.0$ and $\Delta z_L = 1.0$. The right block spacings are $\Delta x_R = A_r$ and $\Delta z_R = 1.0$. The direction of propagation is orthogonal to the grid interface with $\alpha = 0°$. The analytical expressions of $\tilde{k}_x$ and $\tilde{k}_z$ can be deduced from (2.14) and (2.15) in Sec. 2.2. Remind that for the right block $k \cdot \Delta x_R \in [0, \pi / A_r]$ (see end of Sec. 2.2.5).

a. $h = 0.0$.

In this case, the grid interface is a conforming grid interface. The coarsening along the x-axis causes dispersion and amplification as shown in Fig. 2.7. The larger A_r is, the larger dispersion

and amplification become. The numerical computation would be unstable for all coarsening ratio $A_r > 1$.



(a) *Dispersion.* **(b)** *Dissipation.*

Fig. 2.7. *Coarsening along the x-axis,* $\Delta x_L = \Delta z_L = \Delta z_R = 1.0$ *and* $\Delta x_R = A_r$ *with* $\alpha = 0°$ *and* $h = 0.0$.

The mesh sizes used for the numerical computations are summarized in Tab. 2.2, and the corresponding coarsening ratio is pointed out. Cuts at $x = 1.5\ [m]$ show the x-component of the velocity fluctuation for different A_r in Fig. 2.8, and cuts at $z = 0.5\ [m]$ show the z-component of the velocity fluctuation for different A_r in Fig. 2.9. In this case, the conforming grid interface generates instabilities because of the coarsening along the x-axis.

Tab. 2.2. *Coarsening along the x-axis.*

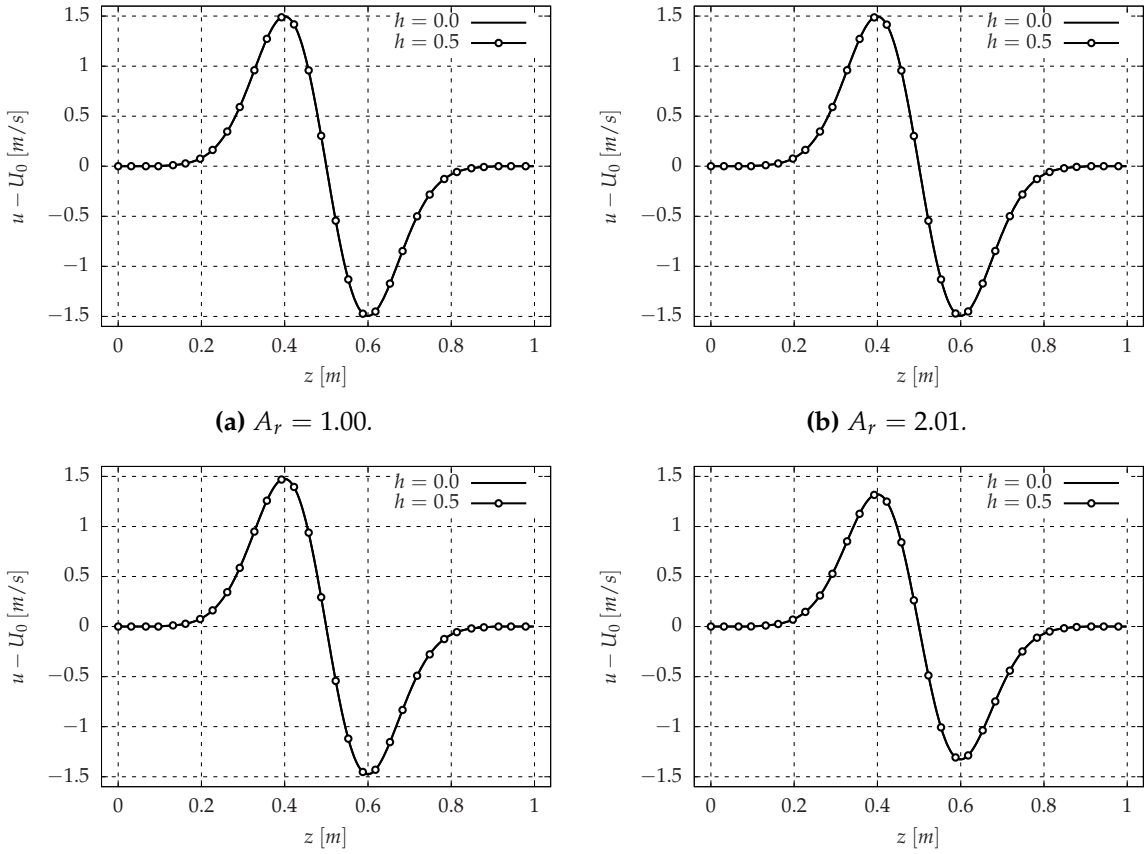
Block	N_x	N_z	A_r
Left block	200	400	1.00
Right block	200	400	
Left block	200	400	2.01
Right block	100	400	
Left block	200	400	4.06
Right block	50	400	
Left block	200	400	8.29
Right block	25	400	

b. $h = 0.5$.
In this case, the grid interface is a nonconforming grid interface. Indeed, for $\Delta z_L = 1.0$, $h \in [0, 0.5]$. So, the upper bound $h = 0.5$ is considered to gauge the influence of the nonconforming grid interface.

Regardless of coarsening ratio, computations with $h = 0.0$ or $h = 0.5$ exactly give the same results as shown in Fig. 2.8 and Fig. 2.9, and in Tab. 2.3. This implies that dissipation and dispersion are not due to the nonconforming interface, but they are caused by the coarsening along the x-axis.

Tab. 2.3. *Error analysis on a non-uniform grid $\Delta x_L = \Delta z_L = \Delta z_R = 1.0$ and $\Delta x_R = A_r$ with $\alpha = 0°$. h effect on the error of x-component and z-component of the velocity fluctuation.*

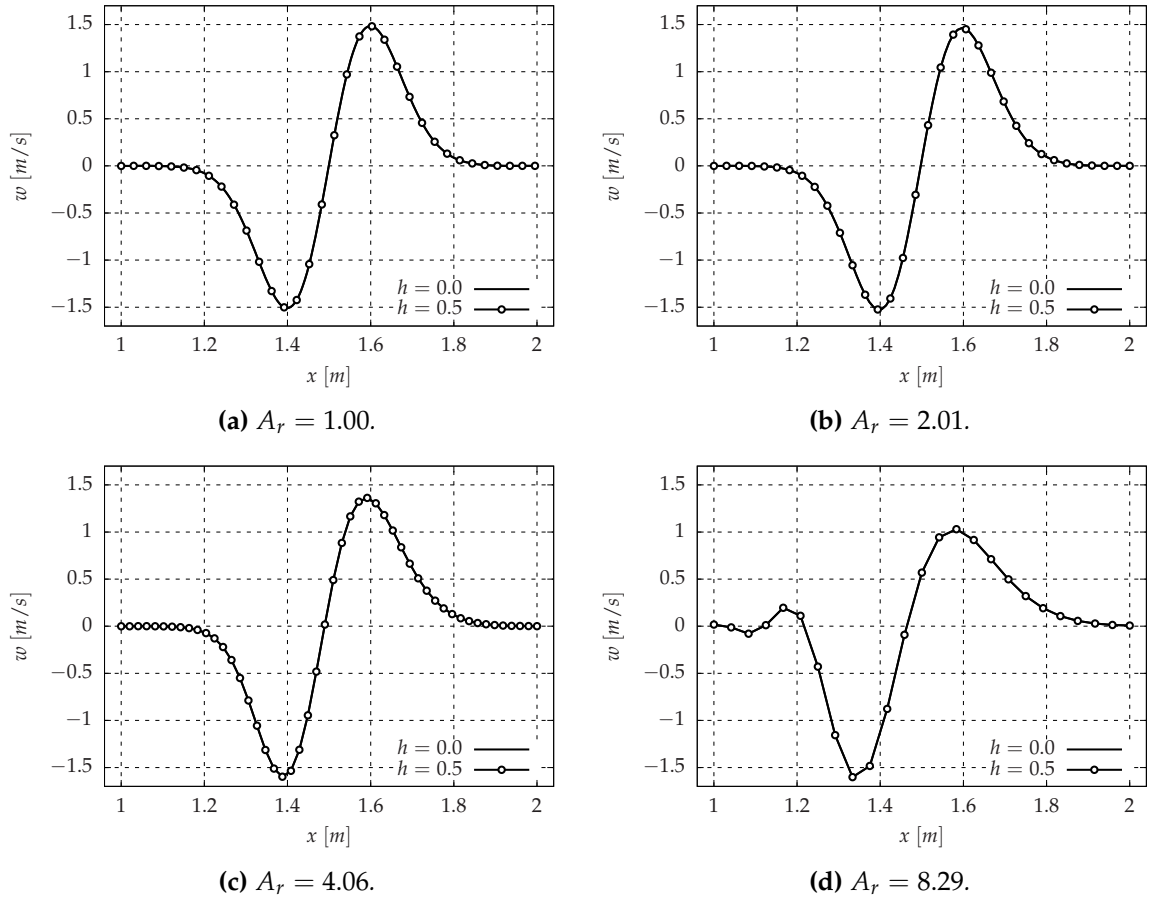
Error	A_r $[-]$	ξ $[-]$		η $[-]$	
		$h = 0.0$	$h = 0.5$	$h = 0.0$	$h = 0.5$
$u - U_0$ $[m/s]$	1.00	$4.232 \cdot 10^{-3}$	$4.221 \cdot 10^{-3}$	0.282%	0.282%
	2.01	$6.532 \cdot 10^{-3}$	$6.530 \cdot 10^{-3}$	0.435%	0.436%
	4.06	$2.621 \cdot 10^{-2}$	$2.621 \cdot 10^{-2}$	1.748%	1.748%
	8.29	$1.783 \cdot 10^{-1}$	$1.784 \cdot 10^{-1}$	11.888%	11.897%
w $[m/s]$	1.00	$1.857 \cdot 10^{-2}$	$1.861 \cdot 10^{-2}$	1.238%	1.241%
	2.01	$6.633 \cdot 10^{-2}$	$6.638 \cdot 10^{-2}$	4.429%	4.433%
	4.06	$2.686 \cdot 10^{-1}$	$2.687 \cdot 10^{-1}$	18.030%	18.035%
	8.29	$8.534 \cdot 10^{-1}$	$8.535 \cdot 10^{-1}$	58.601%	58.608%



(a) $A_r = 1.00$. **(b)** $A_r = 2.01$.

(c) $A_r = 4.06$. **(d)** $A_r = 8.29$.

Fig. 2.8. *Coarsening along the x-axis, $\Delta x_L = \Delta z_L = \Delta z_R = 1.0$ and $\Delta x_R = A_r$ with $\alpha = 0°$. h effect on the x-component of the velocity fluctuation. Cases $A_r = 8.29$, $h = 0$ or $h = 0.5$ lead to the largest dissipation.*



(a) $A_r = 1.00$. **(b)** $A_r = 2.01$.

(c) $A_r = 4.06$. **(d)** $A_r = 8.29$.

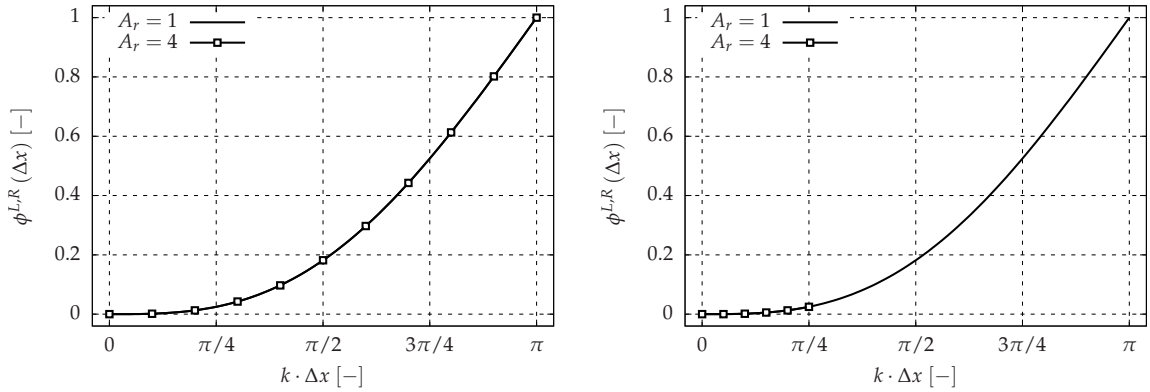
Fig. 2.9. *Coarsening along the x-axis,* $\Delta x_L = \Delta z_L = \Delta z_R = 1.0$ *and* $\Delta x_R = A_r$ *with* $\alpha = 0°$. *h effect on the z-component of the velocity fluctuation. Cases* $A_r = 8.29$, $h = 0$ *or* $h = 0.5$ *lead to the largest instability.*

2.3.3 Coarsening along the z-axis

After the study of the coarsening along the x-axis, the coarsening ratio A_r is now used to investigate the coarsening along the z-axis. The left block spacings are still $\Delta x_L = 1.0$ and $\Delta z_L = 1.0$, but the right block spacings are now $\Delta x_R = 1.0$ and $\Delta z_R = A_r$.

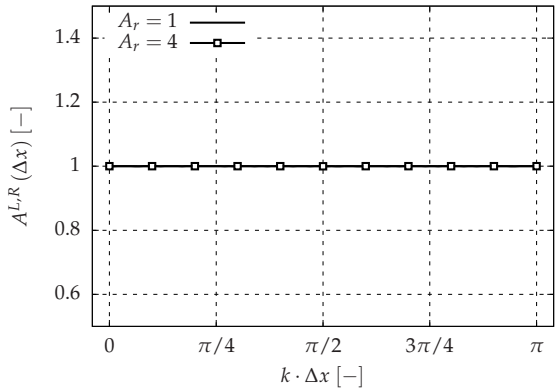
a. $\alpha = 0°$, $h = 0.0$.

If the direction of propagation is orthogonal to the coarsening, then the theoretical results show that the nonconforming interface does not cause dissipation nor dispersion as shown in Fig. 2.10.

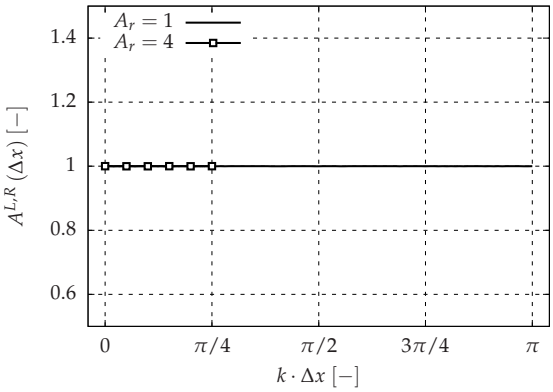


(a) *Dispersion for the left block.*

(b) *Dispersion for the right block.*



(c) *Dissipation for the left block.*



(d) *Dissipation for the right block.*

Fig. 2.10. *Coarsening along the z-axis,* $\Delta x_L = \Delta x_R = \Delta z_L = 1.0$ *and* $\Delta z_R = A_r$ *with* $\alpha = 0°$ *and* $h = 0$.

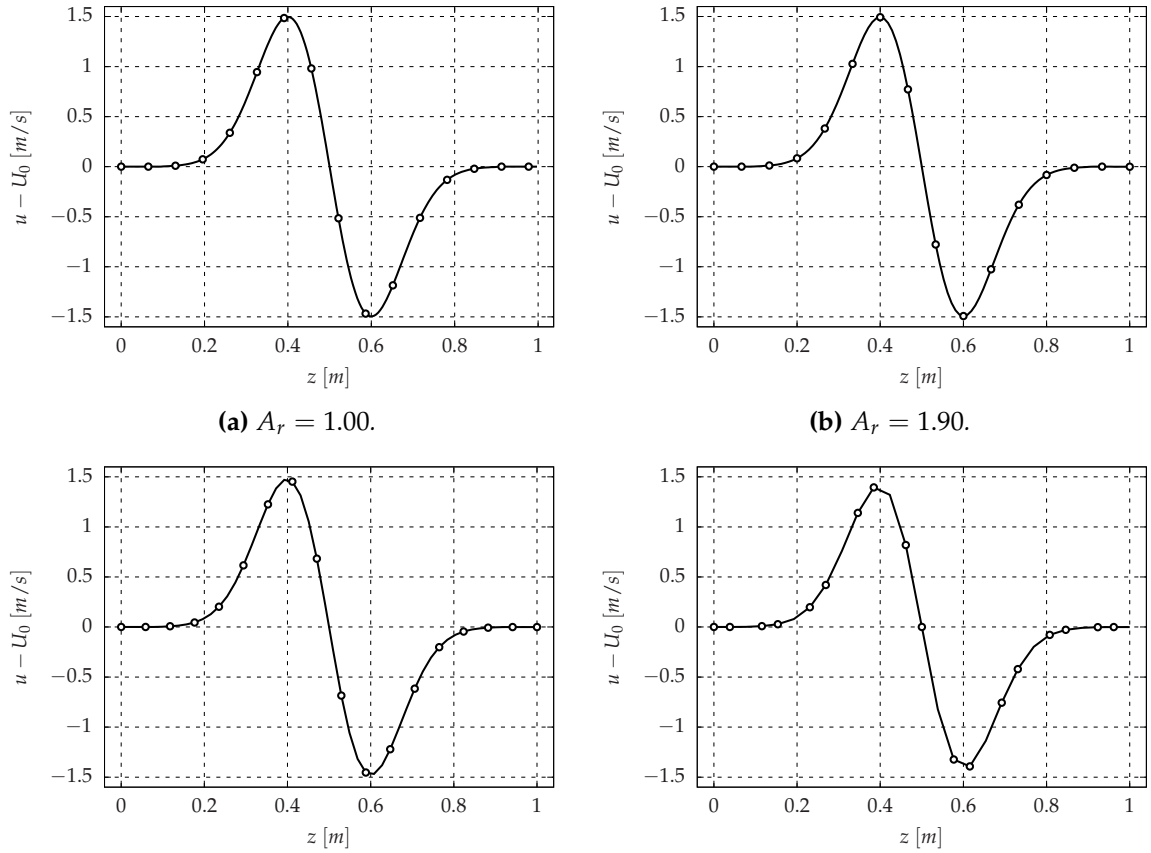
The mesh sizes used for the numerical computations are summarized in Tab. 2.4. The prime numbers ensure the existence of a nonconforming interface. As shown in Fig. 2.11 and Fig. 2.12, and in Tab. 2.5, the numerical computations also indicate that a moderate coarsening along the z-axis has no effect on dispersion and dissipation. For the largest coarsening ratio ($A_r = 7.67$), the right block is not refined enough to accurately capture the vortex after the nonconforming grid interface, and dispersion and dissipation occur.

Tab. 2.4. *Coarsening along the z-axis.*

Block	N_x	N_z	A_r
Left block	200	400	1.00
Right block	200	400	
Left block	200	400	1.90
Right block	200	211	
Left block	200	400	3.91
Right block	200	103	
Left block	200	400	7.67
Right block	200	53	

Tab. 2.5. *Error analysis on a non-uniform grid $\Delta x_L = \Delta x_R = \Delta z_L = 1.0$ and $\Delta z_R = A_r$ with $h = 0.0$. The direction of propagation is orthogonal to the interface ($\alpha = 0°$). Effect of coarsening along the z-axis on the error of x-component and z-component of the velocity fluctuation.*

Error	A_r $[-]$	ξ $[-]$	η $[-]$
$u - U_0$ $[m/s]$	1.00	$4.232 \cdot 10^{-3}$	0.282%
	1.90	$8.206 \cdot 10^{-3}$	0.547%
	3.91	$2.695 \cdot 10^{-2}$	1.808%
	7.67	$9.406 \cdot 10^{-2}$	6.414%
w $[m/s]$	1.00	$1.857 \cdot 10^{-2}$	1.238%
	1.90	$1.934 \cdot 10^{-2}$	1.290%
	3.91	$2.600 \cdot 10^{-2}$	1.734%
	7.67	$2.678 \cdot 10^{-2}$	1.786%



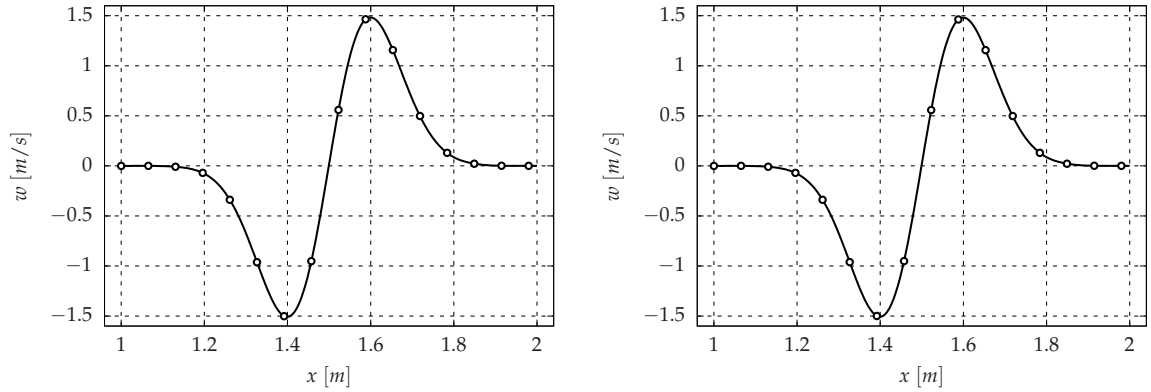
(a) $A_r = 1.00$.

(b) $A_r = 1.90$.

(c) $A_r = 3.91$.

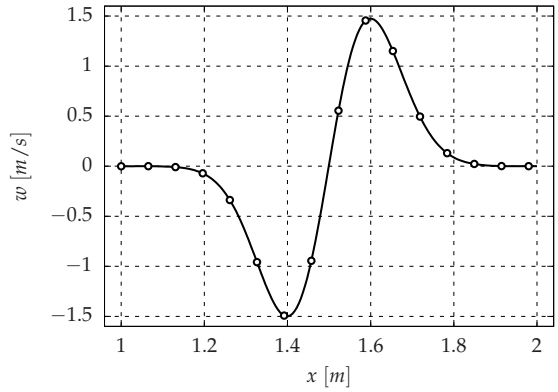
(d) $A_r = 7.67$.

Fig. 2.11. *Effect of coarsening along the z-axis on the x-component of the velocity fluctuation. $\Delta x_L = \Delta x_R = \Delta z_L = 1.0$ and $\Delta z_R = A_r$ with $\alpha = 0°$ and $h = 0$.*

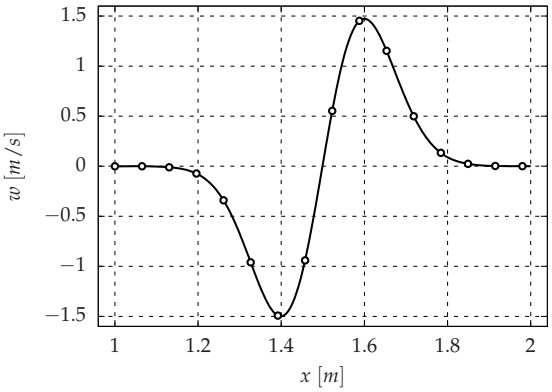


(a) $A_r = 1.00$.

(b) $A_r = 1.90$.



(c) $A_r = 3.91$.

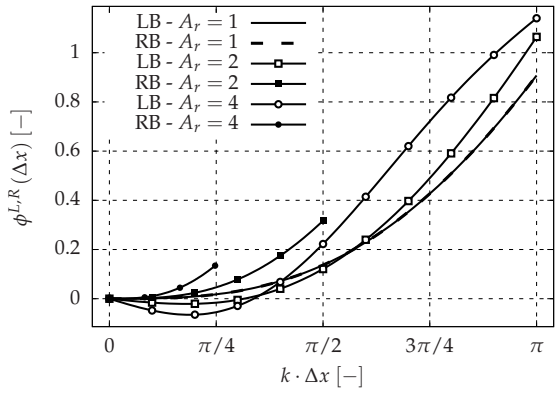


(d) $A_r = 7.67$.

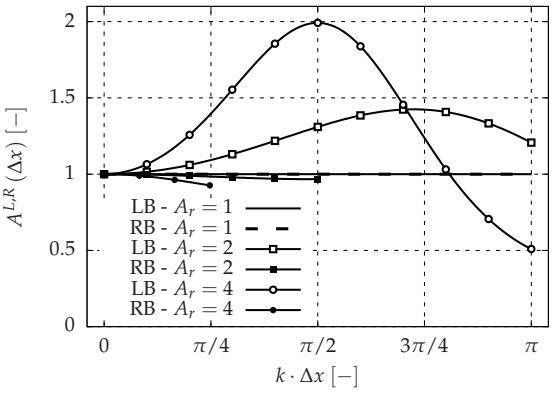
Fig. 2.12. *Effect of coarsening along the z-axis on the z-component of the velocity fluctuation.* $\Delta x_L = \Delta x_R = \Delta z_L = 1.0$ *and* $\Delta z_R = A_r$ *with* $\alpha = 0°$ *and* $h = 0$.

b. $\alpha = 45°$, $h = 0.0$.
Now, the direction of propagation is not orthogonal to the direction of coarsening anymore. A significant effect on dispersion and dissipation is shown in Fig. 2.13 compared to Fig. 2.10.

c. $\alpha = 45°$, $h = 0.5$.
In that case shown in Fig. 2.14, the nonconforming interface has a stabilizing effect, but it is more dispersive. Indeed, the amplification upper bound decreases from 2 to less than 1.5, and dispersion is always positive.

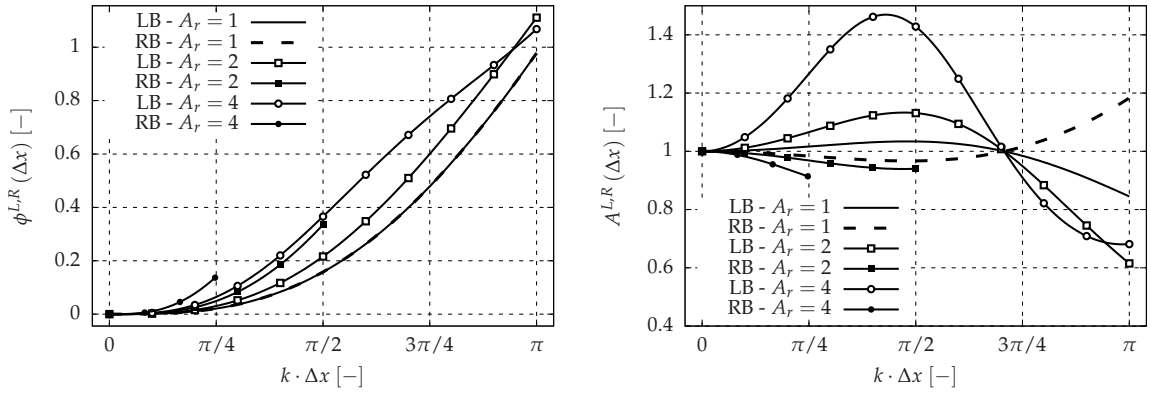


(a) *Dispersion.*



(b) *Dissipation.*

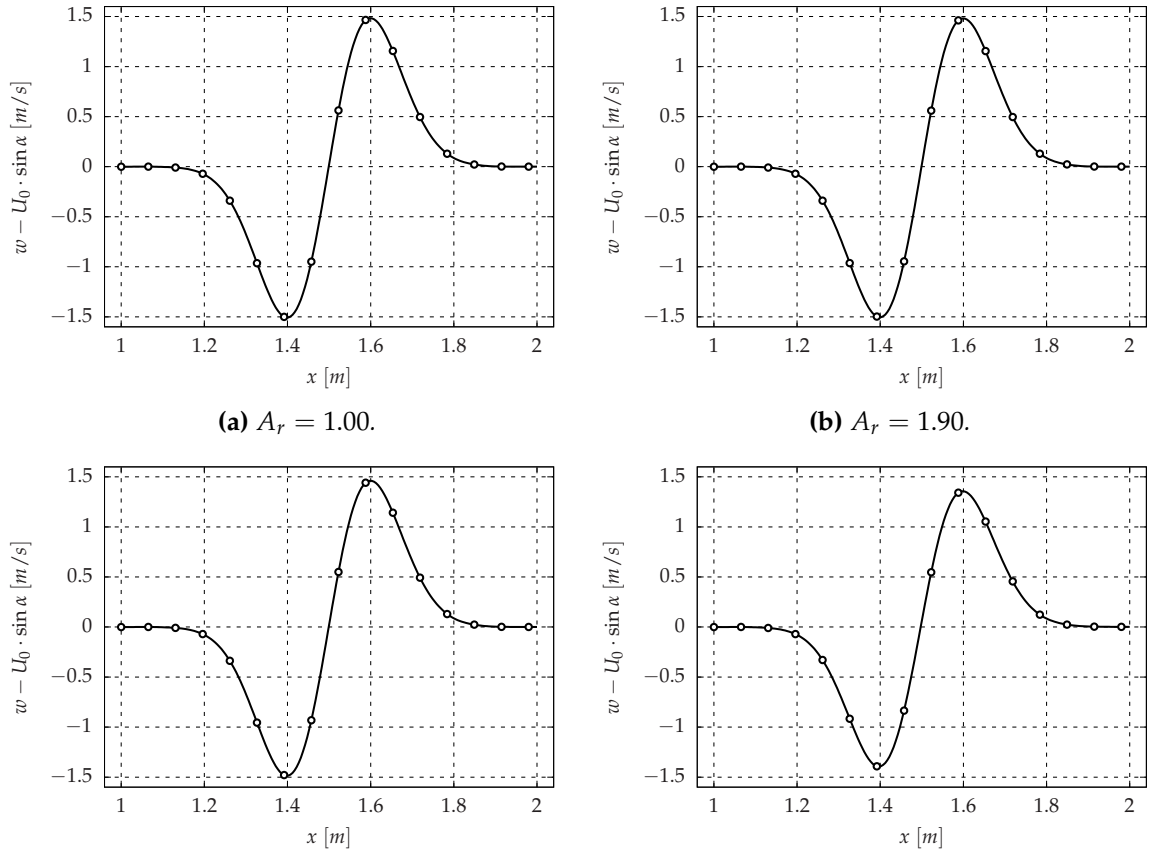
Fig. 2.13. *Coarsening along the z-axis,* $\Delta x_L = \Delta x_R = \Delta z_L = 1.0$ *and* $\Delta z_R = A_r$ *with* $\alpha = 45°$ *and* $h = 0.0$.



(a) *Dispersion.* **(b)** *Dissipation.*

Fig. 2.14. *Coarsening along the z-axis,* $\Delta x_L = \Delta x_R = \Delta z_L = 1.0$ *and* $\Delta z_R = A_r$ *with* $\alpha = 45°$ *and* $h = 0.5$.

Instead of being convected over one meter, the vortex is now convected over $\sqrt{2}$ meters. This implies to increase the number of iterations to maintain a constant CFL number. The mesh sizes are still summarized in Tab. 2.4. Fig. 2.15 and Fig. 2.16, and Tab. 2.6 show the effect of the coarsening along the z-axis on the z-component and the x-component of the velocity fluctuation.



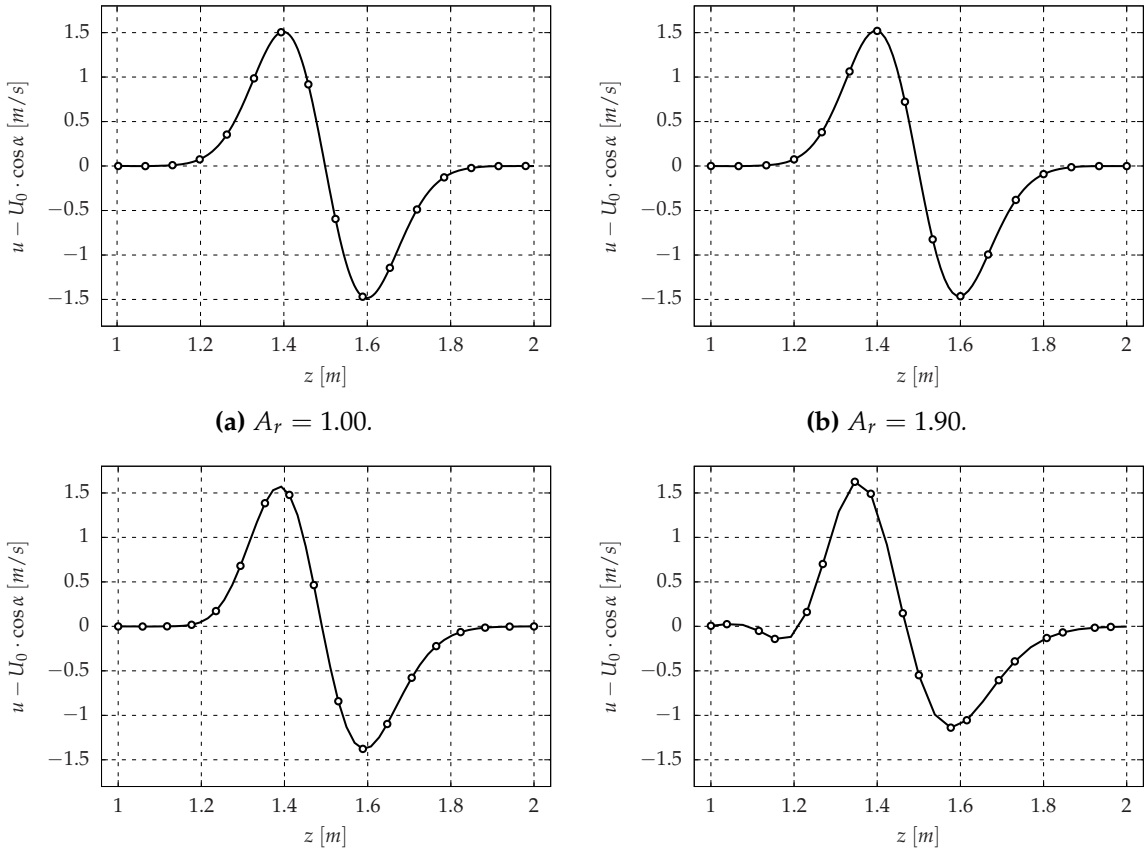
(a) $A_r = 1.00$. **(b)** $A_r = 1.90$.

(c) $A_r = 3.91$. **(d)** $A_r = 7.67$.

Fig. 2.15. *Effect of coarsening along the z-axis on the z-component of the velocity fluctuation.* $\Delta x_L = \Delta x_R = \Delta z_L = 1.0$ *and* $\Delta z_R = A_r$ *with* $h = 0.0$ *and* $\alpha = 45°$.

Tab. 2.6. *Error analysis on a non-uniform grid $\Delta x_L = \Delta x_R = \Delta z_L = 1.0$ and $\Delta z_R = A_r$ with $h = 0.0$. The direction of propagation is not orthogonal to the interface ($\alpha = 45°$). Effect of coarsening along the z-axis on the error of x-component and z-component of the velocity fluctuation.*

Error	A_r $[-]$	ξ $[-]$	η $[-]$
$u - U_0 \cdot \cos\alpha$ $[m/s]$	1.00	$2.899 \cdot 10^{-2}$	1.933%
	1.90	$7.010 \cdot 10^{-2}$	4.673%
	3.91	$2.453 \cdot 10^{-1}$	16.454%
	7.67	$7.352 \cdot 10^{-1}$	50.130%
$w - U_0 \cdot \sin\alpha$ $[m/s]$	1.00	$2.217 \cdot 10^{-2}$	1.479%
	1.90	$2.067 \cdot 10^{-2}$	1.379%
	3.91	$3.883 \cdot 10^{-2}$	2.590%
	7.67	$1.455 \cdot 10^{-1}$	9.705%



(a) $A_r = 1.00$.

(b) $A_r = 1.90$.

(c) $A_r = 3.91$.

(d) $A_r = 7.67$.

Fig. 2.16. *Effect of coarsening along the z-axis on the x-component of the velocity fluctuation. $\Delta x_L = \Delta x_R = \Delta z_L = 1.0$ and $\Delta z_R = A_r$ with $h = 0.0$ and $\alpha = 45°$.*

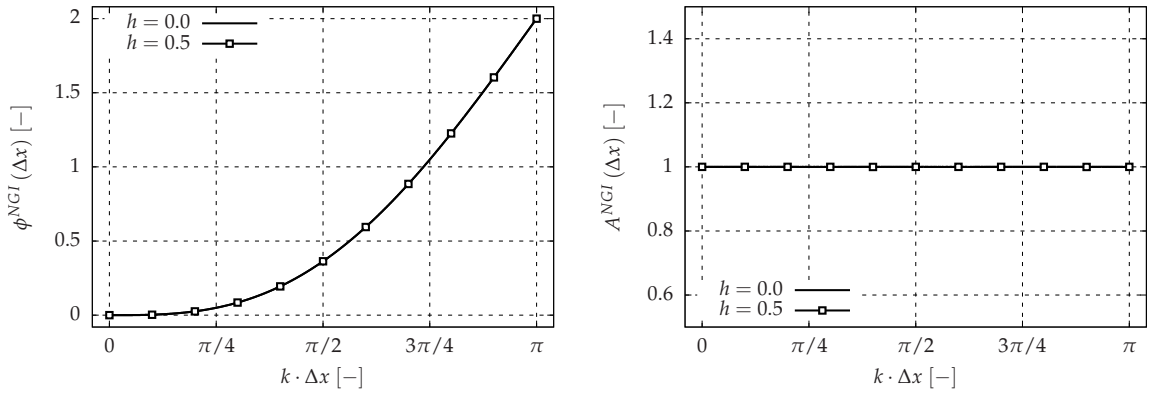
2.3.4 Specific case of uniform grids

The effect of the grid interface is studied on uniform grids, with $\Delta x_L = \Delta x_R$ and $\Delta z_L = \Delta z_R$. Without loss of generality, these values are set to 1 so that $h \in [0, 0.5]$. The overall effect of the grid interface is considered by plotting A^{NGI} and ϕ^{NGI}. These variables are compared with the ***elsA*** numerical computations. Computations with different directions of propagation are

analyzed.

a. $\alpha = 0°$.

First, the direction of propagation is orthogonal to the interface. If h is equal to 0, then the grid interface is conforming, whereas the grid interface is nonconforming if h is equal to 0.5. Looking at (2.19) and (2.20), α must be different from 0 so that h could have an effect. Fig. 2.17 confirms that h has no effect on dispersion and dissipation for $\alpha = 0°$.



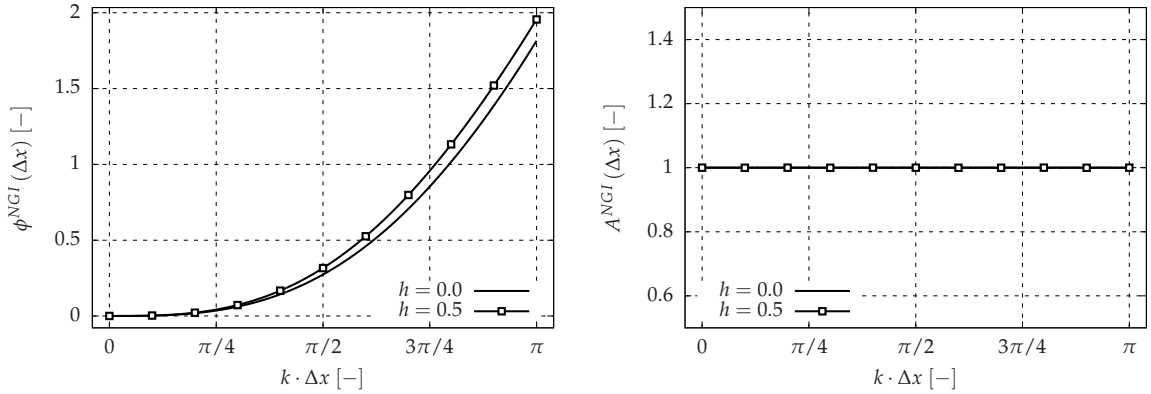
(a) ıDispersion. **(b)** *Dissipation.*

Fig. 2.17. *h effect with* $\alpha = 0°$ *on uniform grids* $\Delta x_L = \Delta x_R = \Delta z_L = \Delta z_R$.

Numerical computations were made with $h = 0.0$ and $h = 0.5$. As shown by Fig. 2.8 and Fig. 2.9 with $A_r = 1$, the vortex is perfectly convected, and, as expected, h has no effect on dispersion and dissipation. The error analysis is available in Tab. 2.3.

b. $\alpha = 45°$.

In this case, Fig. 2.18 shows that the grid interface is more dispersive for $h = 0.5$ than for $h = 0.0$. Our interpretation is that, if a high-frequency signal reaches its maximum on a node of the left block, this maximum may not be well-discretized on the right block because nodes of the right block are shifted from those of the left one ($h = 0.5$). Consequently, the grid interface becomes more dispersive. It is still stable, as shown in Fig. 2.18, since the amplification is equal to one.



(a) *Dispersion.* **(b)** *Dissipation.*

Fig. 2.18. *h effect with* $\alpha = 45°$ *on uniform grids* $\Delta x_L = \Delta x_R = \Delta z_L = \Delta z_R$.

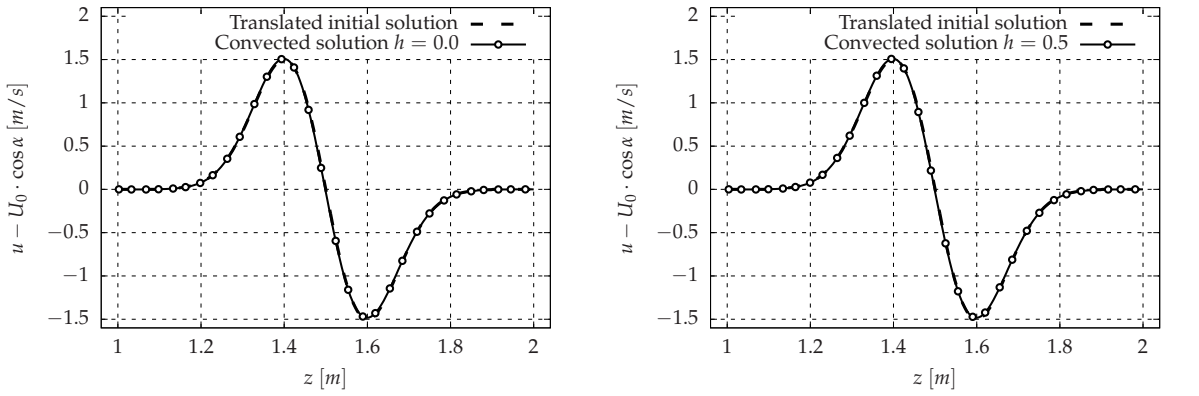
Numerical computations were performed with $h = 0.0$ and $h = 0.5$. Instead of being convected over one meter, the vortex is now convected over $\sqrt{2}$ meters. As expected, there is

no amplification in Fig. 2.19. The dispersive effect does not appear because the frequency of the signal is not high enough.

To conclude this section, if the grid is uniform, the grid interface has a good behavior regardless of the direction of propagation as shown in Tab. 2.3 and Tab. 2.7.

Tab. 2.7. *Error analysis on a uniform grid* $\Delta x_L = \Delta x_R = \Delta z_L = \Delta z_R$. α *is equal to* $45°$.

Error	h $[-]$	ξ $[-]$	η $[-]$
$u - U_0 \cdot \cos\alpha$ $[m/s]$	0.0	$2.899 \cdot 10^{-2}$	1.933%
	0.5	$2.899 \cdot 10^{-2}$	1.933%
$w - U_0 \cdot \sin\alpha$ $[m/s]$	0.0	$2.217 \cdot 10^{-2}$	1.479%
	0.5	$2.215 \cdot 10^{-2}$	1.477%



(a) *Comparison between the initial solution and the convected solution with* $h = 0.0$.

(b) *Comparison between the initial solution and the convected solution with* $h = 0.5$.

Fig. 2.19. *x-component of the velocity fluctuation along the z-axis on uniform grids* $\Delta x_L = \Delta x_R = \Delta z_L = \Delta z_R$ *with* $\alpha = 45°$. *Cut at* $x = 0.5$ $[m]$ *and* $x = 1.5$ $[m]$.

2.3.5 Reflection of high-frequency waves

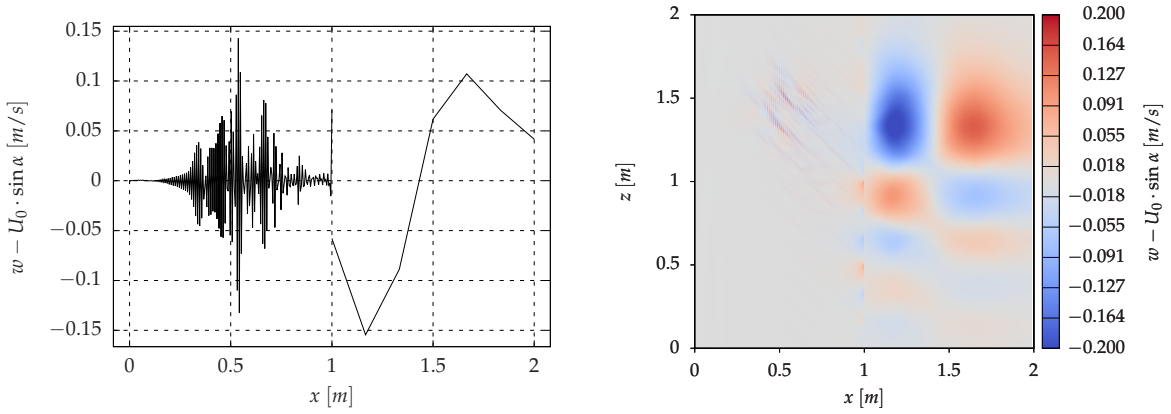
As mentioned in the introduction, the coarsening ratio A_r can be really high for steady RANS computations. It is interesting to see what happens for unsteady flows. The left block sizes are 200×400, and the right block sizes are 7×13. The coarsening ratio A_r is equal to 33.17 along the x-axis, and 33.25 along the z-axis. A numerical computation is performed with parameters $h = 0$ and $\alpha = 45°$. As can be seen in Fig. 2.20, some high-frequency waves are reflected by the interface. When such a grid interface is used, the vortex is expected to dissipate depending on the mesh refinement, but without wave reflection to protect the upstream flow. At this point, this behavior can be due to the metric discontinuity or to the nonconforming grid interface.

The analysis presented in the next section shows that the reflected waves are a consequence of the metric discontinuity. This point can also be explained using a solution analysis based on wave decomposition. Wave analysis has been performed in one dimension by Vichnevetsky and Bowles [5] and by Vichnevetsky [59, 60] for boundary conditions and mesh refinement. They have decomposed any solution as a sum of p and q waves, with p waves traveling in the hyperbolic direction (positive group velocity) and q waves in the opposite direction (negative group velocity). They explain that q waves are not necessary present in the flow. But, if waves with a negative group velocity appear in the flow, they explain that their presence

is a consequence of boundary conditions not adapted. If a nonconforming grid interface is considered as a numerical boundary condition between two blocks, it can be argued that the data reconstruction on the right-hand side of the nonconforming grid interface is not in agreement with waves leaving the left block, hence the production of spurious waves.

There is also another way to analyze the result. The metric discontinuity modifies the dispersion relation as shown in the previous sections, and this is responsible for waves reflection. Analogously to electromagnetic waves, a reflected wave is created to satisfy the boundary conditions at the interface separating two media of different refractive indices.

In the next section, a method is introduced to avoid these spurious modes with a negative group velocity.



(a) *Cut at* $z = 1.5\ [m]$.

(b) *Visualization in the xz plane.*

Fig. 2.20. *Reflection of high-frequency waves. z-component of the velocity fluctuation. The coarsening ratio* A_r *is equal to* 33.17 *along the x-axis and* 33.25 *along the z-axis. h is zero, and* α *is equal to* 45°.

2.4 High-order metric-dependent interpolation for Riemann solver

This section deals with filtering to avoid reflection of high-frequency waves. We therefore attach great importance to study and implement a Riemann solver only at the interface. Tests are performed with the test case described in Sec. 2.3.5. A new numerical scheme is developed to take into account the local metric as in Fig. 2.21. Indeed, the numerical schemes are often built assuming that the grid is uniform, but this can no longer be the case if a nonconforming grid interface is used. Only the flux on the nonconforming grid interface has to be modified. This new numerical scheme is based on the Roe's scheme [57, 58] used for industrial computations. The Roe's numerical flux is given by:

$$F_{m+1/2} = \frac{1}{2}\left[\mathcal{F}\left(W^L_{m+1/2}\right) + \mathcal{F}\left(W^R_{m+1/2}\right)\right] - \frac{1}{2}\left|\overline{\overline{A}}\left(W^L_{m+1/2}, W^R_{m+1/2}\right)\right|\left(W^R_{m+1/2} - W^L_{m+1/2}\right),$$

where $W^L_{m+1/2}$ and $W^R_{m+1/2}$ are the conservative variables interpolated on the left and right sides of the interface $m + 1/2$. $\overline{\overline{A}}$ is the Roe's matrix which satisfies the following list of properties.

- $\overline{\overline{A}}\left(W, W\right) = A\left(W\right)$ where A is the Jacobian matrix
- $\mathcal{F}\left(W^R\right) - \mathcal{F}\left(W^L\right) = \overline{\overline{A}}\left(W^L, W^R\right)\left(W^R - W^L\right)$
- The eigenvectors of $\overline{\overline{A}}\left(W^L, W^R\right)$ are linearly independent

The Roe's solver leads to a first-order solution when it is directly applied to mean quantities issued from the left and right sides of the interface. In order to increase flux accuracy, it is mandatory to find a high-order extrapolation of the solution on the interface, as for example

the following third-order accurate extrapolation,

$$W^L_{m+1/2} = \frac{5}{6}W_m + \frac{1}{3}W_{m+1} - \frac{1}{6}W_{m-1}, \tag{2.31}$$

$$W^R_{m+1/2} = \frac{5}{6}W_{m+1} + \frac{1}{3}W_m - \frac{1}{6}W_{m+2}. \tag{2.32}$$

In the following, these extrapolations are adapted to account for the local metric when the interface $i + 1/2$ corresponds to a grid interface between two blocks. Beginning with $W^L_{i+1/2}$, the expression of the variable is sought in the form $W^L_{i+1/2} = \alpha W_i + \beta W_{i+1} + \gamma W_{i-1}$, with $(\alpha, \beta, \gamma) \in \mathbb{R}^3$. Let us consider the advection equation with a constant and positive transport velocity $c = 1\ [m/s]$. The Roe's numerical flux is then simply the left state. Following (2.5), the new numerical scheme applied to compute the gradient of the variables with respect to x is based on an interpolation that is metric-dependent,

$$\Delta x \left.\frac{\partial W}{\partial x}\right|_i = W^L_{i+1/2} - W^L_{i-1/2} = \left(\alpha - \frac{1}{3}\right) W_i + \beta W_{i+1} + \left(\gamma - \frac{5}{6}\right) W_{i-1} + \frac{1}{6}W_{i-2}. \tag{2.33}$$

For the sake of clarity, all details regarding Taylor series expansions are given. Using notations introduced in Fig. 2.21,

$$W_{i-2} = W_i - 2\Delta x \left.\frac{\partial W}{\partial x}\right|_i + 2\Delta x^2 \left.\frac{\partial^2 W}{\partial x^2}\right|_i + \mathcal{O}\left(\Delta x^3\right), \tag{2.34}$$

$$W_{i-1} = W_i - \Delta x \left.\frac{\partial W}{\partial x}\right|_i + \frac{\Delta x^2}{2} \left.\frac{\partial^2 W}{\partial x^2}\right|_i + \mathcal{O}\left(\Delta x^3\right), \tag{2.35}$$

$$W_{i+1} = W_i + \Delta x \frac{A_r + 1}{2} \left.\frac{\partial W}{\partial x}\right|_i + \frac{\Delta x^2}{2} \left(\frac{A_r + 1}{2}\right)^2 \left.\frac{\partial^2 W}{\partial x^2}\right|_i + \mathcal{O}\left(\Delta x^3\right). \tag{2.36}$$

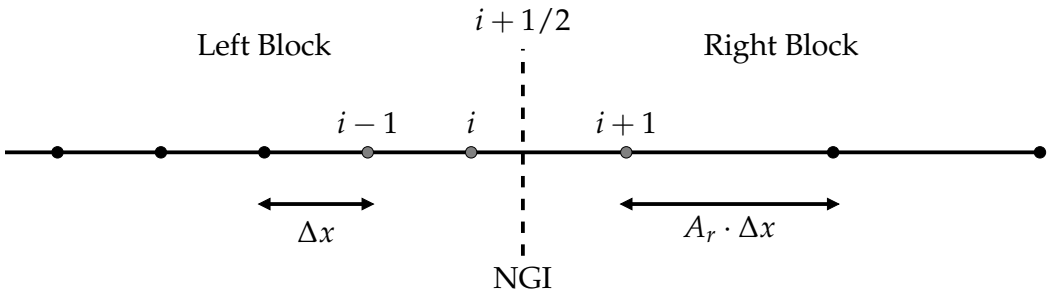


Fig. 2.21. *Stencil for the second-order accurate interpolation of the left state. The grid is non-uniform. The left and right blocks are separated by an interface.*

Substituting (2.34), (2.35), and (2.36) in (2.33), one obtains

$$\begin{aligned}
\Delta x \left.\frac{\partial W}{\partial x}\right|_i = {} & W_i \left(\alpha - \frac{1}{3} + \beta + \gamma - \frac{5}{6} + \frac{1}{6}\right) \\
& + \left.\frac{\partial W}{\partial x}\right|_i \Delta x \left(\frac{A_r + 1}{2}\beta + \left(-\gamma + \frac{5}{6}\right) - \frac{1}{3}\right) \\
& + \left.\frac{\partial^2 W}{\partial x^2}\right|_i \Delta x^2 \left(\frac{1}{2}\left(\frac{A_r + 1}{2}\right)^2 \beta + \frac{1}{2}\left(\gamma - \frac{5}{6}\right) + \frac{1}{3}\right) + \mathcal{O}\left(\Delta x^3\right).
\end{aligned}$$

Note that the zero-order condition is no longer trivially satisfied as it would have been if the metric was continuous across the interface. Only a second-order accurate interpolation is now possible. The coefficients α, β and γ are the solution of the following system of equations:

$$\alpha - \frac{1}{3} + \beta + \gamma - \frac{5}{6} + \frac{1}{6} = 0 \quad \text{is the zeroth-order condition,} \tag{2.37}$$

$$\frac{\beta}{2}(A_r + 1) - \gamma + \frac{5}{6} - \frac{1}{3} = 1 \quad \text{is the first-order condition,} \tag{2.38}$$

$$\frac{\beta}{2}\left(\frac{A_r + 1}{2}\right)^2 + \frac{\gamma - \frac{5}{6}}{2} + \frac{1}{3} = 0 \quad \text{is the second-order condition.} \tag{2.39}$$

Collecting all the terms, one obtains:

$$\begin{pmatrix} 1 & 1 & 1 \\ 0 & \frac{A_r + 1}{2} & -1 \\ 0 & \frac{1}{2}\left(\frac{A_r + 1}{2}\right)^2 & \frac{1}{2} \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \\ \gamma \end{pmatrix} = \begin{pmatrix} 1 \\ \frac{1}{2} \\ \frac{1}{12} \end{pmatrix}.$$

Cramer's rule gives: $\alpha = \dfrac{9A_r + 1}{6(A_r + 1)}$, $\beta = \dfrac{8}{3(A_r + 1)(A_r + 3)}$, and $\gamma = -\dfrac{3A_r + 1}{6(A_r + 3)}$. The new second-order accurate interpolation for the left state is given by

$$W^L_{i+1/2} = \frac{9A_r + 1}{6(A_r + 1)} \cdot W_i + \frac{8}{3(A_r + 1)(A_r + 3)} \cdot W_{i+1} - \frac{3A_r + 1}{6(A_r + 3)} \cdot W_{i-1}.$$

The asymptotic behavior is given by: $\lim\limits_{A_r \to +\infty} \alpha = \dfrac{3}{2}$, $\lim\limits_{A_r \to +\infty} \beta = 0$, and $\lim\limits_{A_r \to +\infty} \gamma = -\dfrac{1}{2}$. The larger A_r is, the less influence the state in cell $i+1$ located in the right block has. Moreover, these new interpolation weights are consistent since they are equal to the third-order interpolation of (2.31) when A_r tends to 1: $\lim\limits_{A_r \to 1} \alpha = \dfrac{5}{6}$, $\lim\limits_{A_r \to 1} \beta = \dfrac{1}{3}$, and $\lim\limits_{A_r \to 1} \gamma = -\dfrac{1}{6}$.

Since the cell centers are not symmetric along the interface, the interpolation for the left and right states is not symmetric, and new coefficients must be established for the right state. To do so, let us consider an advection equation with a constant and negative advection velocity $c = -1$. The Roe's numerical flux is then simply the right state. Following the same approach as for $W^L_{i+1/2}$, the new second-order accurate interpolation for the right state is given by:

$$W^R_{i+1/2} = \frac{7A_r^2 - 3A_r + 6}{6A_r(A_r + 1)} \cdot W_{i+1} + \frac{4A_r(3 - A_r)}{3(1 + A_r)(3A_r + 1)} \cdot W_i + \frac{5A_r^2 - 3A_r - 6}{6A_r(3A_r + 1)} \cdot W_{i+2}.$$

This right-state interpolation is also consistent since $A_r = 1$ leads to the third-order interpolation of (2.32).

Fig. 2.22 shows the effect of the newly high-order metric-dependent interpolation on high-frequency waves at the grid interface. The interpolation using the local metric enables to remove most of the spurious high-frequency waves previously generated with the standard interpolation. However, some residual oscillations still remain. Indeed, the interpolation should take into account the discontinuities of spatial discretization in the three dimensions. In our demonstration, the interpolation is only corrected in the orthogonal direction to the grid interface.

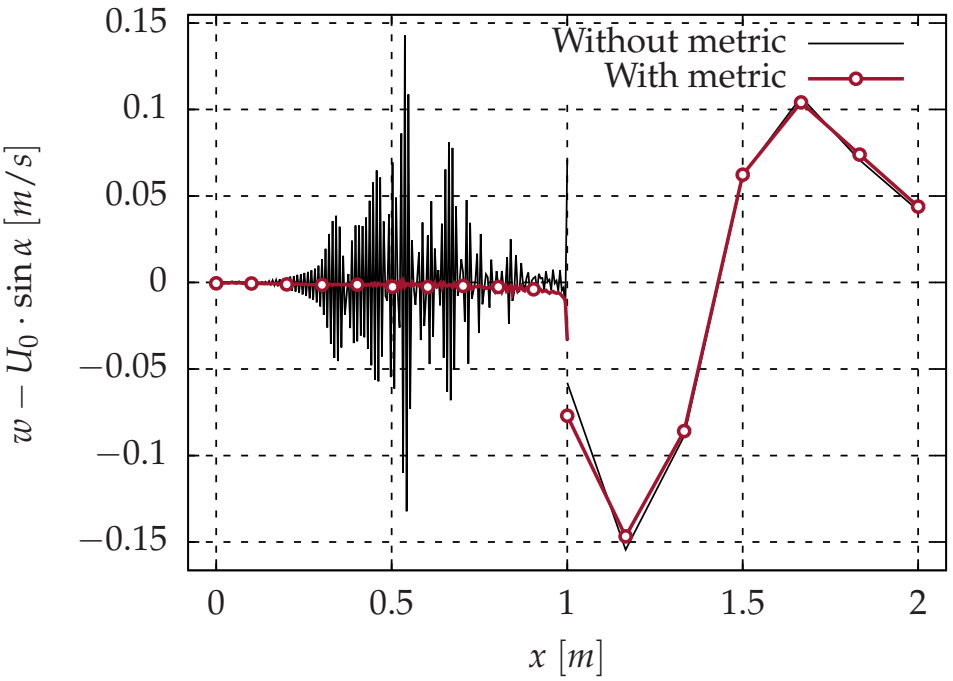


Fig. 2.22. *z-component of the velocity fluctuation. Cut at* $z = 1.5\ [m]$*. The coarsening ratio* A_r *is equal to* 33.17 *along the x-axis and* 33.25 *along the z-axis. h is zero, and* α *is equal to* 45°*. The Roe's scheme which depends on metric is only used at the interface. The centered scheme is used elsewhere. Most of high-frequency waves are removed.*

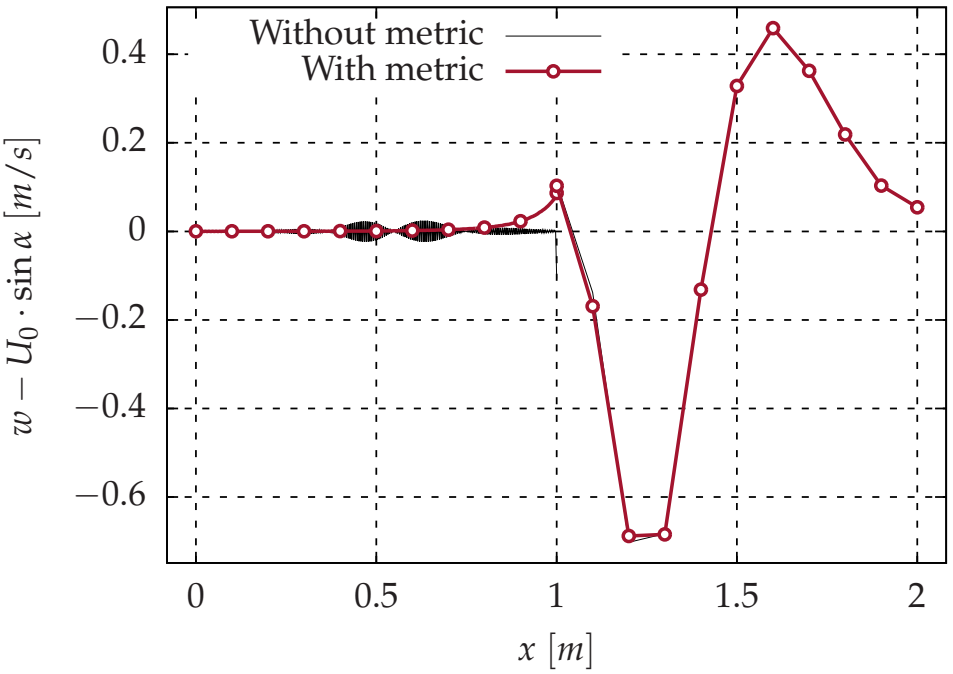


Fig. 2.23. *Effect of interpolation of left and right states dependent on metric which is required for the Riemann solver at the interface. Coarsening along the x-axis* $\Delta x_R = 40 \cdot \Delta x_L$*,* $\Delta z_R = \Delta z_L$ *with* $\alpha = 0°$ *and* $h = 0.0$*.*

A test has been carried out only with a coarsening along the x-axis to verify that these residual oscillations are caused by the interpolation which does not take into account the

transverse directions to the grid interface, but only the orthogonal one. In Fig. 2.23, $\Delta x_R = 40 \cdot \Delta x_L$, $\Delta z_R = \Delta z_L$, and the direction of propagation is orthogonal to the interface ($\alpha = 0°$). Consequently, the grid interface is a conforming interface, and there is only a discontinuity of spatial discretization along the x-axis. As shown in Fig. 2.23, if the interpolation does not take into account the local metric at the interface, high-frequency waves are reflected. Nonetheless, if the interpolation is metric-dependent, high-frequency waves are totally removed. The residual oscillations observed in Fig. 2.22 have totally disappeared, and hence it is absolutely necessary to take into account the local metric for the interpolation of left and right states at the interface.

2.5 Conclusion

Nonconforming grid interfaces are nowadays introduced in most of industrial configurations, either for external aerodynamics or for rotor/stator interactions in turbomachinery. The present chapter addressed the question of the numerical stability of the numerical scheme used at a nonconforming grid interface for an unsteady flow. To do so, two blocks with different discretizations were glued with a grid interface. Theoretical results from Fourier or Von Neumann analysis have been complemented by numerical results for the convection of an isentropic compressible vortex.

Some conclusions can be drawn with uniform block discretizations with the same cell dimensions in both blocks. If the direction of propagation is orthogonal to the interface, both theoretical results and numerical computations show that the grid interface has no effect on dispersion and dissipation. If the direction of propagation is not orthogonal to the interface, it is deduced from theoretical results, and confirmed by numerical results, that the grid interface induces dispersion without dissipation.

Nonconforming grid interfaces are mainly used in computations in order to locally increase the cell size across the interface. They avoid propagating a mesh refinement in low-interest regions of a flow. In such a situation, the coarsening ratio can vary between 2 and more than 40 for industrial steady applications. For unsteady flows, the effect of the grid interface on dispersion and amplification is due to the metric discontinuity rather than to the nonconforming aspect (parameter h). If the metric discontinuity occurs in the direction orthogonal to the direction of propagation, neither amplification nor dispersion are observed. If the metric discontinuity occurs in the direction parallel to the direction of propagation, both amplification and dispersion are observed. In addition, this can lead to reflections of undesirable high-frequency waves.

Avoiding spurious reflections is mandatory to get confidence in simulations. Based on inspiration from studies on NSCBC conditions, an approximate Riemann solver has been selected to develop a new numerical treatment for the nonconforming grid interface. The key point is to introduce a high-order extrapolation of quantities that account for the local metric on both sides of the interface. It has been shown that spurious waves are removed.

3 Vichnevetsky's framework

"The real voyage of discovery consists not in seeking new lands but seeing with new eyes."

— Marcel Proust (1871 - 1922)

Abstract. *In this chapter, Vichnevetsky's framework is detailed to explain in particular the results obtained in Sec. 2.3.5. First of all, the problem is illustrated by a simple numerical experiment which points out the reflection of high-frequency waves called spurious. Then, it is shown that standard tools to analyze schemes stability do not succeed in explaining the observed behavior, hence the necessity of Vichnevetsky's framework. It is shown that the reflection coefficient can be predicted and numerically recovered with a very good accuracy.*

3.1 Introductory numerical experiment

To illustrate Vichnevetsky's framework, let us start by a very simple numerical experiment. In the whole chapter, only the one-dimensional advection equation (3.1) is considered where u is the unknown and c the constant advective velocity.

$$\frac{\partial u}{\partial t} + c\,\frac{\partial u}{\partial x} = 0. \tag{3.1}$$

The initial condition is given by $u\,(x, t = 0) = u_0\,(x)$. This equation is hyperbolic with only one characteristic to define information propagation. Here, the advection velocity c is considered positive in such a way that information propagates from left to right. The dispersion relation is classically given by (3.2) where $\omega\ [s^{-1}]$ is the angular frequency and $k\ [m^{-1}]$ the wavenumber.

$$\omega = ck. \tag{3.2}$$

This equation is solved on a one-dimensional domain with $x \in [0, 1]$ with a cell size $h = 0.005\ [m]$. A Dirichlet boundary condition is applied on the left boundary. A characteristic boundary condition is applied on the right boundary. This is simply achieved with a first-order upwind scheme. Under these conditions, the analytical solution is simply given by:

$$u\,(x, t) = u_0\,(x - ct)\,. \tag{3.3}$$

That means that the problem is physically well-posed, which is a prerequisite to solve it numerically. The space discretization is achieved by a second-order centered Finite Difference scheme defined by:

$$\left.\frac{\partial u}{\partial x}\right|_i = \frac{u_{i+1} - u_{i-1}}{2h}, \tag{3.4}$$

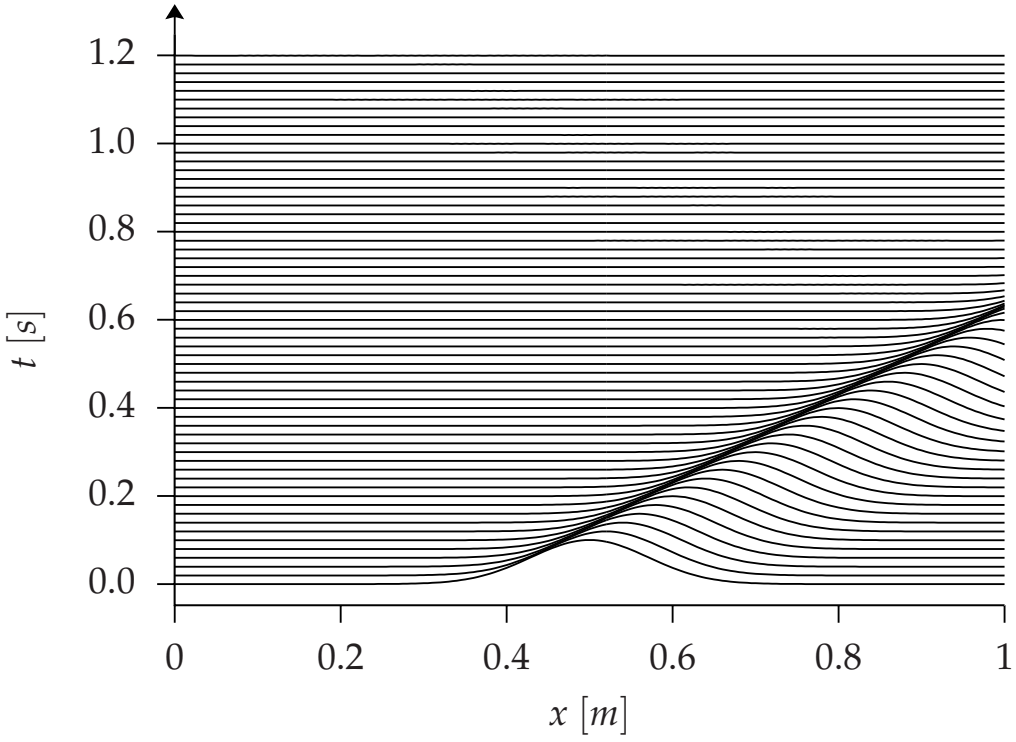
where h is the constant cell size. The characteristic boundary condition is realized by the first-order upwind scheme:

$$\left.\frac{\partial u}{\partial x}\right|_N = \frac{u_i - u_{i-1}}{h}, \tag{3.5}$$

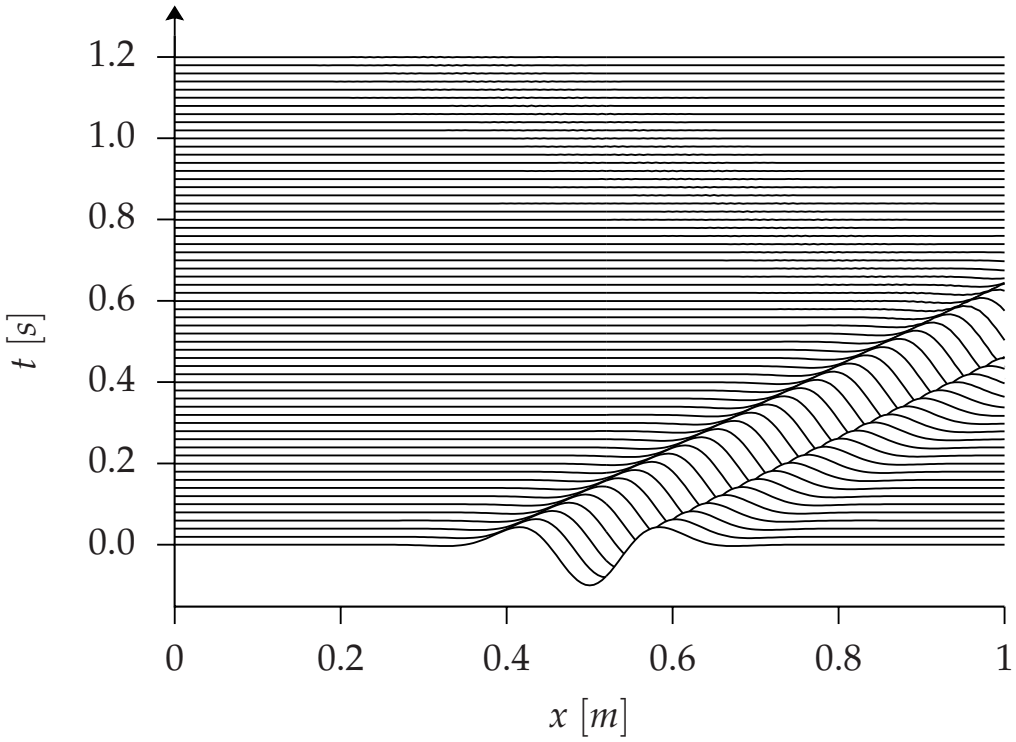
where N is the cell number of the one-dimensional domain. The time integration is performed by a low-storage fourth-order Runge-Kutta (RK4) scheme. The CFL number is taken small enough so that the numerical error associated with the time integration is negligible. The CFL number is taken equal to 0.1. Solving the advection equation with these numerical parameters, one obtains the solution represented in Fig. 3.1a, Fig. 3.1b, Fig. 3.2a and Fig. 3.2b. These solutions are given for a specific initial condition for different values of $kh \in \left\{0, \frac{\pi}{20}, \frac{\pi}{10}, \frac{\pi}{5}\right\}$:

$$u_0(x) = \cos(kx) \exp\left[-\frac{(x - x_c)^2}{R_c^2}\right], \tag{3.6}$$

where the characteristic length $R_c = 0.1\ [m]$ and $x_c = 0.5\ [m]$. In Fig. 3.1b, Fig. 3.2a and Fig. 3.2b, one can see that a high-frequency wave is reflected at the right boundary. Moreover, the amplitude of this reflected wave seems to increase as the wavenumber of the incident wave increases. Since the problem is physically well-posed, the reflection should be caused by a numerical problem. All the next sections will give an answer to this major numerical issue.

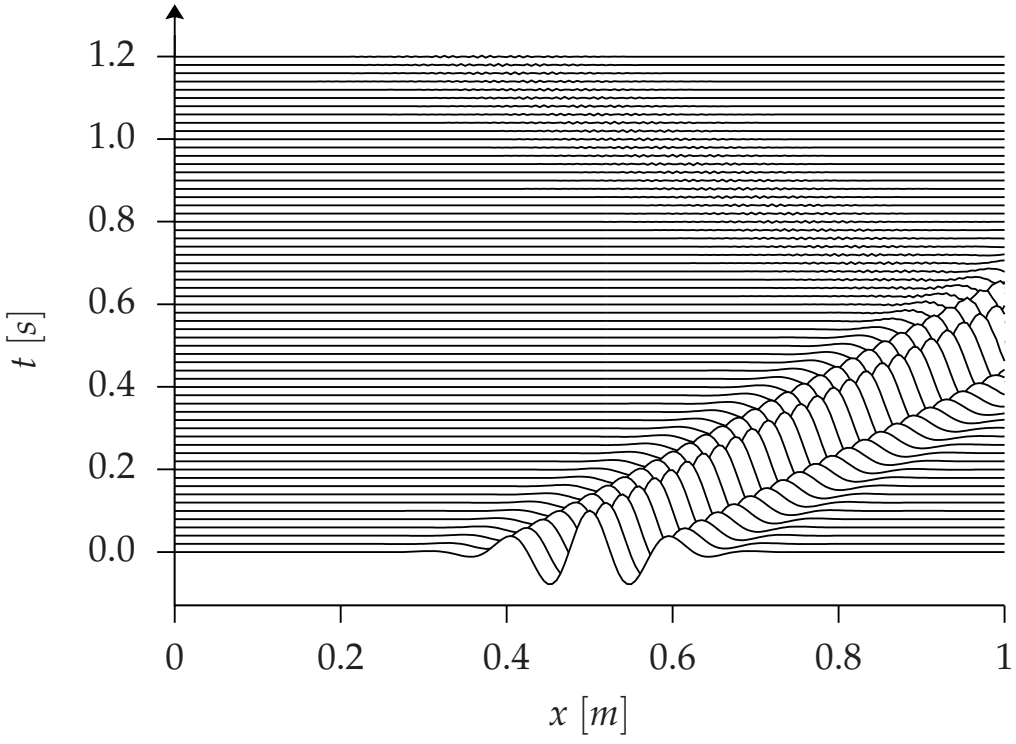


(a) $kh = 0$.

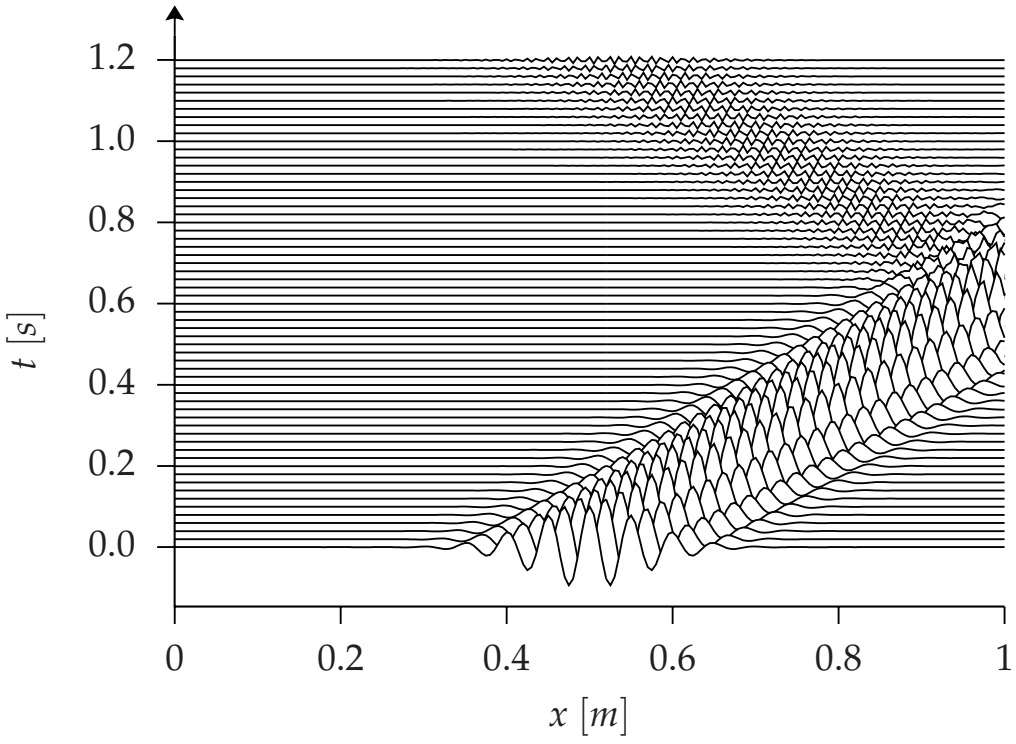


(b) $kh = \frac{\pi}{20}$.

Fig. 3.1. *Advection with Dirichlet boundary conditions.*



(a) $kh = \frac{\pi}{10}$.



(b) $kh = \frac{\pi}{5}$.

Fig. 3.2. *Advection with Dirichlet boundary conditions.*

3.2 Spectral analysis

Performing a standard spatial Fourier transform, the modified wavenumber k_m for the second-order centered scheme, which characterizes the dispersion and the dissipation is given by:

$$k_m h = \sin(kh). \tag{3.7}$$

Since the scheme is centered, there is no dissipation. The imaginary part of the modified wavenumber is indeed equal to zero. The real part of the modified wave number is shown in Fig. 3.3. At this stage, the high-frequency reflection remains unexplained.

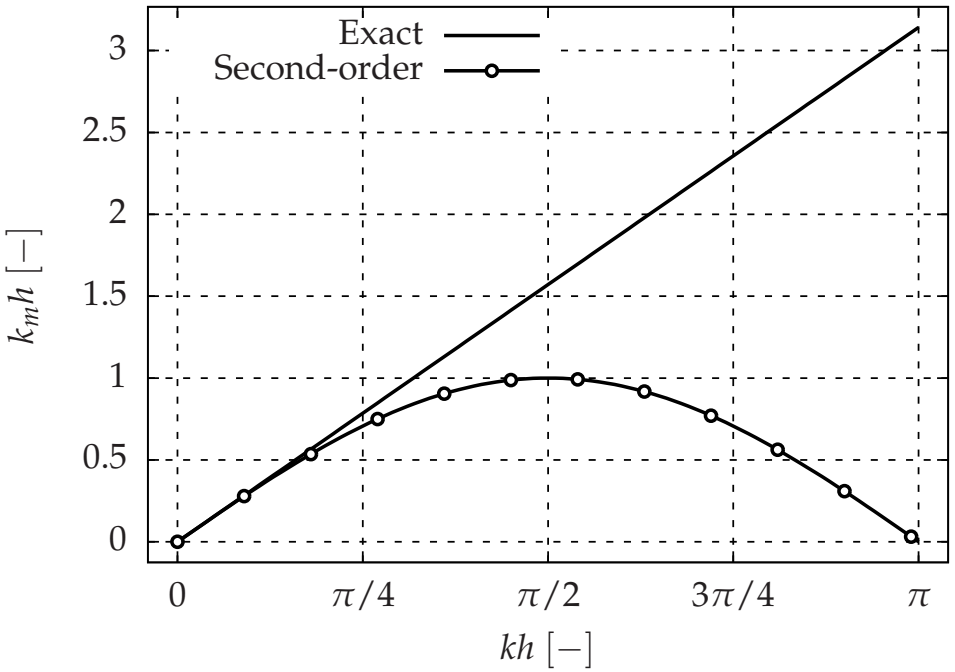
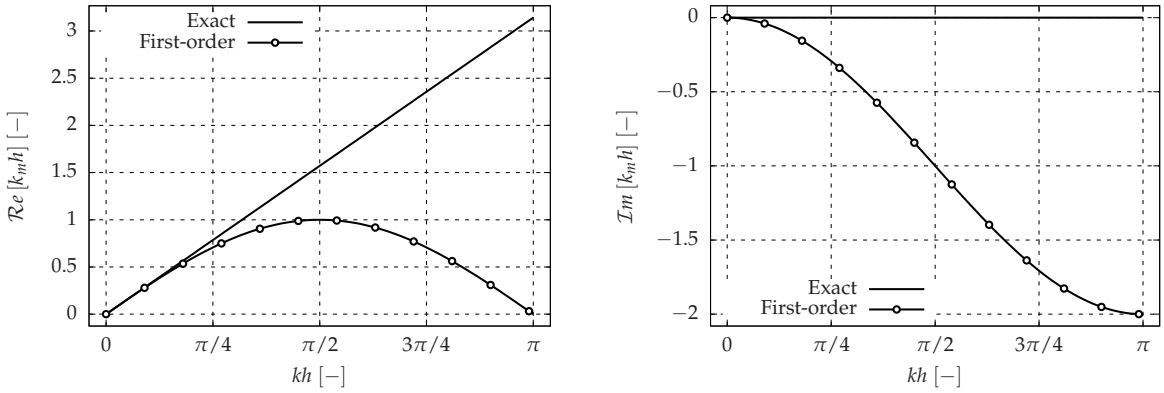


Fig. 3.3. *Dispersion for the second-order centered Finite Difference scheme.*

Again, performing a standard spatial Fourier transform, the modified wavenumber k_m for the first-order upwind scheme, which characterizes the dispersion and the dissipation is given by:

$$k_m h = \sin(kh) + j\left[\cos(kh) - 1\right]. \tag{3.8}$$

The second-order centered and the first-order upwind Finite Difference scheme have the same dispersion, as shown in Fig. 3.4a. A classical result from the literature is recovered. The first-order upwind Finite Difference scheme dissipates all wavenumbers and in particular the high wavenumbers, as shown in Fig. 3.4b. It implies stability but again, the high-frequency reflection still remains unexplained.



(a) *Dispersion for the first-order upwind Finite Difference scheme.*

(b) *Dissipation for the first-order upwind Finite Difference scheme.*

Fig. 3.4. *Spectral analysis for the first-order upwind scheme.*

3.3 Stability analysis

To perform a stability analysis, the advection equation is discretized over the whole one-dimensional domain by a second-order centered Finite Difference scheme (FDC2) for interior points, a Dirichlet boundary condition on the left boundary and a first-order upwind Finite Difference scheme (FDCU1) on the right boundary:

$$\frac{\partial \boldsymbol{U}}{\partial t} = -\frac{c}{h} A \cdot \boldsymbol{U} = -\frac{c}{h} \begin{bmatrix} h & 0 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ -1/2 & 0 & 1/2 & \ddots & & & & \vdots \\ 0 & -1/2 & 0 & 1/2 & \ddots & & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & & \vdots \\ \vdots & & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & & & \ddots & -1/2 & 0 & 1/2 & 0 \\ \vdots & & & & \ddots & -1/2 & 0 & 1/2 \\ 0 & \cdots & \cdots & \cdots & \cdots & 0 & -1 & 1 \end{bmatrix} \cdot \boldsymbol{U}, \quad (3.9)$$

where $A \left[a_{ij}\right]_{\substack{1 \leqslant i \leqslant N \\ 1 \leqslant j \leqslant N}}$ is a square matrix of size $N \times N$, a_{ij} its coefficients and $\boldsymbol{U} = [u_i]_{1 \leqslant i \leqslant N}$ is the solution vector for each mesh point. Then, the fourth-order Runge-Kutta (RK4) scheme is applied to perform the time integration. The CFL number ν is still equal to 0.1. This can be written in a matrix form:

$$\boldsymbol{U}^{n+1} = \left[I_N + \sum_{m=1}^{4} (-1)^m \frac{\nu^m}{m!} A^m \right] \cdot \boldsymbol{U}^n = G \cdot \boldsymbol{U}^n, \quad (3.10)$$

where $\boldsymbol{U}^n = \left[u_i^n\right]_{1 \leqslant i \leqslant N}$ is the solution vector for each mesh point at the time step n. Thus, one obtains a matrix which couples space and time discretizations. A geometric progression converges if the ratio is lower than 1. The spectral radius of a matrix G is the maximum among the absolute value of eigenvalues. Since (3.10) is a geometric progression, the scheme is stable if the spectral radius is smaller than 1. To obtain this spectral radius, one computes the eigenvalues of the matrix G as shown in Fig. 3.5. The cell size is equal to $h = 0.02\ [m]$ to obtain a reasonable number of eigenvalues. In Fig. 3.5, all the eigenvalues are inside the unit circle which proves the scheme stability but again, the high-frequency wave reflection still remains unexplained. At this time, several interesting conclusions can be reached that explain the necessity for Vichnevetsky's approach.

- The standard approaches are unable to predict the reflection of high-frequency waves.
- Worse: two numerical schemes, which are absolutely stable when they are taken separately, can lead to instabilities and generate the so-called spurious modes when they are coupled at a boundary condition.
- These waves can be considered as bounded instability and the standard approaches only detect unbounded instability.

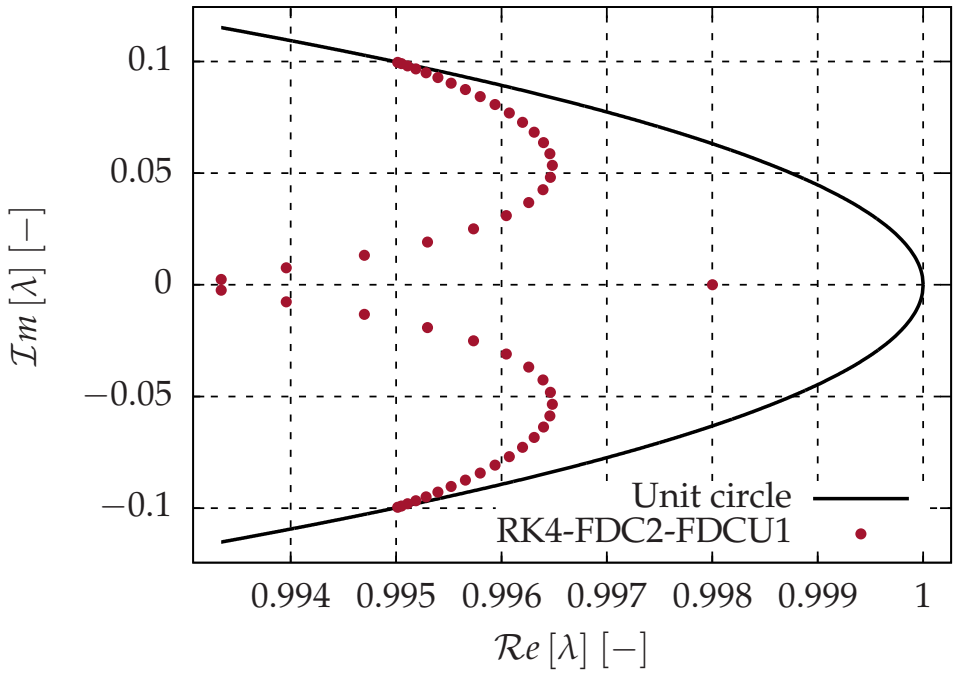


Fig. 3.5. *Stability analysis for the second-order centered Finite Difference scheme coupled with the first-order Finite Difference scheme at boundaries.*

3.4 Necessity of Vichnevetsky's framework

3.4.1 Inside the domain

Based on [5, 59, 60], let us consider the one-dimensional advection equation taken as model of hyperbolic equations.

$$\frac{\partial u}{\partial t} + c\,\frac{\partial u}{\partial x} = 0. \tag{3.11}$$

The semi-discretization is given by:

$$\frac{\mathrm{d}u_i}{\mathrm{d}t} = -c\left[\frac{u_{i+1} - u_{i-1}}{2h}\right]. \tag{3.12}$$

Let us note the time Fourier Transform of the semi-discrete numerical solution:

$$\widehat{u}_i\,(\omega) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{+\infty} u_i\,(t)\,\mathrm{e}^{-j\omega t}\,\mathrm{d}t. \tag{3.13}$$

Then, a Fourier Transform is performed in the time domain and not in the space domain where ω is the angular frequency and k the wavenumber. Taking the Fourier Transform in the time domain of (3.12):

$$j\omega\widehat{u}_i = -c\left[\frac{\widehat{u}_{i+1} - \widehat{u}_{i-1}}{2h}\right], \tag{3.14}$$

and rearranging, one obtains:

$$\widehat{u}_{i+1} + 2j\left[\frac{\omega h}{c}\right]\widehat{u}_i - \widehat{u}_{i-1} = 0, \tag{3.15}$$

which defines a progression by a recurrence equation. The solution of a geometric progression is given by the ratio defined by:

$$\widehat{u}_{i+1} = \widehat{E}(\omega) \cdot \widehat{u}_i. \tag{3.16}$$

This ratio must satisfy the characteristic equation:

$$\widehat{E}^2 + 2j\left[\frac{\omega h}{c}\right]\widehat{E} - 1 = 0. \tag{3.17}$$

It is a second-order recurrence relation that can be resolved by seeking characteristic roots. Assuming $\omega h/c \leqslant 1$ (For more details about the case where $\omega h/c \geqslant 1$, see [5]), the equation has two characteristic roots:

$$\widehat{E}_1 = -j\left[\frac{\omega h}{c}\right] + \sqrt{1 - \left[\frac{\omega h}{c}\right]^2}, \tag{3.18}$$

$$\widehat{E}_2 = -j\left[\frac{\omega h}{c}\right] - \sqrt{1 - \left[\frac{\omega h}{c}\right]^2}. \tag{3.19}$$

Note that $\left\|\widehat{E}_1\right\| = \left\|\widehat{E}_2\right\| = 1$ since the second-order centered Finite Difference scheme does not dissipate and $\widehat{E}_1^{-1} = -\widehat{E}_2$. The final solution is the sum of both fundamental solutions:

$$u_i(t) = p_i(t) + q_i(t), \tag{3.20}$$

where, using (3.16),

$$\widehat{p}_i(\omega) = \widehat{p}_0(\omega)\left[\widehat{E}_1(\omega)\right]^i, \tag{3.21}$$

$$\widehat{q}_i(\omega) = \widehat{q}_0(\omega)\left[\widehat{E}_2(\omega)\right]^i. \tag{3.22}$$

These two solutions are usually referred to as the p-wave solution and the q-wave solution. Note the interest of a Fourier Transform in the time domain and not in the space domain since p and q waves have the same angular frequency but different wavenumbers, as shown in Fig. 3.6a. As stated in [5], for comparison with (3.20), let us take the Fourier Transform in the time domain of (3.11):

$$j\omega\widehat{u} + c\frac{\partial\widehat{u}}{\partial x} = 0. \tag{3.23}$$

By analytical integration, the solution is given by:

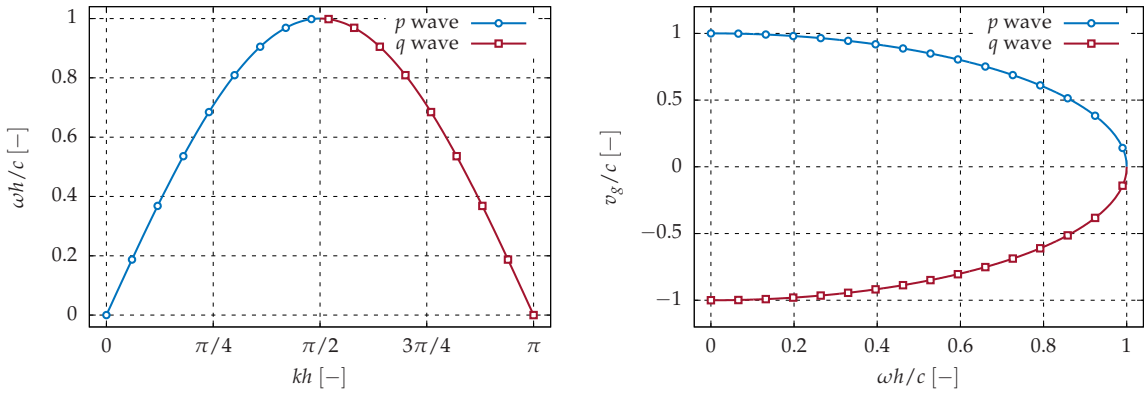
$$\widehat{u}(x,\omega) = \widehat{u}(0,\omega) \cdot \exp\left[-\frac{j\omega x}{c}\right]. \tag{3.24}$$

It is then possible to find by identification the phase velocity, the group velocity and the dispersion relation for each wave as shown in Fig. 3.6. Vichnevetsky has shown that the p wave has a positive group velocity, it propagates from left to right, and the q wave has a negative

group velocity, it propagates from right to left. In particular, the group velocity v_g^p for the p wave and v_g^q for the q wave are given by:

$$v_g^p = \quad c\sqrt{1-\left[\frac{\omega h}{c}\right]^2}, \tag{3.25}$$

$$v_g^q = -c\sqrt{1-\left[\frac{\omega h}{c}\right]^2}. \tag{3.26}$$



(a) *Dispersion relation.* **(b)** *Group velocity.*

Fig. 3.6. *Dispersion relation and group velocity of p and q waves.*

3.4.2 Reflection coefficient

A p or a q wave is solution of the semi-discretized equation inside the domain. Since the problem is linear, a linear combination of p and/or q is still a solution. Therefore a linear combination of p and q waves is sought to be a solution of the complete problem with the boundary conditions. In this case, with the second-order centered scheme and the first-order upwind scheme, the linear combination is trivially solution at all points except for the point on the boundary. Consequently only one equation will be obtained for one reflection coefficient. Now, it is possible to compute the reflection coefficient which is the ratio of the q wave by the p wave. Applying the boundary condition at the right boundary and using the semi-discretization:

$$\frac{\mathrm{d}u_N}{\mathrm{d}t} = -c\left[\frac{u_N - u_{N-1}}{h}\right], \tag{3.27}$$

Applying the temporal Fourier Transform, one obtains:

$$j\omega\widehat{u}_N = -c\left[\frac{\widehat{u}_N - \widehat{u}_{N-1}}{h}\right]. \tag{3.28}$$

Moreover, $\widehat{u}_N$ (resp. $\widehat{u}_{N-1}$) can be expressed as function of $\widehat{p}_N$ and $\widehat{q}_N$ (resp. $\widehat{p}_{N-1}$ and $\widehat{q}_{N-1}$):

$$\widehat{u}_N = \widehat{p}_N + \widehat{q}_N, \tag{3.29}$$

$$\widehat{u}_{N-1} = \widehat{p}_{N-1} + \widehat{q}_{N-1}. \tag{3.30}$$

Substituting (3.29) and (3.30) in (3.28), one obtains:

$$j\omega\left(\widehat{p}_N + \widehat{q}_N\right) = -c\left[\frac{\widehat{p}_N + \widehat{q}_N - \widehat{p}_{N-1} - \widehat{q}_{N-1}}{h}\right]. \tag{3.31}$$

Then, using both equalities $\widehat{p}_{N-1} = \widehat{p}_N \widehat{E}_1^{-1}$ and $\widehat{q}_{N-1} = \widehat{q}_N \widehat{E}_2^{-1}$, (3.31) becomes:

$$\begin{aligned} j\omega\left(\widehat{p}_N + \widehat{q}_N\right) &= -c\left[\frac{\widehat{p}_N + \widehat{q}_N - \widehat{p}_N\widehat{E}_1^{-1} - \widehat{q}_N\widehat{E}_2^{-1}}{h}\right], \\ &= -c\left[\frac{\widehat{p}_N + \widehat{q}_N + \widehat{p}_N\widehat{E}_2 + \widehat{q}_N\widehat{E}_1}{h}\right]. \end{aligned} \tag{3.32}$$

Hence, the reflection coefficient $\rho(\omega) = \dfrac{\widehat{q}_N}{\widehat{p}_N}$ is given by:

$$\rho(\omega) = \frac{\widehat{q}_N}{\widehat{p}_N} = -\frac{j\left[\frac{\omega h}{c}\right] + 1 + \widehat{E}_2}{j\left[\frac{\omega h}{c}\right] + 1 + \widehat{E}_1} = -\frac{1 - \sqrt{1 - \left[\frac{\omega h}{c}\right]^2}}{1 + \sqrt{1 - \left[\frac{\omega h}{c}\right]^2}}. \tag{3.33}$$

It is even possible to use the following second-order upwind scheme at the right boundary to decrease the amplitude of the reflected wave:

$$\frac{\mathrm{d}u_N}{\mathrm{d}t} = -c\left[\frac{3u_N - 4u_{N-1} + u_{N-2}}{2h}\right]. \tag{3.34}$$

The reflection coefficient is then:

$$\rho(\omega) = \frac{\widehat{q}_N}{\widehat{p}_N} = \frac{3 + 2j\left[\frac{\omega h}{c}\right] + 4\widehat{E}_2 + \widehat{E}_2^2}{3 + 2j\left[\frac{\omega h}{c}\right] + 4\widehat{E}_1 + \widehat{E}_1^2}. \tag{3.35}$$

3.4.3 Reflection coefficient measurement

The reflection coefficients are measured in a numerical experiment to validate the expressions given by (3.33) and (3.35). To do so, the Fourier Transform in the space domain is computed for the p and the q wave. Then, the reflection coefficient is simply computed by taking the ratio of amplitudes of Fourier Transform peaks in the space domain. A very good agreement is obtained as shown in Fig. 3.7.

3.5 Conclusion

The standard approaches are unable to predict the reflection of high-frequency waves. Even more unfortunate, when they are coupled together, two stable numerical schemes can lead to instabilities. These waves can be considered as bounded instability and the standard approaches only detect unbounded instability. Hence the necessity of adopting the Vichnevetsky's approach. The reflection of high-frequency waves can be predicted by Vichnevetsky's approach and numerically verified. More generally, the reflection of high-frequency waves appears every time there is a numerical discontinuity such as:

- Numerical schemes,
- Refinement or coarsening,
- Element type if the mesh is unstructured,
- Nonconforming grid interface,
- Hybrid nonconforming grid interface.

One possibility to avoid the reflection of high-frequency waves is to change the numerical scheme at the boundary. It has been shown, for instance, that using a second-order upwind scheme at the boundary leads to a smaller reflection coefficient compared to a first-order upwind scheme. This explains the results obtained in Chap. 2. The nonconforming grid interface should be carefully treated and this is why, in this manuscript, the nonconforming grid interface will be treated with Riemann solvers.

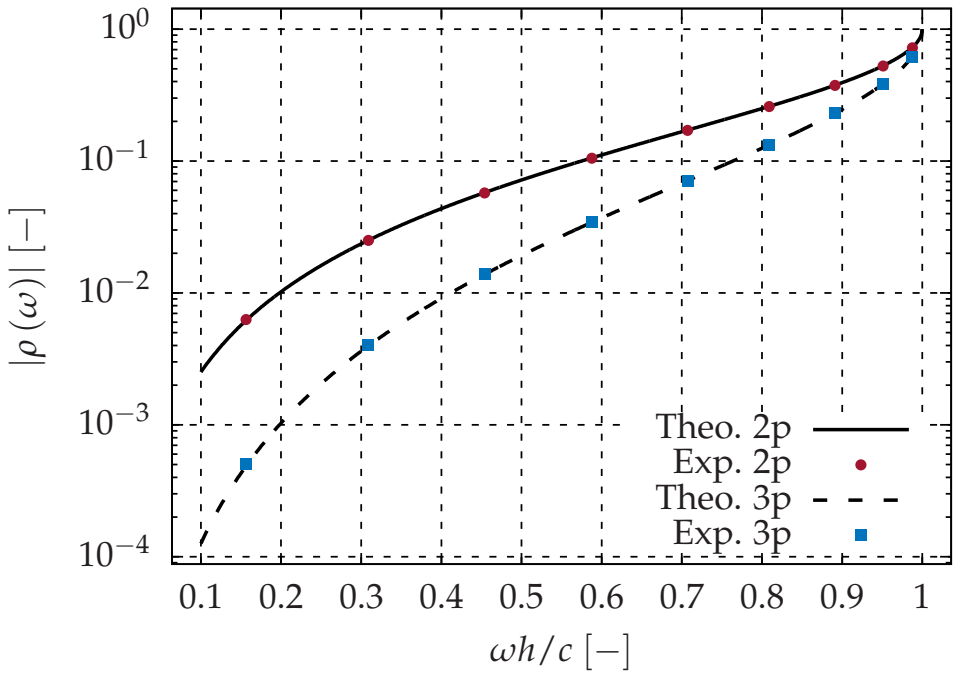


Fig. 3.7. *Reflection coefficient for the first-order upwind scheme (2p scheme) and for the second-order upwind scheme (3p scheme). Comparison with numerical experiments. A perfect agreement is obtained between theoretical and numerical results.*

III Hybrid approach validation

4 Hybrid convective fluxes 55

4.1 Preliminary remarks
4.2 The convection of a vortex: the COVO test case
4.3 Results for the first-order Roe scheme
4.4 Results for the first-order Roe scheme on a short convection distance
4.5 Results for the first-order Roe scheme with NGI on a short convection distance
4.6 Results for the second-order Roe scheme
4.7 Results for the second-order Roe scheme with NGI

5 Hybrid diffusive fluxes 85

5.1 The Taylor-Green Vortex: the TGV test case
5.2 Results
5.3 Conclusion

4 Hybrid convective fluxes

"When radium was discovered, no one knew that it would prove useful in hospitals. The work was one of pure science. And this is a proof that scientific work must not be considered from the point of view of the direct usefulness of it."

— Marie Skłodowska Curie (1867 - 1934)

Abstract. *In this chapter, the hybrid approach is validated for the convective fluxes. First of all, some general results are established to properly estimate the effective order of accuracy of a numerical scheme. Then, the test case, which deals with the convection of an isentropic compressible vortex, is used to validate the hybrid approach for the first and second-order Roe schemes. The validation is performed on several meshes with different topologies to test the implementation.*

4.1 Preliminary remarks

Let us consider the one-dimensional advection equation (4.1) where c is a constant advective velocity with $c = 1.0\ [m/s]$ and f is the dimensionless unknown.

$$\frac{\partial f}{\partial t} + c\,\frac{\partial f}{\partial x} = 0. \tag{4.1}$$

The linear hyperbolic partial derivative equation (4.1) is used to study the behavior of numerical schemes. (4.1) is discretized with a first-order upwind scheme for the convective flux and a second-order six-stage Runge-Kutta scheme RKo6s [44] for the time integration. For completeness, the RKo6s coefficients are given in Tab. 9.5. The CFL number is taken small enough so that the numerical error from the time integration is negligible. CFL $= 0.1$ is a good compromise. The resolution domain is for $x \in [0, L]$ with $L = 0.2\ [m]$ with periodic boundary conditions. The unknown at the initial time f_0 is gaussian:

$$f\left(x, t = 0\right) = f_0\left(x\right) = \exp\left[-\left(\frac{x - x_c}{R_c}\right)^2\right], \tag{4.2}$$

with $x_c = 0.05\ [m]$ and $R_c = 0.01\ [m]$. Then, the error ε_f is computed with the ℓ^2-norm for a given convection distance d:

$$\varepsilon_f^2 = \int_0^L \left[f\left(x, t\right) - f_0\left(x - ct\right)\right]^2 \mathrm{d}x. \tag{4.3}$$

In Fig. 4.1, the error ε_f versus the inverse of the number of degrees of freedom $\#DOF$ is shown for different convection distances d. Moreover, first-order slopes are also plotted to estimate the effective order of accuracy. As expected, when the distance of propagation increases, the error also increases. Also note that the effective order of accuracy decreases when the convection distance increases. The effective order of accuracy strongly depends on the convection distance

d. It means that the effective order of accuracy should be measured in a numerical computation with a small convection distance to recover the first order. This is due to the large dissipation of the first-order upwind scheme. The error is indeed bounded on one hand by the machine epsilon for high numbers of degrees of freedom, and on the other hand by the ℓ^2-norm of the initial solution for low numbers of degrees of freedom. Consequently, an error curve always has a sigmoid shape.

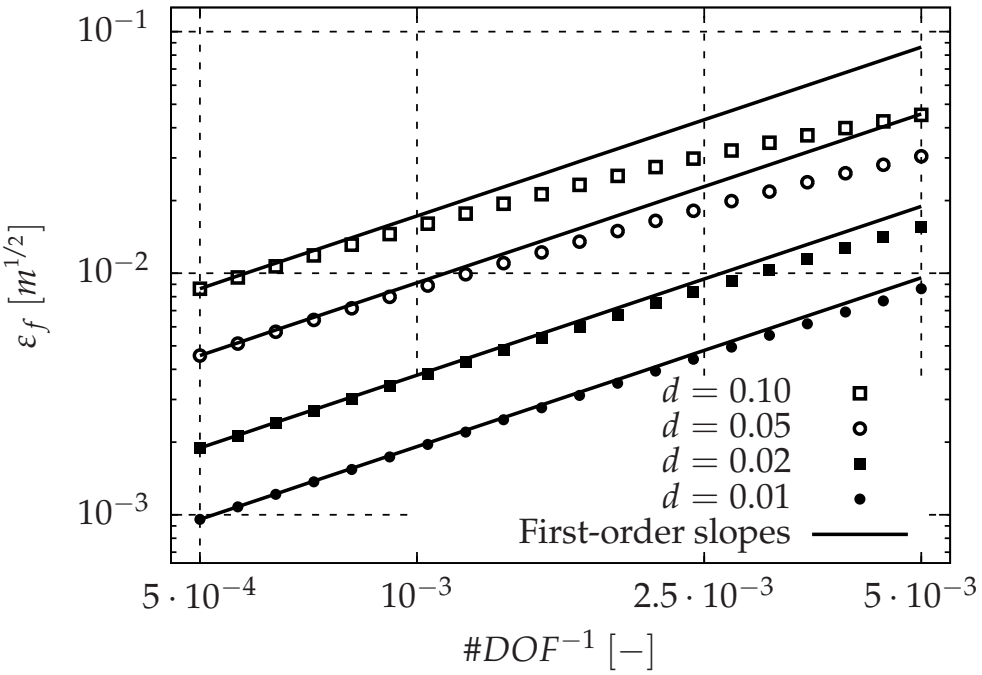


Fig. 4.1. *Effective order of accuracy for the first-order upwind scheme discretizing the one-dimensional linear advection equation. The effective order of accuracy strongly depends on the distance over which the unknown is convected.*

These results can be analytically recovered. The error ε_f can be analytically computed and is given by Theo. 4.1.1.

Theorem 4.1.1 — Analytical error. Considering a one-dimensional linear advection equation with a constant advective velocity spatially semi-discretized by the first-order upwind scheme, the numerical error ε_f with the ℓ^2-norm induced by this numerical scheme for an initial gaussian solution with a characteristic length R_c is given by:

$$\varepsilon_f^2(\eta, R_c) = R_c \sqrt{\frac{\pi}{2}} \left[\frac{1}{\sqrt{1+2\eta}} - \frac{2}{\sqrt{1+\eta}} + 1 \right], \tag{4.4}$$

where $\eta = \dfrac{d}{\Delta x} \left(\dfrac{R_c}{\Delta x} \right)^{-2}$ with d the convection distance and Δx the cell size.

This error depends on the characteristic length R_c of the initial gaussian solution and $\eta = \dfrac{d}{\Delta x} \left(\dfrac{R_c}{\Delta x} \right)^{-2}$ as shown in Fig. 4.2. η is the ratio between $\dfrac{d}{\Delta x}$ and $\left(\dfrac{R_c}{\Delta x} \right)^2$ where $\dfrac{d}{\Delta x}$ represents the dimensionless convection distance whereas $\left(\dfrac{R_c}{\Delta x} \right)^2$ represents the square of the dimensionless characteristic length of the initial gaussian solution. When the convection

distance decreases, the error decreases. When the number of degrees of freedom used to discretize the initial solution increases, the error decreases.

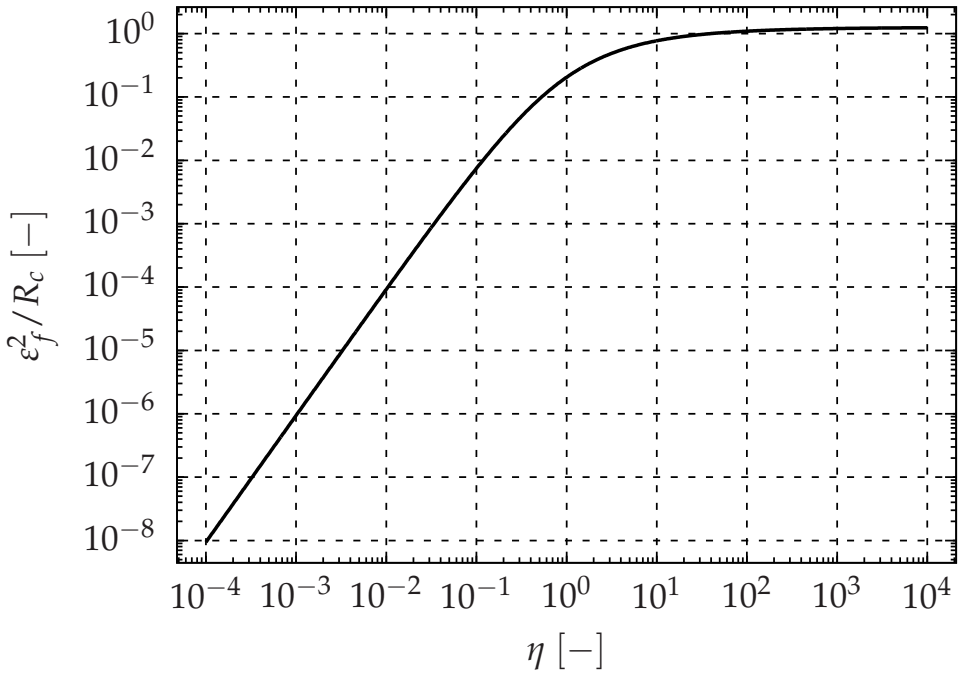


Fig. 4.2. *Ratio between ε_f^2 and R_c versus η.*

Proof 4.1.1 Considering the Taylor series expansion

$$f(x-\Delta x) = f(x) - \Delta x \frac{\partial f}{\partial x}(x) + \frac{\Delta x^2}{2}\frac{\partial^2 f}{\partial x^2}(x) + \mathcal{O}\left(\Delta x^3\right), \tag{4.5}$$

the first-order upwind scheme is recovered:

$$\frac{\partial f}{\partial x}(x) = \frac{f(x) - f(x-\Delta x)}{\Delta x} + \frac{1}{2}\Delta x \frac{\partial^2 f}{\partial x^2}(x) + \mathcal{O}\left(\Delta x^2\right). \tag{4.6}$$

Discretizing (4.1) by the first-order upwind scheme, one obtains:

$$\frac{\partial f}{\partial t} + c\frac{f(x) - f(x-\Delta x)}{\Delta x} = 0. \tag{4.7}$$

By injecting (4.6) into (4.7), the modified equation is obtained:

$$\frac{\partial f}{\partial t} + c\frac{\partial f}{\partial x} = \frac{c\Delta x}{2}\frac{\partial^2 f}{\partial x^2}(x) + \mathcal{O}\left(\Delta x^2\right). \tag{4.8}$$

The discretization of the advection equation with the first-order upwind scheme gives the solution of the advection-diffusion equation with an advective velocity c and a diffusivity $\nu = c\Delta x/2$. The diffusivity does depend on Δx which confirms that this upwind scheme is first-order accurate. Now, solving

$$\frac{\partial f}{\partial t} + c\frac{\partial f}{\partial x} = \nu\frac{\partial^2 f}{\partial x^2} \tag{4.9}$$

for $(x,t) \in \mathbb{R} \times [0,+\infty]$ with the initial condition $f_0(x) = \exp\left[-\left(\frac{x}{R_c}\right)^2\right]$, one obtains

$$f(x,t) = \frac{R_c}{\sqrt{4\nu t + R_c^2}} \exp\left[-\frac{(x-ct)^2}{4\nu t + R_c^2}\right]. \tag{4.10}$$

Finally, computing

$$\varepsilon_f^2 = \int_{-\infty}^{+\infty} \left\{ \frac{R_c}{\sqrt{4\nu t + R_c^2}} \exp\left[-\frac{(x-ct)^2}{4\nu t + R_c^2}\right] - \exp\left[-\left(\frac{x-ct}{R_c}\right)^2\right] \right\}^2 \mathrm{d}x, \tag{4.11}$$

one can recover the equation (4.4). More details can be found in App. B. ■

In Fig. 4.3, the analytical error given by (4.4) is compared to results obtained in Fig. 4.1. They are in perfect agreement and therefore the approach is validated.

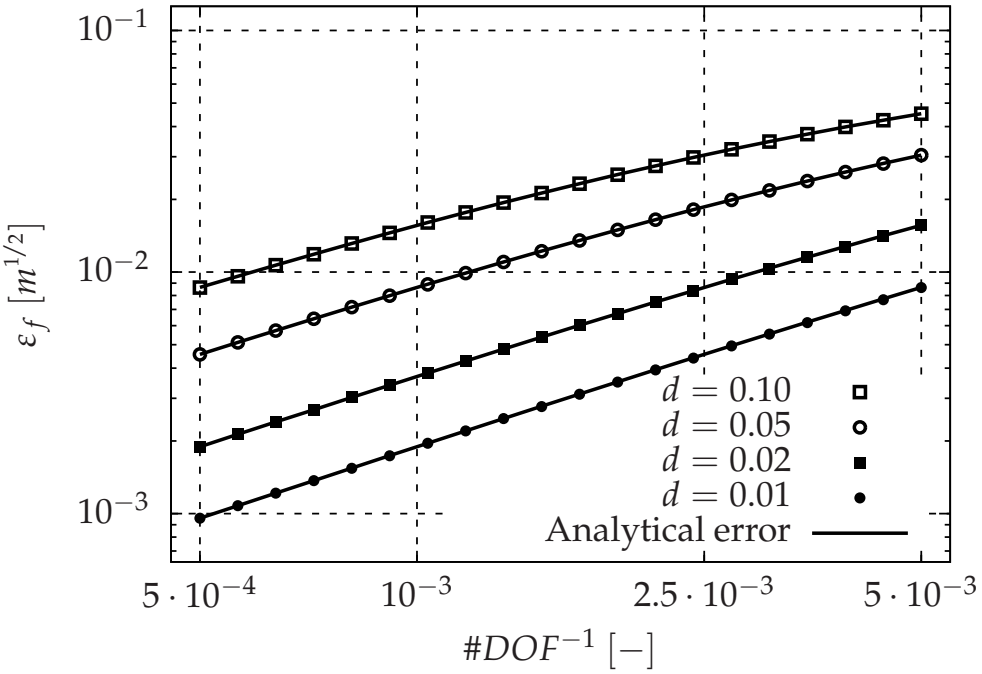


Fig. 4.3. *The analytical errors for different convection distances are compared to those obtained numerically. They are in perfect agreement.*

The order of accuracy α can then be deduced from the analytical expression of the error (4.4) in Theo. 4.1.2.

Theorem 4.1.2 — Order of accuracy. Considering a one-dimensional linear advection equation with a constant advective velocity spatially semi-discretized by the first-order upwind scheme, the effective order of accuracy α of this numerical scheme for an initial gaussian solution with a characteristic length R_c is given by:

$$\alpha = \frac{\eta}{2} \cdot \frac{(1+\eta)^{-3/2} - (1+2\eta)^{-3/2}}{(1+2\eta)^{-1/2} - 2(1+\eta)^{-1/2} + 1} \tag{4.12}$$

where $\eta = \dfrac{d}{\Delta x}\left(\dfrac{R_c}{\Delta x}\right)^{-2}$, d the convection distance and Δx the cell size.

Proof 4.1.2 The order of accuracy is defined as the slope of $\log \varepsilon_f$ with respect to $\log \eta$. Hence,

$$\alpha = \frac{\mathrm{d}\left[\log \varepsilon_f\right]}{\mathrm{d}\left[\log \eta\right]} = \frac{\mathrm{d}\left[\ln \varepsilon_f\right]}{\mathrm{d}\left[\ln \eta\right]} = \frac{\mathrm{d}\left[\ln \varepsilon_f\right]}{\mathrm{d}\varepsilon_f}\frac{\mathrm{d}\varepsilon_f}{\mathrm{d}\eta}\frac{\mathrm{d}\eta}{\mathrm{d}\left[\ln \eta\right]} = \frac{\eta}{\varepsilon_f}\frac{\mathrm{d}\varepsilon_f}{\mathrm{d}\eta}. \tag{4.13}$$

Moreover, $\dfrac{\mathrm{d}\varepsilon_f^2}{\mathrm{d}\eta} = \dfrac{\mathrm{d}\varepsilon_f^2}{\mathrm{d}\varepsilon_f}\dfrac{\mathrm{d}\varepsilon_f}{\mathrm{d}\eta} = 2\varepsilon_f\dfrac{\mathrm{d}\varepsilon_f}{\mathrm{d}\eta}$ gives $\dfrac{\mathrm{d}\varepsilon_f}{\mathrm{d}\eta} = \dfrac{1}{2\varepsilon_f}\dfrac{\mathrm{d}\varepsilon_f^2}{\mathrm{d}\eta}$. Consequently, one obtains:

$$\alpha = \frac{\eta}{2\varepsilon_f^2}\frac{\mathrm{d}\varepsilon_f^2}{\mathrm{d}\eta}, \tag{4.14}$$

$$= \frac{\eta}{2\varepsilon_f^2} R_c \sqrt{\frac{\pi}{2}}\left[\frac{\mathrm{d}}{\mathrm{d}\eta}\left[(1+2\eta)^{-1/2}\right] - 2\frac{\mathrm{d}}{\mathrm{d}\eta}\left[(1+\eta)^{-1/2}\right]\right], \tag{4.15}$$

$$= \frac{\eta}{2\varepsilon_f^2} R_c \sqrt{\frac{\pi}{2}}\left[(1+\eta)^{-3/2} - (1+2\eta)^{-3/2}\right], \tag{4.16}$$

$$= \frac{\eta}{2} \cdot \frac{(1+\eta)^{-3/2} - (1+2\eta)^{-3/2}}{(1+2\eta)^{-1/2} - 2(1+\eta)^{-1/2} + 1}. \tag{4.17}$$

■

In Fig. 4.4, this analytical order of accuracy is compared with a numerical experiment. The numerical order of accuracy is estimated thanks to a first-order upwind scheme between two consecutive refinements. They are in perfect agreement which indicates the approach relevance. To recover the order of accuracy in a simulation performed with a first-order upwind scheme, the simulation parameter has to be chosen such that η is small enough. Indeed,

$$\alpha(\eta) \underset{\eta \to 0}{\sim} 1 - \frac{5}{4}\eta + \frac{95}{48}\eta^2 - \frac{55}{16}\eta^3 + \frac{14507}{2304}\eta^4 + \mathcal{O}\left(\eta^5\right). \tag{4.18}$$

This gives best practices to measure the effective order of accuracy of a numerical scheme based on numerical computations.

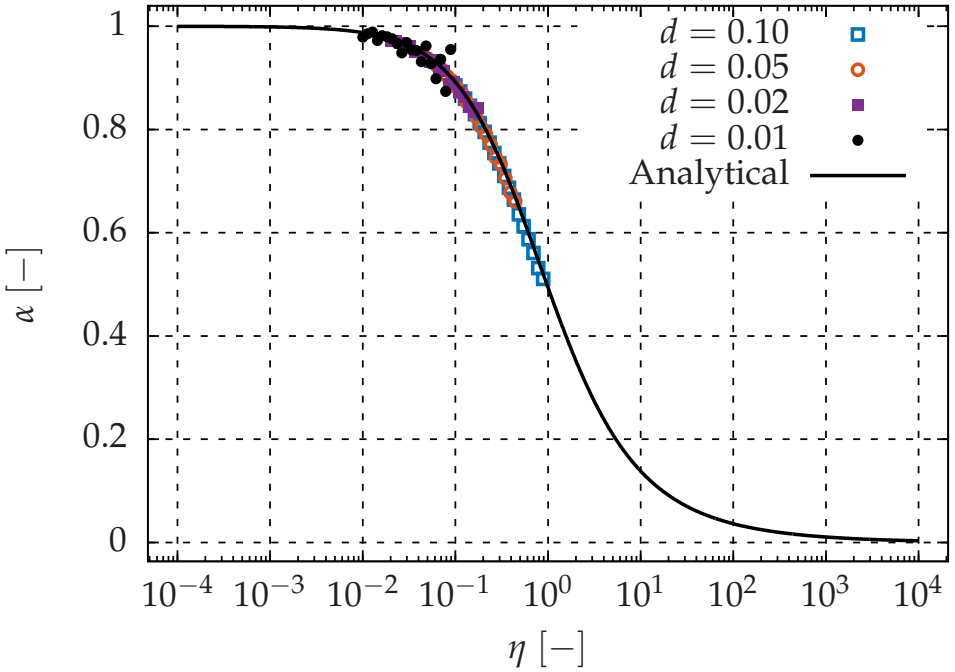


Fig. 4.4. *Order of accuracy for the first-order upwind scheme. Comparison between the analytical and numerical order of accuracy.*

4.2 The convection of a vortex: the COVO test case

4.2.1 Objectives

This problem is designed to test whether the hybrid approach preserves vorticity in an unsteady inviscid flow. Since it is an analytical solution of the unsteady Euler equations, an error analysis will be conducted to check the effective order of accuracy for a given scheme. Moreover, this error analysis will be performed on the same topological mesh but with three different formalisms: structured, unstructured and hybrid. Only the data structure changes. For a given scheme, the error analysis should give the same results on these three meshes. Indeed, the extrapolation is done with the Green-Gauss gradient for the three meshes. For unstructured meshes with a Cartesian topology, the Green-Gauss algorithm is equivalent to a directional approach such as for the structured mesh.

4.2.2 Description

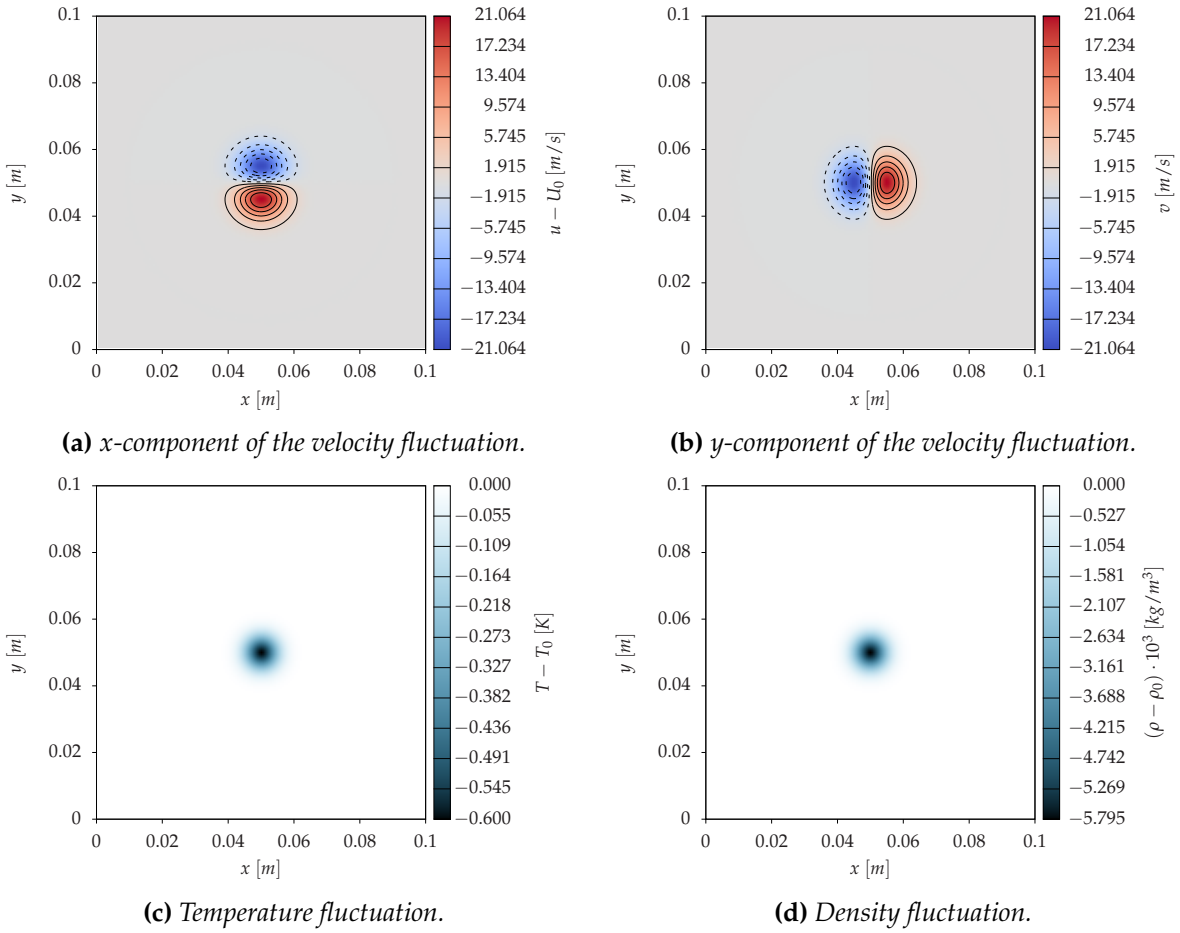
The test case deals with the convection of a compressible and isentropic vortex within a mean and constant speed flow ($p_0 = 10^5\ [Pa]$, $T_0 = 300.0\ [K]$, $M_0 = U_0/\sqrt{\gamma R_{gas} T_0} = 0.5$). This test case is extracted from the fast vortex test case of the High-Order Workshop [2]. The characteristic vortex radius R_c is equal to $0.005\ [m]$ and the vortex intensity β is equal to $0.2\ [-]$. The fluid is assumed to be a perfect gas. The ratio of specific heats γ is equal to 1.4, and the gas constant R_{gas} is equal to $287.15\ [J/kg/K]$. The vortex is initialized around the point of coordinates ($x_c = 0.05\ [m], y_c = 0.05\ [m]$) and convected over $0.1\ [m]$. For test cases with a convection over a short distance in Sec. 4.4 and Sec. 4.5, it is initialized around the point of coordinates ($x_c = 0.049\ [m], y_c = 0.05\ [m]$) and convected over $0.002\ [m]$. The flow is initialized by the velocity $\boldsymbol{V} = (u, v)^\top$ and the temperature T with:

$$u = U_0 - \frac{\beta U_0}{R_c}(y - y_c)\exp\left[-\frac{r^2}{2}\right], \tag{4.19}$$

$$v = \frac{\beta U_0}{R_c}(x - x_c)\exp\left[-\frac{r^2}{2}\right], \tag{4.20}$$

$$T = T_0 - \frac{\beta^2 U_0^2}{2C_p}\exp\left[-r^2\right], \tag{4.21}$$

where $C_p = \frac{\gamma R_{gas}}{\gamma - 1}$ and $r^2 = \frac{(x - x_c)^2 + (y - y_c)^2}{R_c^2}$. Since the vortex is isentropic, the density is computed using $\rho = \rho_0 \cdot \left(\frac{T}{T_0}\right)^{\frac{1}{\gamma - 1}}$. x- and y-component of the velocity, temperature and density fluctuations are shown in Fig. 4.5. As mentioned in the introduction, this flow is an analytical solution of the unsteady compressible Euler equations. The expected solution is the initial vortex convected without deformation.



(a) *x-component of the velocity fluctuation.*

(b) *y-component of the velocity fluctuation.*

(c) *Temperature fluctuation.*

(d) *Density fluctuation.*

Fig. 4.5. *Initial solution for the COVO test case.*

4.2.3 Meshes

The computational domain is square with $(x, y) \in [0\ [m], 0.1\ [m]] \times [0\ [m], 0.1\ [m]]$. Translational boundary conditions are imposed on the left/right and top/bottom boundaries. Nine mesh topologies have been tested and are summarized in Tab. 4.1. The computational domain is divided into three domains:

- the first domain is defined by $(x, y) \in [0.000, 0.025] \times [0, 0.1]$,
- the second domain is defined by $(x, y) \in [0.025, 0.075] \times [0, 0.1]$,
- the third domain is defined by $(x, y) \in [0.075, 0.100] \times [0, 0.1]$.

Each domain can be either Structured (S) or Unstructured (U). An Unstructured (U) domain can be composed with either Quadrangles (Q) or Triangles (T). Finally, an Unstructured (U) domain can have either perturbed elements (T) which stands for True and (F) which stands for False. A perturbed mesh is computed from a non-perturbed mesh. All nodes, except the boundary nodes, are displaced to obtain the new set of coordinates

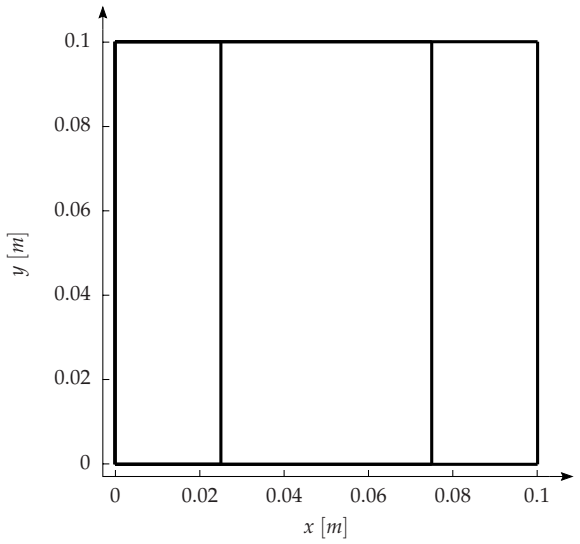
$$x_p = x + 0.15\, r\Delta x\, \cos\theta, \tag{4.22}$$
$$y_p = y + 0.15\, r\Delta x\, \sin\theta, \tag{4.23}$$

where (x, y) are the original coordinates, r a random number in $[0, 1]$ and θ a random number in $[0, 2\pi]$. The meshes are represented in Fig. 4.6 and Fig. 4.7. They are drawn for the coarse case 32×32. For hybrid meshes, the grid interface is topologically conforming but is numerically treated a nonconforming grid interface.

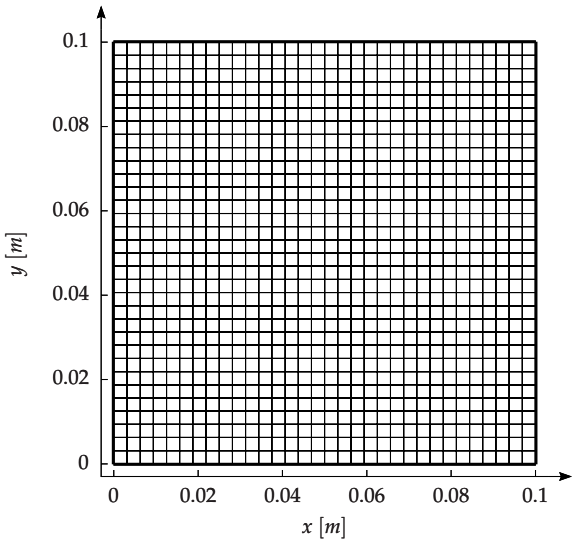
Finally, meshes with topologically nonconforming grid interface were also tested in Fig. 4.8. Regarding naming conventions, SUS/QF/NGI is the topological nonconforming version of SUS/QF and so on for other hybrid meshes. The nonconforming grid interface is ensured by adding two cells in the unstructured domain.

Tab. 4.1. *Nine mesh topologies for the order analysis.*

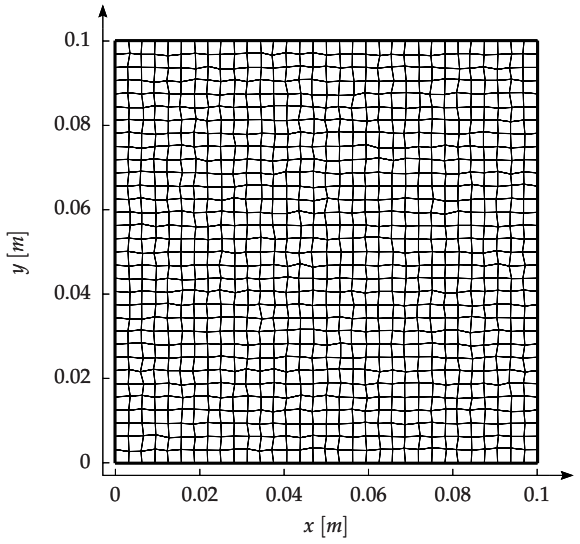
Mesh	1st domain	2nd domain	3rd domain	Elements	Perturbed
SSS	Structured	Structured	Structured	-	-
UUU/QF	Unstructured	Unstructured	Unstructured	Quadrangles	✗
SUS/QF	Structured	Unstructured	Structured	Quadrangles	✗
UUU/QT	Unstructured	Unstructured	Unstructured	Quadrangles	✓
UUU/TF	Unstructured	Unstructured	Unstructured	Triangles	✗
UUU/TT	Unstructured	Unstructured	Unstructured	Triangles	✓
SUS/QT	Structured	Unstructured	Structured	Quadrangles	✓
SUS/TF	Structured	Unstructured	Structured	Triangles	✗
SUS/TT	Structured	Unstructured	Structured	Triangles	✓



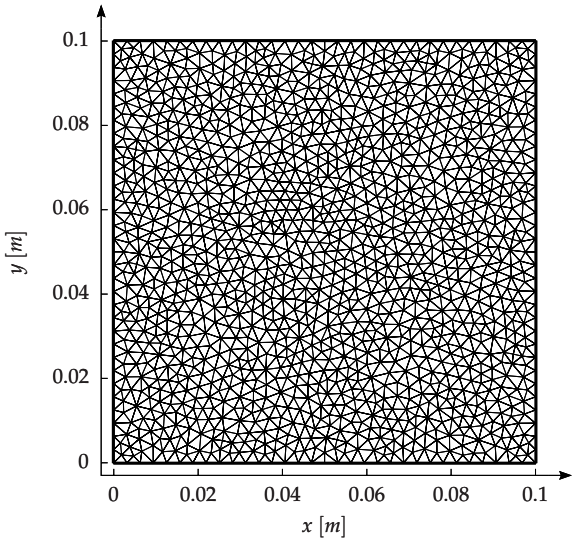
(a) *Mesh topology. The computational domain is divided into three domains.*



(b) *SSS, UUU/QF or SUS/QF mesh. Only the data structure is changed.*

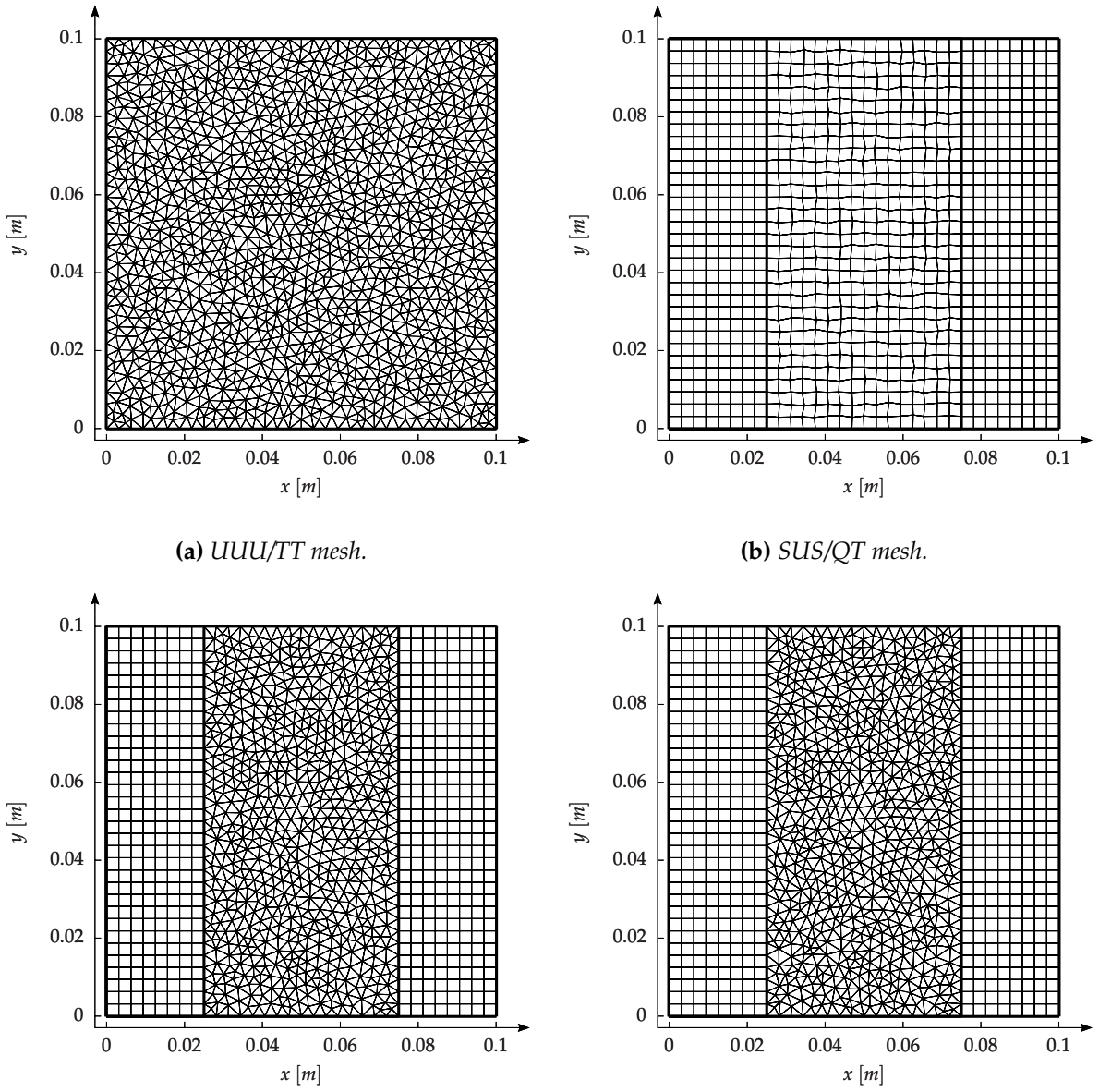


(c) *UUU/QT mesh.*



(d) *UUU/TF mesh.*

Fig. 4.6. *(I)* 32×32 *meshes for the order analysis. The grid interface for hybrid meshes is conforming but is numerically treated as a nonconforming grid interface.*



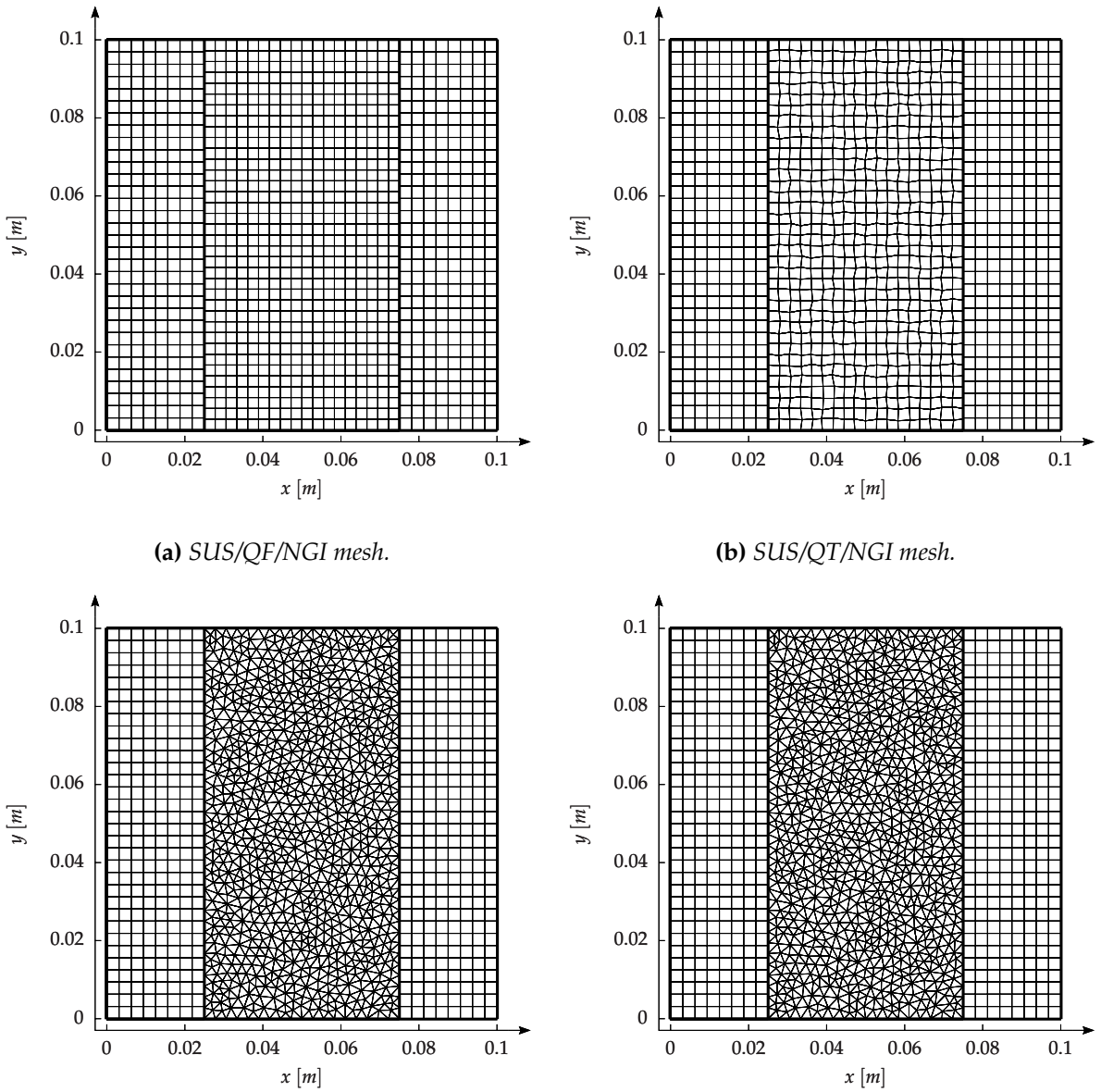
(a) *UUU/TT mesh.*

(b) *SUS/QT mesh.*

(c) *SUS/TF mesh.*

(d) *SUS/TT mesh.*

Fig. 4.7. *(II)* 32×32 *meshes for the order analysis. The grid interface for hybrid meshes is conforming but is numerically treated as a nonconforming grid interface.*



(a) *SUS/QF/NGI mesh.*

(b) *SUS/QT/NGI mesh.*

(c) *SUS/TF/NGI mesh.*

(d) *SUS/TT/NGI mesh.*

Fig. 4.8. 32×32 *meshes for the order analysis. The grid interface for hybrid meshes is nonconforming.*

4.2.4 Numerics

Even though this test case is two-dimensional, only three-dimensional computations have been performed. It has been decided to only develop and validate three-dimensional computations in this work. Depending on the mesh refinement, from 2 to 64 processors were used. The explicit time integration scheme is the second-order six stage Runge-Kutta scheme RK06s [44] where the Courant-Friedrichs-Lewy number is equal to 0.1. Time steps are chosen sufficiently small in order to measure dissipation and dispersion due to the spatial scheme and not due to the temporal integration. Space is discretized with the Roe scheme [57, 58] without any limiter introduced in Sec. 2.4 and detailed in Sec. 6.2 and Sec. 6.3. First-order and second-order extrapolations were validated. Space discretization around the nonconforming grid interface is performed with the formalism introduced in Chap. 2.

4.2.5 Error computation

The error ε_V on the velocity norm $\boldsymbol{V} = (u, v)^\top$ is computed with the ℓ^2-norm as

$$\varepsilon_V = \left[\frac{\int_\Omega (u - u_{exact})^2 + (v - v_{exact})^2 \, \mathrm{d}\mathcal{V}}{\int_\Omega \mathrm{d}\mathcal{V}} \right]^{1/2}, \tag{4.24}$$

where Ω is the computation domain and $\mathcal{V}$ the cell volume. This error ε_V can then be plotted versus $h = 1/\sqrt{\#DOF}$ where $\#DOF$ is the number of degrees of freedom in a plane located at a given z. The numerical order of accuracy is estimated using a first-order upwind Finite Difference formula between two consecutive refinements.

4.3 Results for the first-order Roe scheme

The results for the first-order Roe scheme are given from Tab. 4.2 to Tab. 4.10. Several interesting conclusions can be reached.

- The SSS, UUU/QF and SUS/QF meshes give **exactly** the same results. Indeed, the three meshes are topologically identical. Only the data structure changes. This partially validates the unstructured and hybrid approach for the first-order Roe scheme.
- Perturbating the mesh has a slight effect on the error compared with the non perturbed meshes.
- First-order accuracy is **never** reached. The corresponding $\eta = \dfrac{d}{\Delta x}\left(\dfrac{R_c}{\Delta x}\right)^{-2}$ is too large for the considered computations. As explained in Sec. 4.1, a large value of η can indeed only lead to a very bad estimation of the order of accuracy. To be able to measure the correct order of accuracy, there are several possibilities. The first one is to increase the ratio $\dfrac{R_c}{\Delta x}$ and the second one is to decrease the ratio $\dfrac{d}{\Delta x}$. Out of simplicity, the second possibility was chosen to keep the same meshes of Fig. 4.6, Fig. 4.7 and Fig. 4.8 since increasing the ratio $\dfrac{R_c}{\Delta x}$ would indeed have required an expansion of the computational domain.

Tab. 4.2. *Mesh convergence analysis on the SSS mesh for the first-order Roe scheme.*

h $[-]$	Error ε_V $[m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$6.25000000000 \cdot 10^{-2}$	$3.07883362939 \cdot 10^{0}$	–
$4.16666666667 \cdot 10^{-2}$	$2.98923141241 \cdot 10^{0}$	0.07
$3.12500000000 \cdot 10^{-2}$	$2.95386547496 \cdot 10^{0}$	0.04
$2.08333333333 \cdot 10^{-2}$	$2.87786565459 \cdot 10^{0}$	0.06
$1.56250000000 \cdot 10^{-2}$	$2.80169149728 \cdot 10^{0}$	0.09
$1.04166666667 \cdot 10^{-2}$	$2.65755833346 \cdot 10^{0}$	0.13
$7.81250000000 \cdot 10^{-3}$	$2.52557878590 \cdot 10^{0}$	0.18
$5.20833333333 \cdot 10^{-3}$	$2.29388978685 \cdot 10^{0}$	0.24
$3.90625000000 \cdot 10^{-3}$	$2.09821324754 \cdot 10^{0}$	0.31
$2.60416666667 \cdot 10^{-3}$	$1.78841646048 \cdot 10^{0}$	0.39
$1.95312500000 \cdot 10^{-3}$	$1.55597292148 \cdot 10^{0}$	0.48

Tab. 4.3. *Mesh convergence analysis on the UUU/QF mesh for the first-order Roe scheme.*

h $[-]$	Error ε_V $[m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$6.25000000000 \cdot 10^{-2}$	$3.07887844233 \cdot 10^{0}$	–
$4.16666666667 \cdot 10^{-2}$	$2.98923141241 \cdot 10^{0}$	0.07
$3.12500000000 \cdot 10^{-2}$	$2.95386547496 \cdot 10^{0}$	0.04
$2.08333333333 \cdot 10^{-2}$	$2.87786565459 \cdot 10^{0}$	0.06
$1.56250000000 \cdot 10^{-2}$	$2.80169149728 \cdot 10^{0}$	0.09
$1.04166666667 \cdot 10^{-2}$	$2.65755833346 \cdot 10^{0}$	0.13
$7.81250000000 \cdot 10^{-3}$	$2.52557878590 \cdot 10^{0}$	0.18
$5.20833333333 \cdot 10^{-3}$	$2.29388978685 \cdot 10^{0}$	0.24
$3.90625000000 \cdot 10^{-3}$	$2.09821324754 \cdot 10^{0}$	0.31
$2.60416666667 \cdot 10^{-3}$	$1.78841646048 \cdot 10^{0}$	0.39
$1.95312500000 \cdot 10^{-3}$	$1.55597292148 \cdot 10^{0}$	0.48

Tab. 4.4. *Mesh convergence analysis on the SUS/QF mesh for the first-order Roe scheme.*

h $[-]$	Error ε_V $[m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$6.25000000000 \cdot 10^{-2}$	$3.07884512383 \cdot 10^{0}$	–
$4.16666666667 \cdot 10^{-2}$	$2.98923141241 \cdot 10^{0}$	0.07
$3.12500000000 \cdot 10^{-2}$	$2.95386547496 \cdot 10^{0}$	0.04
$2.08333333333 \cdot 10^{-2}$	$2.87786565459 \cdot 10^{0}$	0.06
$1.56250000000 \cdot 10^{-2}$	$2.80169149728 \cdot 10^{0}$	0.09
$1.04166666667 \cdot 10^{-2}$	$2.65755833346 \cdot 10^{0}$	0.13
$7.81250000000 \cdot 10^{-3}$	$2.52557878590 \cdot 10^{0}$	0.18
$5.20833333333 \cdot 10^{-3}$	$2.29388978685 \cdot 10^{0}$	0.24
$3.90625000000 \cdot 10^{-3}$	$2.09821324754 \cdot 10^{0}$	0.31
$2.60416666667 \cdot 10^{-3}$	$1.78841646048 \cdot 10^{0}$	0.39
$1.95312500000 \cdot 10^{-3}$	$1.55597292148 \cdot 10^{0}$	0.48

Tab. 4.5. *Mesh convergence analysis on the UUU/QT mesh for the first-order Roe scheme.*

h $[-]$	Error ε_V $[m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$6.25000000000 \cdot 10^{-2}$	$3.13657946156 \cdot 10^{0}$	–
$4.16666666667 \cdot 10^{-2}$	$2.99642684757 \cdot 10^{0}$	0.11
$3.12500000000 \cdot 10^{-2}$	$2.96166804152 \cdot 10^{0}$	0.04
$2.08333333333 \cdot 10^{-2}$	$2.88955166167 \cdot 10^{0}$	0.06
$1.56250000000 \cdot 10^{-2}$	$2.81270980805 \cdot 10^{0}$	0.09
$1.04166666667 \cdot 10^{-2}$	$2.66918904578 \cdot 10^{0}$	0.13
$7.81250000000 \cdot 10^{-3}$	$2.53680753347 \cdot 10^{0}$	0.18
$5.20833333333 \cdot 10^{-3}$	$2.30291063598 \cdot 10^{0}$	0.24
$3.90625000000 \cdot 10^{-3}$	$2.10626371314 \cdot 10^{0}$	0.31
$2.60416666667 \cdot 10^{-3}$	$1.79477035724 \cdot 10^{0}$	0.39
$1.95312500000 \cdot 10^{-3}$	$1.56122164325 \cdot 10^{0}$	0.48

Tab. 4.6. *Mesh convergence analysis on the UUU/TF mesh for the first-order Roe scheme.*

h $[-]$	Error ε_V $[m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$3.84615384615 \cdot 10^{-2}$	$3.00507477115 \cdot 10^{0}$	–
$2.54493299280 \cdot 10^{-2}$	$2.93949372687 \cdot 10^{0}$	0.05
$1.94110729049 \cdot 10^{-2}$	$2.85506425890 \cdot 10^{0}$	0.11
$1.26836572358 \cdot 10^{-2}$	$2.70504142671 \cdot 10^{0}$	0.13
$9.63768687807 \cdot 10^{-3}$	$2.56103837738 \cdot 10^{0}$	0.20
$6.37965835292 \cdot 10^{-3}$	$2.29170630287 \cdot 10^{0}$	0.27
$4.81236634590 \cdot 10^{-3}$	$2.09155890990 \cdot 10^{0}$	0.32
$3.21130312701 \cdot 10^{-3}$	$1.74683680720 \cdot 10^{0}$	0.45
$2.40105900452 \cdot 10^{-3}$	$1.49780943705 \cdot 10^{0}$	0.53
$1.60253117779 \cdot 10^{-3}$	$1.17392104186 \cdot 10^{0}$	0.60
$1.20086012388 \cdot 10^{-3}$	$9.59134215906 \cdot 10^{-1}$	0.70

Tab. 4.7. *Mesh convergence analysis on the UUU/TT mesh for the first-order Roe scheme.*

h $[-]$	Error ε_V $[m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$3.84615384615 \cdot 10^{-2}$	$2.99923244430 \cdot 10^{0}$	–
$2.54493299280 \cdot 10^{-2}$	$2.94303946941 \cdot 10^{0}$	0.05
$1.94110729049 \cdot 10^{-2}$	$2.86199802476 \cdot 10^{0}$	0.10
$1.26836572358 \cdot 10^{-2}$	$2.71231029193 \cdot 10^{0}$	0.13
$9.63500241020 \cdot 10^{-3}$	$2.57238368168 \cdot 10^{0}$	0.19
$6.37732276494 \cdot 10^{-3}$	$2.30210274057 \cdot 10^{0}$	0.27
$4.81236634590 \cdot 10^{-3}$	$2.10559275096 \cdot 10^{0}$	0.32
$3.21130312701 \cdot 10^{-3}$	$1.76290165153 \cdot 10^{0}$	0.44
$2.40035335802 \cdot 10^{-3}$	$1.51453333346 \cdot 10^{0}$	0.52
$1.60253117779 \cdot 10^{-3}$	$1.18858735954 \cdot 10^{0}$	0.60
$1.20086012388 \cdot 10^{-3}$	$9.72451758692 \cdot 10^{-1}$	0.70

Tab. 4.8. *Mesh convergence analysis on the SUS/QT mesh for the first-order Roe scheme.*

h $[-]$	Error ε_V $[m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$6.25000000000 \cdot 10^{-2}$	$3.07472074533 \cdot 10^{0}$	–
$4.16666666667 \cdot 10^{-2}$	$2.99945888476 \cdot 10^{0}$	0.06
$3.12500000000 \cdot 10^{-2}$	$2.96217827020 \cdot 10^{0}$	0.04
$2.08333333333 \cdot 10^{-2}$	$2.88282314214 \cdot 10^{0}$	0.07
$1.56250000000 \cdot 10^{-2}$	$2.80796428336 \cdot 10^{0}$	0.09
$1.04166666667 \cdot 10^{-2}$	$2.66347090530 \cdot 10^{0}$	0.13
$7.81250000000 \cdot 10^{-3}$	$2.53124708149 \cdot 10^{0}$	0.18
$5.20833333333 \cdot 10^{-3}$	$2.29837995423 \cdot 10^{0}$	0.24
$3.90625000000 \cdot 10^{-3}$	$2.10245185231 \cdot 10^{0}$	0.31
$2.60416666667 \cdot 10^{-3}$	$1.79154482822 \cdot 10^{0}$	0.39
$1.95312500000 \cdot 10^{-3}$	$1.55859872440 \cdot 10^{0}$	0.48

Tab. 4.9. *Mesh convergence analysis on the SUS/TF mesh for the first-order Roe scheme.*

h $[-]$	Error ε_V $[m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$4.60287308949 \cdot 10^{-2}$	$3.02701061462 \cdot 10^{0}$	–
$3.07147558417 \cdot 10^{-2}$	$2.98004076025 \cdot 10^{0}$	0.04
$2.33507231247 \cdot 10^{-2}$	$2.92674749552 \cdot 10^{0}$	0.07
$1.54746115148 \cdot 10^{-2}$	$2.82392615078 \cdot 10^{0}$	0.09
$1.16122169049 \cdot 10^{-2}$	$2.72663765538 \cdot 10^{0}$	0.12
$7.71654557573 \cdot 10^{-3}$	$2.53686128212 \cdot 10^{0}$	0.18
$5.79381666209 \cdot 10^{-3}$	$2.37364330772 \cdot 10^{0}$	0.23
$3.85781338764 \cdot 10^{-3}$	$2.09145482198 \cdot 10^{0}$	0.31
$2.89537794655 \cdot 10^{-3}$	$1.86874381310 \cdot 10^{0}$	0.39
$1.92836865553 \cdot 10^{-3}$	$1.53684626487 \cdot 10^{0}$	0.48
$1.44796515279 \cdot 10^{-3}$	$1.30407214711 \cdot 10^{0}$	0.57

Tab. 4.10. *Mesh convergence analysis on the SUS/TT mesh for the first-order Roe scheme.*

h $[-]$	Error ε_V $[m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$4.60287308949 \cdot 10^{-2}$	$3.04709224148 \cdot 10^{0}$	–
$3.07147558417 \cdot 10^{-2}$	$2.97064437328 \cdot 10^{0}$	0.06
$2.33507231247 \cdot 10^{-2}$	$2.92834723092 \cdot 10^{0}$	0.05
$1.54746115148 \cdot 10^{-2}$	$2.82601329826 \cdot 10^{0}$	0.09
$1.16122169049 \cdot 10^{-2}$	$2.72887498328 \cdot 10^{0}$	0.12
$7.71424918976 \cdot 10^{-3}$	$2.54137153764 \cdot 10^{0}$	0.17
$5.79206705714 \cdot 10^{-3}$	$2.37840097827 \cdot 10^{0}$	0.23
$3.85603476049 \cdot 10^{-3}$	$2.09721593772 \cdot 10^{0}$	0.31
$2.89542649292 \cdot 10^{-3}$	$1.87354209876 \cdot 10^{0}$	0.39
$1.92804604858 \cdot 10^{-3}$	$1.54231094966 \cdot 10^{0}$	0.48
$1.44782249089 \cdot 10^{-3}$	$1.30855828018 \cdot 10^{0}$	0.57

4.4 Results for the first-order Roe scheme on a short convection distance

The results for the first-order Roe scheme on a short convection distance $L = 0.002\ [m]$ are given from Tab. 4.11 to Tab. 4.19. Several interesting conclusions can be reached.

- The SSS, UUU/QF and SUS/QF meshes give **exactly** the same results. Indeed, the three meshes are topologically identical. Only the data structure changes.
- Perturbating the mesh has a slight effect on the error compared with the non perturbed meshes.
- First-order accuracy is **always** reached and this validates the unstructured and hybrid approach for the first-order Roe scheme.

Tab. 4.11. *Mesh convergence analysis on the SSS mesh for the first-order Roe scheme on short convection distance.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$6.25000000000 \cdot 10^{-2}$	$1.05310593465 \cdot 10^{0}$	–
$4.16666666667 \cdot 10^{-2}$	$8.53279842062 \cdot 10^{-1}$	0.52
$3.12500000000 \cdot 10^{-2}$	$7.10341338625 \cdot 10^{-1}$	0.64
$2.08333333333 \cdot 10^{-2}$	$5.14470546917 \cdot 10^{-1}$	0.80
$1.56250000000 \cdot 10^{-2}$	$4.12185256984 \cdot 10^{-1}$	0.77
$1.04166666667 \cdot 10^{-2}$	$2.89780235709 \cdot 10^{-1}$	0.87
$7.81250000000 \cdot 10^{-3}$	$2.23334380280 \cdot 10^{-1}$	0.91
$5.20833333333 \cdot 10^{-3}$	$1.54324132521 \cdot 10^{-1}$	0.91
$3.90625000000 \cdot 10^{-3}$	$1.17145291459 \cdot 10^{-1}$	0.96
$2.60416666667 \cdot 10^{-3}$	$7.93784788443 \cdot 10^{-2}$	0.96
$1.95312500000 \cdot 10^{-3}$	$6.00239576834 \cdot 10^{-2}$	0.97

Tab. 4.12. *Mesh convergence analysis on the UUU/QF mesh for the first-order Roe scheme on a short convection distance.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$6.25000000000 \cdot 10^{-2}$	$1.05310593465 \cdot 10^{0}$	–
$4.16666666667 \cdot 10^{-2}$	$8.53279842062 \cdot 10^{-1}$	0.52
$3.12500000000 \cdot 10^{-2}$	$7.10341338625 \cdot 10^{-1}$	0.64
$2.08333333333 \cdot 10^{-2}$	$5.14470546917 \cdot 10^{-1}$	0.80
$1.56250000000 \cdot 10^{-2}$	$4.12185256984 \cdot 10^{-1}$	0.77
$1.04166666667 \cdot 10^{-2}$	$2.89780235709 \cdot 10^{-1}$	0.87
$7.81250000000 \cdot 10^{-3}$	$2.23334380280 \cdot 10^{-1}$	0.91
$5.20833333333 \cdot 10^{-3}$	$1.54324132521 \cdot 10^{-1}$	0.91
$3.90625000000 \cdot 10^{-3}$	$1.17145291459 \cdot 10^{-1}$	0.96
$2.60416666667 \cdot 10^{-3}$	$7.93784788443 \cdot 10^{-2}$	0.96
$1.95312500000 \cdot 10^{-3}$	$6.00239576834 \cdot 10^{-2}$	0.97

Tab. 4.13. *Mesh convergence analysis on the SUS/QF mesh for the first-order Roe scheme on a short convection distance.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$6.25000000000 \cdot 10^{-2}$	$1.05310593465 \cdot 10^{0}$	–
$4.16666666667 \cdot 10^{-2}$	$8.53279842062 \cdot 10^{-1}$	0.52
$3.12500000000 \cdot 10^{-2}$	$7.10341338625 \cdot 10^{-1}$	0.64
$2.08333333333 \cdot 10^{-2}$	$5.14470546917 \cdot 10^{-1}$	0.80
$1.56250000000 \cdot 10^{-2}$	$4.12185256984 \cdot 10^{-1}$	0.77
$1.04166666667 \cdot 10^{-2}$	$2.89780235709 \cdot 10^{-1}$	0.87
$7.81250000000 \cdot 10^{-3}$	$2.23334380280 \cdot 10^{-1}$	0.91
$5.20833333333 \cdot 10^{-3}$	$1.54324132521 \cdot 10^{-1}$	0.91
$3.90625000000 \cdot 10^{-3}$	$1.17145291459 \cdot 10^{-1}$	0.96
$2.60416666667 \cdot 10^{-3}$	$7.93784788443 \cdot 10^{-2}$	0.96
$1.95312500000 \cdot 10^{-3}$	$6.00239576834 \cdot 10^{-2}$	0.97

Tab. 4.14. *Mesh convergence analysis on the UUU/QT mesh for the first-order Roe scheme on a short convection distance.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$6.25000000000 \cdot 10^{-2}$	$1.06308011937 \cdot 10^{0}$	–
$4.16666666667 \cdot 10^{-2}$	$8.31001880442 \cdot 10^{-1}$	0.61
$3.12500000000 \cdot 10^{-2}$	$7.23279786131 \cdot 10^{-1}$	0.48
$2.08333333333 \cdot 10^{-2}$	$5.22478576079 \cdot 10^{-1}$	0.80
$1.56250000000 \cdot 10^{-2}$	$4.18355766337 \cdot 10^{-1}$	0.77
$1.04166666667 \cdot 10^{-2}$	$2.94709799188 \cdot 10^{-1}$	0.86
$7.81250000000 \cdot 10^{-3}$	$2.26946210660 \cdot 10^{-1}$	0.91
$5.20833333333 \cdot 10^{-3}$	$1.56476611732 \cdot 10^{-1}$	0.92
$3.90625000000 \cdot 10^{-3}$	$1.19044988917 \cdot 10^{-1}$	0.95
$2.60416666667 \cdot 10^{-3}$	$8.06175310904 \cdot 10^{-2}$	0.96
$1.95312500000 \cdot 10^{-3}$	$6.09583850122 \cdot 10^{-2}$	0.97

Tab. 4.15. *Mesh convergence analysis on the UUU/TF mesh for the first-order Roe scheme on a short convection distance.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$3.84615384615 \cdot 10^{-2}$	$8.13994782528 \cdot 10^{-1}$	–
$2.54493299280 \cdot 10^{-2}$	$5.84394434103 \cdot 10^{-1}$	0.80
$1.94110729049 \cdot 10^{-2}$	$4.50758787532 \cdot 10^{-1}$	0.96
$1.26836572358 \cdot 10^{-2}$	$3.00375031625 \cdot 10^{-1}$	0.95
$9.63768687807 \cdot 10^{-3}$	$2.38002982917 \cdot 10^{-1}$	0.85
$6.37965835292 \cdot 10^{-3}$	$1.61671670227 \cdot 10^{-1}$	0.94
$4.81236634590 \cdot 10^{-3}$	$1.27346023331 \cdot 10^{-1}$	0.85
$3.21130312701 \cdot 10^{-3}$	$8.54845486506 \cdot 10^{-2}$	0.99
$2.40105900452 \cdot 10^{-3}$	$6.45639459852 \cdot 10^{-2}$	0.97
$1.60253117779 \cdot 10^{-3}$	$4.44523371326 \cdot 10^{-2}$	0.92
$1.20086012388 \cdot 10^{-3}$	$3.32416417096 \cdot 10^{-2}$	1.01

Tab. 4.16. *Mesh convergence analysis on the UUU/TT mesh for the first-order Roe scheme on a short convection distance.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$3.84615384615 \cdot 10^{-2}$	$8.43237009344 \cdot 10^{-1}$	–
$2.54493299280 \cdot 10^{-2}$	$6.02945944119 \cdot 10^{-1}$	0.81
$1.94110729049 \cdot 10^{-2}$	$4.67498059719 \cdot 10^{-1}$	0.94
$1.26836572358 \cdot 10^{-2}$	$3.08400163404 \cdot 10^{-1}$	0.98
$9.63500241020 \cdot 10^{-3}$	$2.45007704824 \cdot 10^{-1}$	0.84
$6.37732276494 \cdot 10^{-3}$	$1.66353806693 \cdot 10^{-1}$	0.94
$4.81236634590 \cdot 10^{-3}$	$1.32746606032 \cdot 10^{-1}$	0.80
$3.21130312701 \cdot 10^{-3}$	$8.86458074410 \cdot 10^{-2}$	1.00
$2.40035335802 \cdot 10^{-3}$	$6.73655117608 \cdot 10^{-2}$	0.94
$1.60253117779 \cdot 10^{-3}$	$4.60035709467 \cdot 10^{-2}$	0.94
$1.20086012388 \cdot 10^{-3}$	$3.46921276001 \cdot 10^{-2}$	0.98

Tab. 4.17. *Mesh convergence analysis on the SUS/QT mesh for the first-order Roe scheme on a short convection distance.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$6.25000000000 \cdot 10^{-2}$	$1.06898067226 \cdot 10^{0}$	–
$4.16666666667 \cdot 10^{-2}$	$8.44073225721 \cdot 10^{-1}$	0.58
$3.12500000000 \cdot 10^{-2}$	$7.09435428604 \cdot 10^{-1}$	0.60
$2.08333333333 \cdot 10^{-2}$	$5.21178188660 \cdot 10^{-1}$	0.76
$1.56250000000 \cdot 10^{-2}$	$4.14626360212 \cdot 10^{-1}$	0.80
$1.04166666667 \cdot 10^{-2}$	$2.94374848900 \cdot 10^{-1}$	0.84
$7.81250000000 \cdot 10^{-3}$	$2.26422038514 \cdot 10^{-1}$	0.91
$5.20833333333 \cdot 10^{-3}$	$1.56782012667 \cdot 10^{-1}$	0.91
$3.90625000000 \cdot 10^{-3}$	$1.18975704949 \cdot 10^{-1}$	0.96
$2.60416666667 \cdot 10^{-3}$	$8.06357228203 \cdot 10^{-2}$	0.96
$1.95312500000 \cdot 10^{-3}$	$6.09745141195 \cdot 10^{-2}$	0.97

Tab. 4.18. *Mesh convergence analysis on the SUS/TF mesh for the first-order Roe scheme on a short convection distance.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$4.60287308949 \cdot 10^{-2}$	$7.81269080346 \cdot 10^{-1}$	–
$3.07147558417 \cdot 10^{-2}$	$5.47497649936 \cdot 10^{-1}$	0.88
$2.33507231247 \cdot 10^{-2}$	$4.75248993588 \cdot 10^{-1}$	0.52
$1.54746115148 \cdot 10^{-2}$	$2.95709628753 \cdot 10^{-1}$	1.15
$1.16122169049 \cdot 10^{-2}$	$2.42432135025 \cdot 10^{-1}$	0.69
$7.71654557573 \cdot 10^{-3}$	$1.60042036562 \cdot 10^{-1}$	1.02
$5.79381666209 \cdot 10^{-3}$	$1.25258610577 \cdot 10^{-1}$	0.86
$3.85781338764 \cdot 10^{-3}$	$8.64745745880 \cdot 10^{-2}$	0.91
$2.89537794655 \cdot 10^{-3}$	$6.51102758786 \cdot 10^{-2}$	0.99
$1.92836865553 \cdot 10^{-3}$	$4.38167144180 \cdot 10^{-2}$	0.97
$1.44796515279 \cdot 10^{-3}$	$3.35880942069 \cdot 10^{-2}$	0.93

Tab. 4.19. *Mesh convergence analysis on the SUS/TT mesh for the first-order Roe scheme on a short convection distance.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$4.60287308949 \cdot 10^{-2}$	$8.04462049817 \cdot 10^{-1}$	–
$3.07147558417 \cdot 10^{-2}$	$5.72854537965 \cdot 10^{-1}$	0.84
$2.33507231247 \cdot 10^{-2}$	$4.92305359193 \cdot 10^{-1}$	0.55
$1.54746115148 \cdot 10^{-2}$	$3.10769469267 \cdot 10^{-1}$	1.12
$1.16122169049 \cdot 10^{-2}$	$2.51422280268 \cdot 10^{-1}$	0.74
$7.71424918976 \cdot 10^{-3}$	$1.67862570789 \cdot 10^{-1}$	0.99
$5.79206705714 \cdot 10^{-3}$	$1.29683384334 \cdot 10^{-1}$	0.90
$3.85603476049 \cdot 10^{-3}$	$8.93627295624 \cdot 10^{-2}$	0.92
$2.89542649292 \cdot 10^{-3}$	$6.77703202366 \cdot 10^{-2}$	0.97
$1.92804604858 \cdot 10^{-3}$	$4.55536789093 \cdot 10^{-2}$	0.98
$1.44782249089 \cdot 10^{-3}$	$3.46136867841 \cdot 10^{-2}$	0.96

Fig. 4.9 shows the mesh convergence for the first-order Roe scheme. The UUU/TF and SUS/TF meshes give almost the same results. Because of the short convection distance, the vortex is mostly convected on the unstructured domain leading to the same errors. However, the number of degrees of freedom in both cases is not the same. The hybrid mesh is made of less degrees of freedom because the first and the third domains are structured. For the hybrid mesh, the sum of number of quadrangles in the first and the third domain and the number of triangles in the second domain is less than the number of triangles of the three domains of the unstructured mesh. This explains why it seems that the hybrid mesh gives more accurate results than the unstructured one.

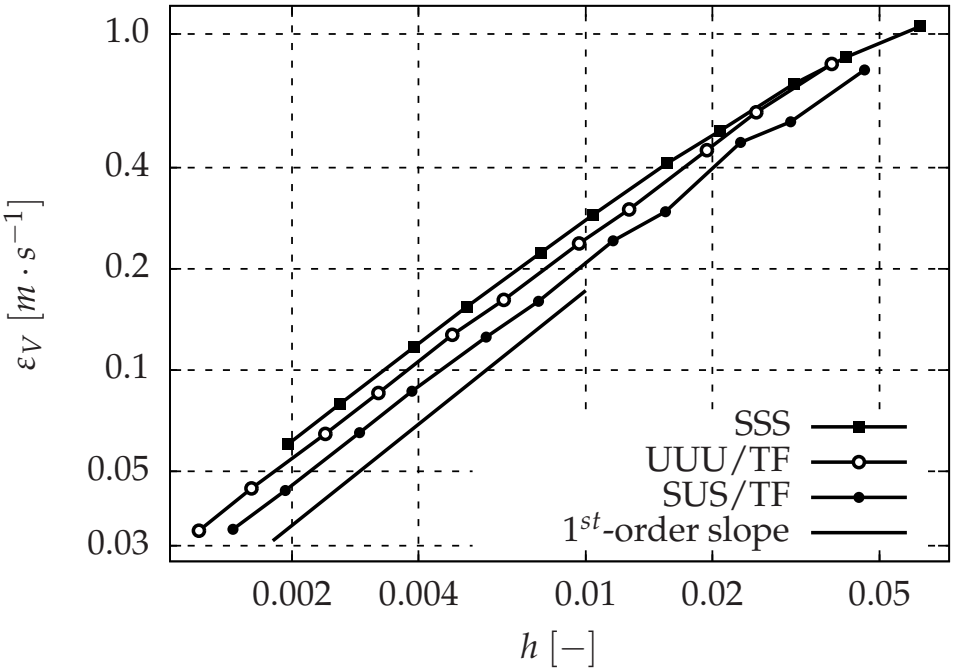


Fig. 4.9. *Mesh convergence for the first-order Roe scheme.*

However, it does not explain why the unstructured mesh seems to be more accurate than the structured one. Considering a mesh of equilateral triangles with sides a_T and a mesh

of squares with sides a_S, a first-order upwind scheme discretizing a simple linear advection equation leads to the same truncation error with the condition that the relation $\frac{\sqrt{3}}{3}a_T = a_S$ is satisfied. Because of the reduced stencil of the first-order upwind scheme, the truncation error is indeed driven by the mean distance between two degrees of freedom. In the case of equilateral triangles, this distance is equal to $\frac{\sqrt{3}}{3}a_T$ whereas in the case of squares, it is equal to a_S. The same argumentation can be found in the work of Yu *et al.* [61]. In this paper, it is stated that the solution errors come either from the space discretization, from the time discretization or from the round-off error. In this case, round-off error and time discretization (the Courant-Friedrichs-Lewy number is equal to 0.1) are negligible. Thus, if the truncation errors are identical because the relation $\frac{\sqrt{3}}{3}a_T = a_S$ is satisfied, this leads to the same solution errors. According to [61], assuming the number of triangles N_T and the number of squares N_S, the total area of triangular cells is equal to that of square cells for the same domain: $N_T \times S_T = N_S \times S_S$, where $S_T = \frac{\sqrt{3}}{4}a_T^2$ is the triangle area and $S_S = a_S^2$ is the square area.

$$N_T \times S_T = N_S \times S_S, \tag{4.25}$$

$$N_T \frac{\sqrt{3}}{4}\left(\frac{3}{\sqrt{3}}a_S\right)^2 = N_S a_S^2, \tag{4.26}$$

$$\frac{3\sqrt{3}}{4}N_T = N_S. \tag{4.27}$$

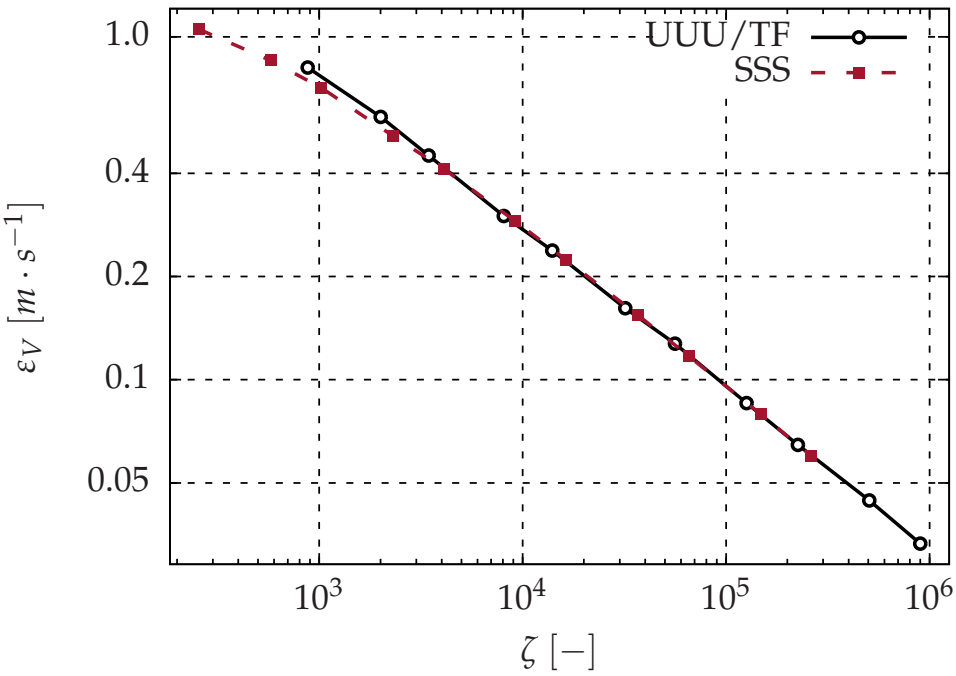


Fig. 4.10. *Mesh convergence for the first-order Roe scheme.*

Consequently, if the relation $\frac{3\sqrt{3}}{4}N_T = N_S$ is satisfied, both triangles and squares meshes discretizing the same domain lead to the same solution error. Furthermore, since $\frac{3\sqrt{3}}{4} > 1$,

this means that to obtain the same solution error, one needs less triangles than squares for a first-order upwind scheme, which is absolutely not intuitive, hence the demonstration. Of course, this can be numerically validated using the results already shown in Fig. 4.9. The error ε_V is now plotted versus $\zeta = N_S$ for the SSS mesh and $\zeta = \frac{3\sqrt{3}}{4} N_T$ for the UUU/TF mesh in Fig. 4.10. Both curves match perfectly, which validates the approach.

4.5 Results for the first-order Roe scheme with NGI on a short convection distance

The results for the first-order Roe scheme on a short convection distance $L = 0.002$ $[m]$ with nonconforming grid interfaces are given from Tab. 4.20 to Tab. 4.23. Several interesting conclusions can be reached.

- Perturbating the mesh has a slight effect on the error compared with the non perturbed meshes.
- First-order accuracy is **always** reached and this validates the unstructured and hybrid approach for the first-order Roe scheme even with nonconforming grid interfaces.

Tab. 4.20. *Mesh convergence analysis on the SUS/QF mesh for the first-order Roe scheme on a short convection distance with a nonconforming grid interface.*

h $[-]$	Error ε_V $[m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$5.52157630374 \cdot 10^{-2}$	$9.30727733702 \cdot 10^{-1}$	–
$3.83482494424 \cdot 10^{-2}$	$7.69681115267 \cdot 10^{-1}$	0.52
$2.93610109757 \cdot 10^{-2}$	$6.51801290881 \cdot 10^{-1}$	0.62
$1.99840191744 \cdot 10^{-2}$	$4.82288417351 \cdot 10^{-1}$	0.78
$1.51445633204 \cdot 10^{-2}$	$3.91541473650 \cdot 10^{-1}$	0.75
$1.02019573313 \cdot 10^{-2}$	$2.79395255903 \cdot 10^{-1}$	0.85
$7.69139752160 \cdot 10^{-3}$	$2.17108234582 \cdot 10^{-1}$	0.89
$5.15436527642 \cdot 10^{-3}$	$1.51348153818 \cdot 10^{-1}$	0.90
$3.87585253908 \cdot 10^{-3}$	$1.15418341602 \cdot 10^{-1}$	0.95
$2.59063880075 \cdot 10^{-3}$	$7.85832695108 \cdot 10^{-2}$	0.95
$1.94551056410 \cdot 10^{-3}$	$5.95685901786 \cdot 10^{-2}$	0.97

Tab. 4.21. *Mesh convergence analysis on the SUS/QT mesh for the first-order Roe scheme on a short convection distance with a nonconforming grid interface.*

h $[-]$	Error ε_V $[m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$5.52157630374 \cdot 10^{-2}$	$9.26257826794 \cdot 10^{-1}$	–
$3.83482494424 \cdot 10^{-2}$	$7.64550405629 \cdot 10^{-1}$	0.53
$2.93610109757 \cdot 10^{-2}$	$6.62656256470 \cdot 10^{-1}$	0.54
$1.99840191744 \cdot 10^{-2}$	$4.94854705962 \cdot 10^{-1}$	0.76
$1.51445633204 \cdot 10^{-2}$	$3.98193012790 \cdot 10^{-1}$	0.78
$1.02019573313 \cdot 10^{-2}$	$2.83467774294 \cdot 10^{-1}$	0.86
$7.69139752160 \cdot 10^{-3}$	$2.20600261151 \cdot 10^{-1}$	0.89
$5.15436527642 \cdot 10^{-3}$	$1.53568992515 \cdot 10^{-1}$	0.90
$3.87585253908 \cdot 10^{-3}$	$1.17142706290 \cdot 10^{-1}$	0.95
$2.59063880075 \cdot 10^{-3}$	$7.98432550816 \cdot 10^{-2}$	0.95
$1.94551056410 \cdot 10^{-3}$	$6.04618745445 \cdot 10^{-2}$	0.97

Tab. 4.22. *Mesh convergence analysis on the SUS/TF mesh for the first-order Roe scheme on a short convection distance with a nonconforming grid interface.*

h $[-]$	Error ε_V $[m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$3.92232270276 \cdot 10^{-2}$	$7.29946071905 \cdot 10^{-1}$	–
$2.75032662068 \cdot 10^{-2}$	$4.99822815450 \cdot 10^{-1}$	1.07
$2.10724875981 \cdot 10^{-2}$	$4.07299503923 \cdot 10^{-1}$	0.77
$1.45064713296 \cdot 10^{-2}$	$2.94648371101 \cdot 10^{-1}$	0.87
$1.10905919595 \cdot 10^{-2}$	$2.31551520744 \cdot 10^{-1}$	0.90
$7.47435092752 \cdot 10^{-3}$	$1.56541517267 \cdot 10^{-1}$	0.99
$5.66883950836 \cdot 10^{-3}$	$1.23518063393 \cdot 10^{-1}$	0.86
$3.79972779725 \cdot 10^{-3}$	$8.41340831758 \cdot 10^{-2}$	0.96
$2.86144409572 \cdot 10^{-3}$	$6.44520327821 \cdot 10^{-2}$	0.94
$1.91558227513 \cdot 10^{-3}$	$4.35928313548 \cdot 10^{-2}$	0.97
$1.43902023413 \cdot 10^{-3}$	$3.31537074428 \cdot 10^{-2}$	0.96

Tab. 4.23. *Mesh convergence analysis on the SUS/TT mesh for the first-order Roe scheme on a short convection distance with a nonconforming grid interface.*

h $[-]$	Error ε_V $[m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$3.92232270276 \cdot 10^{-2}$	$7.35991964124 \cdot 10^{-1}$	–
$2.75032662068 \cdot 10^{-2}$	$4.96761406813 \cdot 10^{-1}$	1.11
$2.10724875981 \cdot 10^{-2}$	$4.17705505185 \cdot 10^{-1}$	0.65
$1.45064713296 \cdot 10^{-2}$	$3.04429332751 \cdot 10^{-1}$	0.85
$1.10905919595 \cdot 10^{-2}$	$2.40333135618 \cdot 10^{-1}$	0.88
$7.47435092752 \cdot 10^{-3}$	$1.63629814828 \cdot 10^{-1}$	0.97
$5.66883950836 \cdot 10^{-3}$	$1.28786814091 \cdot 10^{-1}$	0.87
$3.79972779725 \cdot 10^{-3}$	$8.73884618527 \cdot 10^{-2}$	0.97
$2.86144409572 \cdot 10^{-3}$	$6.69902086403 \cdot 10^{-2}$	0.94
$1.91558227513 \cdot 10^{-3}$	$4.54321098989 \cdot 10^{-2}$	0.97
$1.43902023413 \cdot 10^{-3}$	$3.45193933872 \cdot 10^{-2}$	0.96

4.6 Results for the second-order Roe scheme

The results for the second-order Roe scheme are given from Tab. 4.24 to Tab. 4.32. Several interesting conclusions can be reached.

- The SSS, UUU/QF and SUS/QF meshes give **exactly** the same results. Indeed, the three meshes are topologically identical. Only the data structure changes. The extrapolation is done with the Green-Gauss gradient for the three meshes. For unstructured meshes with a Cartesian topology, the Green-Gauss algorithm is equivalent to a directional approach such as for the structured mesh. This partially validates the unstructured and hybrid approach for the second-order Roe scheme.
- The second-order of accuracy is **always** reached except for triangles and perturbed meshes.
- The perturbed meshes UUU/QT and SUS/QT only give an order between 1.5 and 2. This is even worse for triangular meshes which give a solution with a first-order accuracy. This is basically due to the false second-order extrapolation on these kinds of meshes.

- The SSS mesh gives a more accurate solution than the SUS/TF mesh than the UUU/TF mesh due to the false second-order extrapolation on triangular elements. The more the vortex is convected on quadrangle elements, the more accurate the solution is as shown in Fig. 4.11.

Tab. 4.24. *Mesh convergence analysis on the SSS mesh for the second-order Roe scheme.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$6.25000000000 \cdot 10^{-2}$	$2.81147243868 \cdot 10^{0}$	–
$4.16666666667 \cdot 10^{-2}$	$2.40888269496 \cdot 10^{0}$	0.38
$3.12500000000 \cdot 10^{-2}$	$2.06483497740 \cdot 10^{0}$	0.54
$2.08333333333 \cdot 10^{-2}$	$1.47975379073 \cdot 10^{0}$	0.82
$1.56250000000 \cdot 10^{-2}$	$1.05524464762 \cdot 10^{0}$	1.18
$1.04166666667 \cdot 10^{-2}$	$5.61839327627 \cdot 10^{-1}$	1.55
$7.81250000000 \cdot 10^{-3}$	$3.29932391887 \cdot 10^{-1}$	1.85
$5.20833333333 \cdot 10^{-3}$	$1.47463873979 \cdot 10^{-1}$	1.99
$3.90625000000 \cdot 10^{-3}$	$8.23748502605 \cdot 10^{-2}$	2.02
$2.60416666667 \cdot 10^{-3}$	$3.62812076092 \cdot 10^{-2}$	2.02
$1.95312500000 \cdot 10^{-3}$	$2.03238752709 \cdot 10^{-2}$	2.01

Tab. 4.25. *Mesh convergence analysis on the UUU/QF mesh for the second-order Roe scheme.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$6.25000000000 \cdot 10^{-2}$	$2.81146808423 \cdot 10^{0}$	–
$4.16666666667 \cdot 10^{-2}$	$2.40888267662 \cdot 10^{0}$	0.38
$3.12500000000 \cdot 10^{-2}$	$2.06483494947 \cdot 10^{0}$	0.54
$2.08333333333 \cdot 10^{-2}$	$1.47975375162 \cdot 10^{0}$	0.82
$1.56250000000 \cdot 10^{-2}$	$1.05524460570 \cdot 10^{0}$	1.18
$1.04166666667 \cdot 10^{-2}$	$5.61839290189 \cdot 10^{-1}$	1.55
$7.81250000000 \cdot 10^{-3}$	$3.29932362237 \cdot 10^{-1}$	1.85
$5.20833333333 \cdot 10^{-3}$	$1.47463856382 \cdot 10^{-1}$	1.99
$3.90625000000 \cdot 10^{-3}$	$8.23748392103 \cdot 10^{-2}$	2.02
$2.60416666667 \cdot 10^{-3}$	$3.62812022983 \cdot 10^{-2}$	2.02
$1.95312500000 \cdot 10^{-3}$	$2.03238722212 \cdot 10^{-2}$	2.01

Tab. 4.26. *Mesh convergence analysis on the SUS/QF mesh for the second-order Roe scheme.*

h $[-]$	Error ε_V $[m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$6.25000000000 \cdot 10^{-2}$	$2.81146798894 \cdot 10^{0}$	–
$4.16666666667 \cdot 10^{-2}$	$2.40888269496 \cdot 10^{0}$	0.38
$3.12500000000 \cdot 10^{-2}$	$2.06483497740 \cdot 10^{0}$	0.54
$2.08333333333 \cdot 10^{-2}$	$1.47975379073 \cdot 10^{0}$	0.82
$1.56250000000 \cdot 10^{-2}$	$1.05524464763 \cdot 10^{0}$	1.18
$1.04166666667 \cdot 10^{-2}$	$5.61839327648 \cdot 10^{-1}$	1.55
$7.81250000000 \cdot 10^{-3}$	$3.29932391917 \cdot 10^{-1}$	1.85
$5.20833333333 \cdot 10^{-3}$	$1.47463874019 \cdot 10^{-1}$	1.99
$3.90625000000 \cdot 10^{-3}$	$8.23748503035 \cdot 10^{-2}$	2.02
$2.60416666667 \cdot 10^{-3}$	$3.62812076547 \cdot 10^{-2}$	2.02
$1.95312500000 \cdot 10^{-3}$	$2.03238753171 \cdot 10^{-2}$	2.01

Tab. 4.27. *Mesh convergence analysis on the UUU/QT mesh for the second-order Roe scheme.*

h $[-]$	Error ε_V $[m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$6.25000000000 \cdot 10^{-2}$	$2.88647090597 \cdot 10^{0}$	–
$4.16666666667 \cdot 10^{-2}$	$2.44362757826 \cdot 10^{0}$	0.41
$3.12500000000 \cdot 10^{-2}$	$2.11164857274 \cdot 10^{0}$	0.51
$2.08333333333 \cdot 10^{-2}$	$1.52772015522 \cdot 10^{0}$	0.80
$1.56250000000 \cdot 10^{-2}$	$1.11023874943 \cdot 10^{0}$	1.11
$1.04166666667 \cdot 10^{-2}$	$6.08577021354 \cdot 10^{-1}$	1.48
$7.81250000000 \cdot 10^{-3}$	$3.71996670731 \cdot 10^{-1}$	1.71
$5.20833333333 \cdot 10^{-3}$	$1.74003520576 \cdot 10^{-1}$	1.87
$3.90625000000 \cdot 10^{-3}$	$1.02042297180 \cdot 10^{-1}$	1.86
$2.60416666667 \cdot 10^{-3}$	$4.99514171466 \cdot 10^{-2}$	1.76
$1.95312500000 \cdot 10^{-3}$	$3.15623421777 \cdot 10^{-2}$	1.60

Tab. 4.28. *Mesh convergence analysis on the UUU/TF mesh for the second-order Roe scheme.*

h $[-]$	Error ε_V $[m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$3.84615384615 \cdot 10^{-2}$	$2.44112519913 \cdot 10^{0}$	–
$2.54493299280 \cdot 10^{-2}$	$1.79410699784 \cdot 10^{0}$	0.75
$1.94110729049 \cdot 10^{-2}$	$1.35421079748 \cdot 10^{0}$	1.04
$1.26836572358 \cdot 10^{-2}$	$7.76326882690 \cdot 10^{-1}$	1.31
$9.63768687807 \cdot 10^{-3}$	$4.79899421057 \cdot 10^{-1}$	1.75
$6.37965835292 \cdot 10^{-3}$	$2.82269286029 \cdot 10^{-1}$	1.29
$4.81236634590 \cdot 10^{-3}$	$1.96098661362 \cdot 10^{-1}$	1.29
$3.21130312701 \cdot 10^{-3}$	$1.12699162782 \cdot 10^{-1}$	1.37
$2.40105900452 \cdot 10^{-3}$	$8.61692652399 \cdot 10^{-2}$	0.92
$1.60253117779 \cdot 10^{-3}$	$5.60737726277 \cdot 10^{-2}$	1.06
$1.20086012388 \cdot 10^{-3}$	$4.17267636645 \cdot 10^{-2}$	1.02

Tab. 4.29. *Mesh convergence analysis on the UUU/TT mesh for the second-order Roe scheme.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$3.84615384615 \cdot 10^{-2}$	$2.60949461942 \cdot 10^{0}$	–
$2.54493299280 \cdot 10^{-2}$	$1.85371528407 \cdot 10^{0}$	0.83
$1.94110729049 \cdot 10^{-2}$	$1.49117966005 \cdot 10^{0}$	0.80
$1.26836572358 \cdot 10^{-2}$	$8.99768938655 \cdot 10^{-1}$	1.19
$9.63500241020 \cdot 10^{-3}$	$6.25913618564 \cdot 10^{-1}$	1.32
$6.37732276494 \cdot 10^{-3}$	$3.69604325935 \cdot 10^{-1}$	1.28
$4.81236634590 \cdot 10^{-3}$	$2.78206205318 \cdot 10^{-1}$	1.01
$3.21130312701 \cdot 10^{-3}$	$1.76270538302 \cdot 10^{-1}$	1.13
$2.40035335802 \cdot 10^{-3}$	$1.37442173938 \cdot 10^{-1}$	0.85
$1.60253117779 \cdot 10^{-3}$	$8.98958691494 \cdot 10^{-2}$	1.05
$1.20086012388 \cdot 10^{-3}$	$6.83276982272 \cdot 10^{-2}$	0.95

Tab. 4.30. *Mesh convergence analysis on the SUS/QT mesh for the second-order Roe scheme.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$6.25000000000 \cdot 10^{-2}$	$2.81916562394 \cdot 10^{0}$	–
$4.16666666667 \cdot 10^{-2}$	$2.43104131195 \cdot 10^{0}$	0.37
$3.12500000000 \cdot 10^{-2}$	$2.09168258677 \cdot 10^{0}$	0.52
$2.08333333333 \cdot 10^{-2}$	$1.49695688810 \cdot 10^{0}$	0.83
$1.56250000000 \cdot 10^{-2}$	$1.08106825502 \cdot 10^{0}$	1.13
$1.04166666667 \cdot 10^{-2}$	$5.86363910906 \cdot 10^{-1}$	1.51
$7.81250000000 \cdot 10^{-3}$	$3.50237123459 \cdot 10^{-1}$	1.79
$5.20833333333 \cdot 10^{-3}$	$1.58831282792 \cdot 10^{-1}$	1.95
$3.90625000000 \cdot 10^{-3}$	$9.13665249969 \cdot 10^{-2}$	1.92
$2.60416666667 \cdot 10^{-3}$	$4.26585386637 \cdot 10^{-2}$	1.88
$1.95312500000 \cdot 10^{-3}$	$2.52560327165 \cdot 10^{-2}$	1.82

Tab. 4.31. *Mesh convergence analysis on the SUS/TF mesh for the second-order Roe scheme.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$4.60287308949 \cdot 10^{-2}$	$2.61823113285 \cdot 10^{0}$	–
$3.07147558417 \cdot 10^{-2}$	$2.16674316292 \cdot 10^{0}$	0.47
$2.33507231247 \cdot 10^{-2}$	$1.74916375702 \cdot 10^{0}$	0.78
$1.54746115148 \cdot 10^{-2}$	$1.13030380756 \cdot 10^{0}$	1.06
$1.16122169049 \cdot 10^{-2}$	$7.67315788067 \cdot 10^{-1}$	1.35
$7.71654557573 \cdot 10^{-3}$	$3.81991363863 \cdot 10^{-1}$	1.71
$5.79381666209 \cdot 10^{-3}$	$2.25734341898 \cdot 10^{-1}$	1.84
$3.85781338764 \cdot 10^{-3}$	$1.13606585427 \cdot 10^{-1}$	1.69
$2.89537794655 \cdot 10^{-3}$	$7.26525343186 \cdot 10^{-2}$	1.56
$1.92836865553 \cdot 10^{-3}$	$4.11236953310 \cdot 10^{-2}$	1.40
$1.44796515279 \cdot 10^{-3}$	$2.97592473536 \cdot 10^{-2}$	1.13

Tab. 4.32. *Mesh convergence analysis on the SUS/TT mesh for the second-order Roe scheme.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$4.60287308949 \cdot 10^{-2}$	$2.67179464940 \cdot 10^{0}$	–
$3.07147558417 \cdot 10^{-2}$	$2.19035418978 \cdot 10^{0}$	0.49
$2.33507231247 \cdot 10^{-2}$	$1.80204902284 \cdot 10^{0}$	0.71
$1.54746115148 \cdot 10^{-2}$	$1.19355674754 \cdot 10^{0}$	1.00
$1.16122169049 \cdot 10^{-2}$	$8.29808675508 \cdot 10^{-1}$	1.27
$7.71424918976 \cdot 10^{-3}$	$4.34378021666 \cdot 10^{-1}$	1.58
$5.79206705714 \cdot 10^{-3}$	$2.71267493161 \cdot 10^{-1}$	1.64
$3.85603476049 \cdot 10^{-3}$	$1.44858849075 \cdot 10^{-1}$	1.54
$2.89542649292 \cdot 10^{-3}$	$9.76450302275 \cdot 10^{-2}$	1.38
$1.92804604858 \cdot 10^{-3}$	$5.98568059585 \cdot 10^{-2}$	1.20
$1.44782249089 \cdot 10^{-3}$	$4.26284494967 \cdot 10^{-2}$	1.18

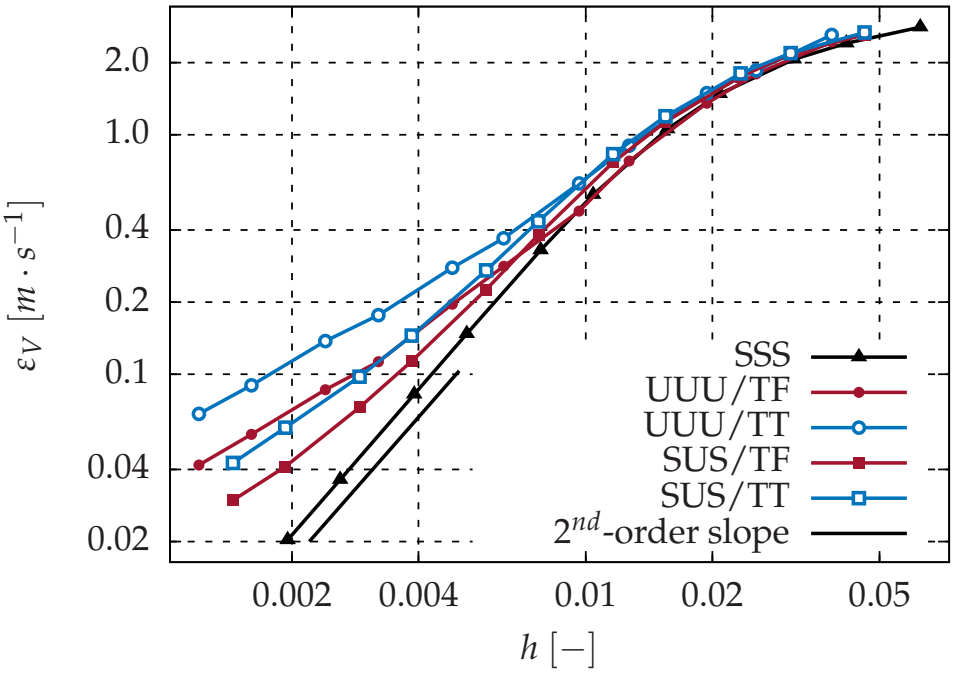


Fig. 4.11. *Mesh convergence for the second-order Roe scheme.*

4.7 Results for the second-order Roe scheme with NGI

The results for the second-order Roe scheme with nonconforming grid interfaces are given from Tab. 4.33 to Tab. 4.36. Several interesting conclusions can be reached.

- The second-order of accuracy is **only** reached for the SUS/QF mesh.
- The perturbed mesh SUS/QT only gives an order between 1.5 and 2. This is even worse for triangular meshes which give a solution with a first-order accuracy. Again, this is basically due to the false second-order extrapolation on these kinds of meshes.
- The nonconforming grid interface does not change the results obtained in Sec. 4.6, which validates the approach for the second-order Roe scheme.

Tab. 4.33. *Mesh convergence analysis on the SUS/QF mesh for the second-order Roe scheme.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$5.52157630374 \cdot 10^{-2}$	$2.69859453278 \cdot 10^{0}$	–
$3.83482494424 \cdot 10^{-2}$	$2.34862127438 \cdot 10^{0}$	0.38
$2.93610109757 \cdot 10^{-2}$	$2.00398056363 \cdot 10^{0}$	0.59
$1.99840191744 \cdot 10^{-2}$	$1.42871107619 \cdot 10^{0}$	0.88
$1.51445633204 \cdot 10^{-2}$	$1.01747718790 \cdot 10^{0}$	1.22
$1.02019573313 \cdot 10^{-2}$	$5.43641412663 \cdot 10^{-1}$	1.59
$7.69139752160 \cdot 10^{-3}$	$3.21004896133 \cdot 10^{-1}$	1.87
$5.15436527642 \cdot 10^{-3}$	$1.44650832286 \cdot 10^{-1}$	1.99
$3.87585253908 \cdot 10^{-3}$	$8.11839544528 \cdot 10^{-2}$	2.03
$2.59063880075 \cdot 10^{-3}$	$3.59301304915 \cdot 10^{-2}$	2.02
$1.94551056410 \cdot 10^{-3}$	$2.01761869114 \cdot 10^{-2}$	2.02

Tab. 4.34. *Mesh convergence analysis on the SUS/QT mesh for the second-order Roe scheme.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$5.52157630374 \cdot 10^{-2}$	$2.69911316214 \cdot 10^{0}$	–
$3.83482494424 \cdot 10^{-2}$	$2.37191980892 \cdot 10^{0}$	0.35
$2.93610109757 \cdot 10^{-2}$	$2.02940792717 \cdot 10^{0}$	0.58
$1.99840191744 \cdot 10^{-2}$	$1.45863919094 \cdot 10^{0}$	0.86
$1.51445633204 \cdot 10^{-2}$	$1.04933806324 \cdot 10^{0}$	1.19
$1.02019573313 \cdot 10^{-2}$	$5.61030066516 \cdot 10^{-1}$	1.58
$7.69139752160 \cdot 10^{-3}$	$3.39272090329 \cdot 10^{-1}$	1.78
$5.15436527642 \cdot 10^{-3}$	$1.55965344749 \cdot 10^{-1}$	1.94
$3.87585253908 \cdot 10^{-3}$	$9.06490198650 \cdot 10^{-2}$	1.90
$2.59063880075 \cdot 10^{-3}$	$4.22903928713 \cdot 10^{-2}$	1.89
$1.94551056410 \cdot 10^{-3}$	$2.52570366494 \cdot 10^{-2}$	1.80

Tab. 4.35. *Mesh convergence analysis on the SUS/TF mesh for the second-order Roe scheme.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$3.92232270276 \cdot 10^{-2}$	$2.58177861153 \cdot 10^{0}$	–
$2.75032662068 \cdot 10^{-2}$	$2.17006249623 \cdot 10^{0}$	0.49
$2.10724875981 \cdot 10^{-2}$	$1.74798733054 \cdot 10^{0}$	0.81
$1.45064713296 \cdot 10^{-2}$	$1.13430841708 \cdot 10^{0}$	1.16
$1.10905919595 \cdot 10^{-2}$	$7.60544582398 \cdot 10^{-1}$	1.49
$7.47435092752 \cdot 10^{-3}$	$3.75927964556 \cdot 10^{-1}$	1.79
$5.66883950836 \cdot 10^{-3}$	$2.30320926856 \cdot 10^{-1}$	1.77
$3.79972779725 \cdot 10^{-3}$	$1.14157683699 \cdot 10^{-1}$	1.75
$2.86144409572 \cdot 10^{-3}$	$7.31421349101 \cdot 10^{-2}$	1.57
$1.91558227513 \cdot 10^{-3}$	$4.11078114818 \cdot 10^{-2}$	1.44
$1.43902023413 \cdot 10^{-3}$	$2.95310259562 \cdot 10^{-2}$	1.16

Tab. 4.36. *Mesh convergence analysis on the SUS/TT mesh for the second-order Roe scheme.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$3.92232270276 \cdot 10^{-2}$	$2.62158635738 \cdot 10^{0}$	–
$2.75032662068 \cdot 10^{-2}$	$2.20047835635 \cdot 10^{0}$	0.49
$2.10724875981 \cdot 10^{-2}$	$1.78998293584 \cdot 10^{0}$	0.78
$1.45064713296 \cdot 10^{-2}$	$1.18528766445 \cdot 10^{0}$	1.10
$1.10905919595 \cdot 10^{-2}$	$8.13807966070 \cdot 10^{-1}$	1.40
$7.47435092752 \cdot 10^{-3}$	$4.25141543137 \cdot 10^{-1}$	1.65
$5.66883950836 \cdot 10^{-3}$	$2.70849483481 \cdot 10^{-1}$	1.63
$3.79972779725 \cdot 10^{-3}$	$1.42927670571 \cdot 10^{-1}$	1.60
$2.86144409572 \cdot 10^{-3}$	$9.53171279842 \cdot 10^{-2}$	1.43
$1.91558227513 \cdot 10^{-3}$	$5.80466382240 \cdot 10^{-2}$	1.24
$1.43902023413 \cdot 10^{-3}$	$4.26375827412 \cdot 10^{-2}$	1.08

5 Hybrid diffusive fluxes

"A numerical simulation is like sex. If it is good, then it is great. If it is bad, then it is still better than nothing."

— Denys Dutykh (1982 -)

Abstract. *In this chapter, the hybrid approach is validated for the diffusive fluxes using the Taylor-Green Vortex (TGV) test case. It is designed to test the accuracy of the direct numerical simulation of a flow transitioning to turbulence. It is tested on the same topological meshes but with different data structures, namely structured, unstructured and hybrid, to validate the hybrid approach.*

5.1 The Taylor-Green Vortex: the TGV test case

5.1.1 Objectives

As stated during the High-Order Workshop [2], the Direct Numerical Simulation (DNS) of the Taylor-Green Vortex at $Re = 1600$ is designed to test the accuracy of the DNS of the Navier-Stokes equations. It uses a three-dimensional periodic and transitional flow initialized as follow:

$$u_0 = V_\infty \sin\left(\frac{x}{L}\right)\cos\left(\frac{y}{L}\right)\cos\left(\frac{z}{L}\right), \tag{5.1}$$

$$v_0 = -V_\infty \cos\left(\frac{x}{L}\right)\sin\left(\frac{y}{L}\right)\cos\left(\frac{z}{L}\right), \tag{5.2}$$

$$w_0 = 0, \tag{5.3}$$

$$p_0 = p_\infty + \frac{\rho_\infty V_\infty^2}{16}\left[\cos\left(\frac{2x}{L}\right)+\cos\left(\frac{2y}{L}\right)\right]\left[\cos\left(\frac{2z}{L}\right)+2\right]. \tag{5.4}$$

The quantity u (resp. v, w and p) represents the velocity along the x-axis (resp. the velocity along the y-axis, the velocity along the z-axis and the pressure). All these quantities depend on space (x, y, z) and time t. The notation u_0 (resp. v_0, w_0 and p_0) stands for $u\,(x, y, z, t = 0)$ (resp. $v\,(x, y, z, t = 0)$, $w\,(x, y, z, t = 0)$ and $p\,(x, y, z, t = 0)$). This flow transitions to turbulence, with the creation of small scales, followed by a decay phase similar to decaying homogeneous turbulence (yet here non isotropic) according to [2].

5.1.2 Description

This test case is computed by discretizing the three-dimensional compressible Navier-Stokes equations although the flow is governed by the three-dimensional incompressible Navier-Stokes equations. The Mach number $M_\infty = V_\infty/\sqrt{\gamma R_{gas} T_\infty} = 0.1\ [-]$ is taken small enough to obtain solutions as close as possible to incompressible solutions. The fluid is assumed to be a perfect gas. The ratio of specific heats γ is equal to 1.4, the Prandtl number Pr is equal to 0.71 $[-]$, and the gas constant R_{gas} is equal to 287.058 $[J/kg/K]$. The initial temperature field is taken uniform $T = T_\infty = 300.0\ [K]$. The Reynolds number of the flow is defined as

$Re = \rho_\infty V_\infty L/\mu_\infty = 1600$ with $L = 1.0\ [m]$. The dynamic viscosity is computed using the Sutherland's law [62]:

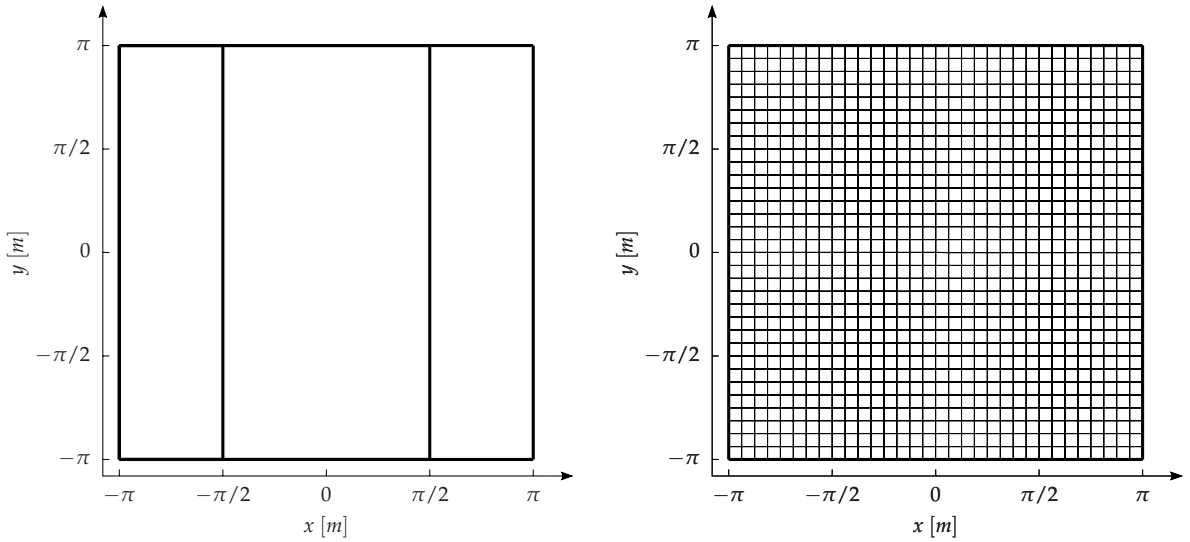
$$\mu_\infty = \mu_{ref} \left[\frac{T_\infty}{T_{ref}}\right]^{3/2} \frac{T_{ref} + S}{T_\infty + S}, \tag{5.5}$$

with the reference temperature $T_{ref} = 288.15\ [K]$, the reference dynamic viscosity at the reference temperature $\mu_{ref} = 1.78938 \cdot 10^{-5}\ [kg/m/s]$ and the Sutherland temperature $S = 110.4\ [K]$. From the Reynolds number expression, one obtains the density $\rho_\infty = 8.506 \cdot 10^{-4}\ [kg/m^3]$ and finally, thanks to the perfect gas law, one obtains the pressure $p_\infty = 73.254\ [Pa]$. The physical duration of the computation is based on the characteristic convective time $t_c = L/V_0$ and is set to $t_{final} = 20\, t_c$.

5.1.3 Meshes

The computational domain is cubic with $(x, y, z) \in [-\pi L, \pi L]^3$. Translational boundary conditions are imposed in the three directions. The computational domain is divided into three domains:

- the first domain is defined by $(x, y, z) \in [-\pi L \quad, -\pi L/2] \times [-\pi L, \pi L] \times [-\pi L, \pi L]$,
- the second domain is defined by $(x, y, z) \in [-\pi L/2, \quad \pi L/2] \times [-\pi L, \pi L] \times [-\pi L, \pi L]$,
- the third domain is defined by $(x, y, z) \in [\quad \pi L/2, \quad \pi L \quad] \times [-\pi L, \pi L] \times [-\pi L, \pi L]$.



(a) *Mesh topology. The computational domain is divided into three domains.*

(b) *SSS, UUU or SUS mesh. Only the data structure is changed.*

Fig. 5.1. $32 \times 32 \times 32$ *example mesh for the TGV test case. The grid interface for hybrid meshes is conforming but is numerically treated as a nonconforming grid interface.*

Each domain can be either Structured (S) or Unstructured (U). An Unstructured (U) domain is only composed of hexahedra. Three meshes have been tested with a total size of $256 \times 256 \times 256$:

- the SSS mesh composed of Structured meshes for the three domains,
- the UUU mesh composed of Unstructured meshes for the three domains,
- the SUS mesh composed of Structured meshes for the first and the third domains and an Unstructured mesh for the second domain.

For the hybrid mesh, the grid interface is topologically conforming but is numerically treated as a nonconforming grid interface. Since all the unstructured elements are hexahedra, the three meshes SSS, UUU and SUS are topologically identical as shown in Fig. 5.1. Only the data structure changes.

5.1.4 Numerics

The computations have been carried out on 256 processors. The explicit time integration scheme is the second-order six stage Runge-Kutta scheme RKo6s [44] where the Courant-Friedrichs-Lewy (CFL) number is equal to 0.7. Space is discretized by the second order Roe scheme [57, 58] without any limiter introduced in Sec. 2.4. Space discretization around the nonconforming grid interface is performed with the formalism introduced in Chap. 2.

5.1.5 Error computation

The main interest of this test case is to compute the kinetic energy dissipation rate ε in order to see if the numerical scheme is able to reproduce the temporal evolution of the kinetic energy E_k of a transitional flow. Roughly speaking, the kinetic energy is defined by

$$E_k(t) = \int_\Omega \frac{1}{2}\rho\left[u^2 + v^2 + w^2\right] \mathrm{d}\mathcal{V}, \tag{5.6}$$

where Ω is the computation domain and $\mathcal{V}$ the cell volume, and the kinetic energy dissipation rate ε by

$$\varepsilon(t) = -\frac{\mathrm{d}E_k}{\mathrm{d}t}(t). \tag{5.7}$$

The kinetic energy dissipation rate is computed by derivating the kinetic energy with a first-order upwind scheme. Then, the results are compared to a reference incompressible flow solution given by the the High-Order Workshop [2]. This solution was obtained using a dealiased pseudo-spectral code (developed at Université catholique de Louvain, UCL) for which, spatially, neither numerical dissipation nor numerical dispersion errors occur. The time-integration is performed using a low-storage 3-steps Runge-Kutta scheme [63], with a dimensionless timestep of 10^{-3}. These results have been grid-converged on a $512 \times 512 \times 512$ grid (a grid convergence study for a spectral discretization was also done by van Rees *et al.* in [64]). This means that all Fourier modes up to the 256th harmonic with respect to the domain length have been captured exactly (apart from the time integration error of the Runge-Kutta scheme).

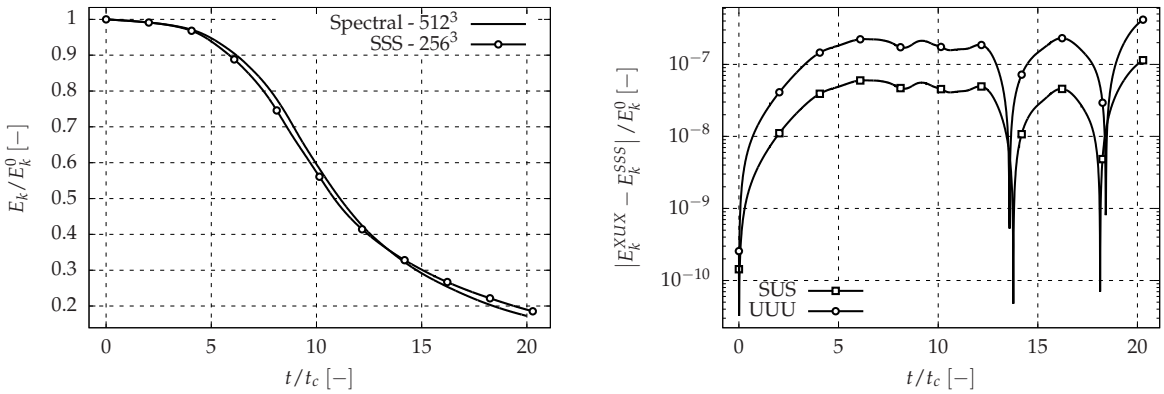
5.2 Results

The time is rendered dimensionless by $t_c = L/V_\infty$. The kinetic energy is rendered dimensionless by the initial kinetic energy $E_k^0 = E_k(t=0)$ computed by (5.8).

$$E_k^0 = \int_{-\pi L}^{+\pi L}\int_{-\pi L}^{+\pi L}\int_{-\pi L}^{+\pi L} \frac{1}{2}\rho_0\left[u_0^2 + v_0^2 + w_0^2\right] \mathrm{d}x\,\mathrm{d}y\,\mathrm{d}z = \pi^3 L^3 \rho_\infty V_\infty^2. \tag{5.8}$$

The kinetic energy dissipation rate is rendered dimensionless by $\varepsilon_c = E_k^0/t_c = \pi^3 L^2 \rho_\infty V_\infty^3$. In the High-Order Workshop [2], the kinetic energy dissipation rate is wrongly non-dimensionalized by $8\pi^3 L^2 \rho_\infty V_\infty^3 = 8\varepsilon_c$. Fig. 5.2a shows the comparison between the solution computed with the spectral code and the one computed with the SSS mesh. A good agreement is observed except that the kinetic energy dissipation rate is overestimated between $t/t_c \in [5, 10]$ as shown in Fig. 5.3 leading to a lower kinetic energy. This is not surprising since the second-order

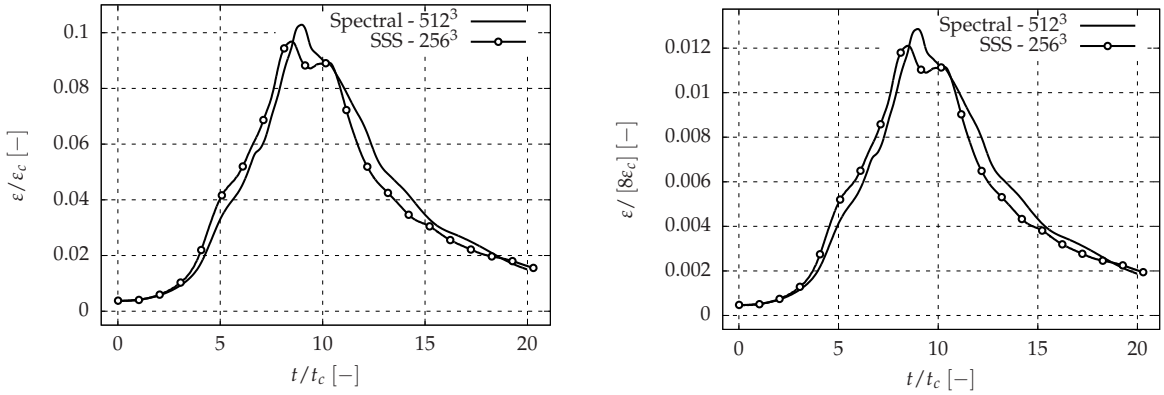
Roe scheme dissipates more than the spectral code. Fig. 5.2b shows the absolute value of the difference of the dimensionless kinetic energy between the one computed with the SUS or UUU mesh and the one computed with the SSS mesh. A perfect agreement is achieved which validates the hybrid and unstructured approaches. This is not surprising as well since the three meshes SSS, UUU and SUS are topologically identical as shown in Fig. 5.1. Only the data structure changes.



(a) *The dimensionless kinetic energy for the SSS mesh. Comparison with a spectral code.*

(b) *The kinetic energy difference for the SUS and UUU meshes with respect to the SSS mesh.*

Fig. 5.2. *The kinetic energy for three meshes. All give the same solution. Indeed, the three meshes SSS, UUU and SUS are topologically identical and only the data structure structure changes.*



(a) *The dimensionless kinetic energy dissipation rate for the SSS mesh. Comparison with a spectral code.*

(b) *The dimensionless kinetic energy dissipation rate for the SSS mesh. Comparison with a spectral code. The quantities are non-dimensionalized like in the High-Order Workshop [2].*

Fig. 5.3. *The kinetic energy dissipation rate.*

5.3 Conclusion

In both previous chapters, the hybrid approach was validated for both convective and diffusive fluxes. First of all, some general results were established to properly estimate the effective order of accuracy of a numerical scheme. For the convective fluxes, the validation has covered a large variety of meshes, including structured, unstructured and hybrid meshes, with the convection of an isentropic compressible vortex. Since it is an analytical solution of the Euler equations, it has been possible to compute the error to assess the hybrid approach. Using a first-order

Roe scheme, an effective first-order accuracy was always reached even with nonconforming grid interface. Using a second-order Roe scheme, an effective second-order accuracy was reached except with meshes composed of triangles or perturbed elements. This being said, note that the presence of nonconforming grid interfaces does not affect the effective order of accuracy. For diffusive fluxes, the validation was performed on the Taylor-Green Vortex (TGV) test case with the same topological meshes but with different data structures, namely structured, unstructured and hybrid. For applications where second-order schemes are broadly acceptable, typically dealing with RANS or URANS simulations, the hybrid approach was successfully validated and could be used for these kinds of applications. However, for more complex applications requiring high-order schemes, typically dealing with LES applications convecting unsteady flows over long distances or computing farfield aeroacoustics, the hybrid approach failed. Indeed, for structured zones, high-order schemes are available since the work of Fosso [6] but for unstructured zones, the situation is much more complex and will be the topic of the next part of this manuscript.

IV Extension to high-order unstructured FV schemes

6 High-order reconstruction on unstructured grids 93
6.1 The Roe solver
6.2 First-order extrapolation
6.3 Second-order extrapolation
6.4 Third-order extrapolation
6.5 Results for the third-order Roe scheme

7 The unstructured interface gradient 99
7.1 Finite Volume gradient review
7.2 The new interface gradient scheme

8 k-exact schemes 115
8.1 Bibliography
8.2 One-dimensional Finite Difference example on a non-uniform mesh
8.3 One-dimensional Finite Volume example on a non-uniform mesh
8.4 HPC efficiency

6 High-order reconstruction on unstructured grids

"Science, my lad, is made up of mistakes, but they are mistakes which it is useful to make, because they lead little by little to the truth."

— Jules Gabriel Verne (1828 - 1905)

Abstract. *In this chapter, a high-order extrapolation is developed to feed a Riemann solver for given left and right states. First of all, the first-order and the second-order extrapolations are developed. Based on the observation that the second-order extrapolation formulation is compacted by using the state and the gradient of this state for each cell, a third-order extrapolation is developed. Numerical tests are then carried out to validate this approach at least on the UUU/QF mesh.*

6.1 The Roe solver

Widely used in industrial computations, Roe's numerical flux is given by:

$$F_{m+1/2} = \frac{1}{2}\left[\mathcal{F}\left(W^L_{m+1/2}\right) + \mathcal{F}\left(W^R_{m+1/2}\right)\right] - \frac{1}{2}\left|\bar{\bar{A}}\left(W^L_{m+1/2}, W^R_{m+1/2}\right)\right|\left(W^R_{m+1/2} - W^L_{m+1/2}\right),$$

where $W^L_{m+1/2}$ and $W^R_{m+1/2}$ are the conservative variables interpolated on the left and right sides of the interface $m + 1/2$. The next section will describe how these variables can be interpolated. $\bar{\bar{A}}$ is the Roe's matrix which satisfies the following list of properties.

- $\bar{\bar{A}}\left(W, W\right) = A\left(W\right)$ where A is the Jacobian matrix.
- $\mathcal{F}\left(W^R\right) - \mathcal{F}\left(W^L\right) = \bar{\bar{A}}\left(W^L, W^R\right)\left(W^R - W^L\right)$.
- The eigenvectors of $\bar{\bar{A}}\left(W^L, W^R\right)$ are linearly independent.

6.2 First-order extrapolation

Analogously to the demonstration of Sec. 2.4 and considering the advection equation with a constant and positive transport velocity $c = 1\ [m/s]$, the Roe's numerical flux is then simply the left state and one obtains the flux balance $\delta W|_i$:

$$\delta W|_i = \Delta x \left.\frac{\partial W}{\partial x}\right|_i = W^L_{i+1/2} - W^L_{i-1/2}. \tag{6.1}$$

The Roe solver leads to a first-order solution when it is directly applied to mean quantities issued from the left and right sides of the interface. It is simply realized by taking $W^L_{m+1/2} = W_m$ and $W^R_{m+1/2} = W_{m+1}$. Applying the first-order Roe scheme, one obtains the flux balance $\delta W|^1_i$, where the superscript 1 means that the flux balance is discretized by a first-order extrapolation,

$$\delta W|^1_i = W_i - W_{i-1}. \tag{6.2}$$

Considering the Taylor series expansion of W_{i-1}, one obtains:

$$W_{i-1} = W_i - \Delta x \left.\frac{\partial W}{\partial x}\right|_i + \mathcal{O}\left(\Delta x^2\right). \tag{6.3}$$

Substituting (6.3) in (6.2), one proves that this scheme is first-order accurate:

$$\delta W|_i^1 = \Delta x \left.\frac{\partial W}{\partial x}\right|_i + \mathcal{O}\left(\Delta x^2\right). \tag{6.4}$$

Indeed, the first derivative is first-order accurate if the flux balance is second-order accurate. One order of convergence is lost when taking the derivative of an expression.

6.3 Second-order extrapolation

To obtain a second-order extrapolation, the left and the right states are extrapolated [65] by:

$$W_{m+1/2}^L = W_m + \frac{1}{2}\nabla \boldsymbol{W}_{\boldsymbol{m}}^{\boldsymbol{cc}} \cdot \boldsymbol{O_m O_{m+1}}, \tag{6.5}$$

and

$$W_{m+1/2}^R = W_{m+1} - \frac{1}{2}\nabla \boldsymbol{W}_{\boldsymbol{m+1}}^{\boldsymbol{cc}} \cdot \boldsymbol{O_m O_{m+1}}, \tag{6.6}$$

where O_m is the center of cell m and the cell centered gradient $\nabla \boldsymbol{W}_{\boldsymbol{m}}^{\boldsymbol{cc}}$ for the cell m is defined by:

$$\nabla \boldsymbol{W}_{\boldsymbol{m}}^{\boldsymbol{cc}} = \frac{W_{m+1} - W_{m-1}}{2\Delta x}. \tag{6.7}$$

Applying the Roe scheme, one obtains the flux balance $\delta W|_i^2$, where the superscript 2 means that the flux balance is discretized by a second-order extrapolation,

$$\delta W|_i^2 = \left[W_i + \frac{W_{i+1} - W_{i-1}}{4}\right] - \left[W_{i-1} + \frac{W_i - W_{i-2}}{4}\right], \tag{6.8}$$

$$= \frac{1}{4}W_{i-2} - \frac{5}{4}W_{i-1} + \frac{3}{4}W_i + \frac{1}{4}W_{i+1}. \tag{6.9}$$

Considering the Taylor series expansion of W_{i-2}, W_{i-1} and W_{i+1}, one obtains:

$$W_{i-2} = W_i - 2\Delta x \left.\frac{\partial W}{\partial x}\right|_i + 2\Delta x^2 \left.\frac{\partial^2 W}{\partial x^2}\right|_i + \mathcal{O}\left(\Delta x^3\right), \tag{6.10}$$

$$W_{i-1} = W_i - \Delta x \left.\frac{\partial W}{\partial x}\right|_i + \frac{\Delta x^2}{2} \left.\frac{\partial^2 W}{\partial x^2}\right|_i + \mathcal{O}\left(\Delta x^3\right), \tag{6.11}$$

$$W_{i+1} = W_i + \Delta x \left.\frac{\partial W}{\partial x}\right|_i + \frac{\Delta x^2}{2} \left.\frac{\partial^2 W}{\partial x^2}\right|_i + \mathcal{O}\left(\Delta x^3\right). \tag{6.12}$$

Substituting (6.10), (6.11) and (6.12) in (6.8), one proves that this scheme is second-order accurate:

$$\delta W|_i^2 = \Delta x \left.\frac{\partial W}{\partial x}\right|_i + \mathcal{O}\left(\Delta x^3\right). \tag{6.13}$$

6.4 Third-order extrapolation

To obtain a third-order extrapolation, the left and the right states are extrapolated by:

$$W^L_{m+1/2} = W_m + \frac{1}{2}\left[(1-\beta)\nabla \boldsymbol{W}^{cc}_{\boldsymbol{m}} + \beta \nabla \boldsymbol{W}^{ic}_{\boldsymbol{m}}\right] \cdot \boldsymbol{O_m O_{m+1}}, \tag{6.14}$$

and

$$W^R_{m+1/2} = W_{m+1} - \frac{1}{2}\left[(1-\beta)\nabla \boldsymbol{W}^{cc}_{\boldsymbol{m+1}} + \beta \nabla \boldsymbol{W}^{ic}_{\boldsymbol{m}}\right] \cdot \boldsymbol{O_m O_{m+1}}, \tag{6.15}$$

where the interface centered gradient $\nabla \boldsymbol{W}^{ic}_{\boldsymbol{m}}$ for the cell m is defined by:

$$\nabla \boldsymbol{W}^{ic}_{\boldsymbol{m}} = \frac{W_{m+1} - W_m}{\Delta x}. \tag{6.16}$$

The parameter β is free. If $\beta = 0$, the standard second-order extrapolation defined in the previous section is recovered and $\beta = 1/3$ gives a third-order extrapolation. Indeed, applying this extrapolation scheme, one obtains the flux balance $\delta W|_i^3$, where the superscript 3 means that the flux balance is discretized by a third-order extrapolation,

$$\begin{aligned} \delta W|_i^3 &= \left[W_i + (1-\beta)\frac{W_{i+1} - W_{i-1}}{4} + \beta \frac{W_{i+1} - W_i}{2}\right] \qquad (6.17) \\ &\quad - \left[W_{i-1} + (1-\beta)\frac{W_i - W_{i-2}}{4} + \beta \frac{W_i - W_{i-1}}{2}\right] \\ &= \frac{1}{6}W_{i-2} - W_{i-1} + \frac{1}{2}W_i + \frac{1}{3}W_{i+1}. \qquad (6.18) \end{aligned}$$

Considering the Taylor series expansion of W_{i-2}, W_{i-1} and W_{i+1}, one obtains:

$$W_{i-2} = W_i - 2\Delta x \left.\frac{\partial W}{\partial x}\right|_i + 2\Delta x^2 \left.\frac{\partial^2 W}{\partial x^2}\right|_i - \frac{4\Delta x^3}{3}\left.\frac{\partial^3 W}{\partial x^3}\right|_i + \mathcal{O}\left(\Delta x^4\right), \tag{6.19}$$

$$W_{i-1} = W_i - \Delta x \left.\frac{\partial W}{\partial x}\right|_i + \frac{\Delta x^2}{2}\left.\frac{\partial^2 W}{\partial x^2}\right|_i - \frac{\Delta x^3}{6}\left.\frac{\partial^3 W}{\partial x^3}\right|_i + \mathcal{O}\left(\Delta x^4\right), \tag{6.20}$$

$$W_{i+1} = W_i + \Delta x \left.\frac{\partial W}{\partial x}\right|_i + \frac{\Delta x^2}{2}\left.\frac{\partial^2 W}{\partial x^2}\right|_i + \frac{\Delta x^3}{6}\left.\frac{\partial^3 W}{\partial x^3}\right|_i + \mathcal{O}\left(\Delta x^4\right). \tag{6.21}$$

Substituting (6.19), (6.20) and (6.21) in (6.17), one proves that this scheme is third-order accurate:

$$\delta W|_i^3 = \Delta x \left.\frac{\partial W}{\partial x}\right|_i + \mathcal{O}\left(\Delta x^4\right). \tag{6.22}$$

6.5 Results for the third-order Roe scheme

The numerical setup to test the third-order Roe scheme is defined in Sec. 4.2. The aim of this section is to test the unstructured formulation of this Roe scheme. This is why only four unstructured mesh types were tested among those described in Sec. 4.2: the UUU/QF, UUU/QT, UUU/TF and UUU/TT meshes. The results for the third-order Roe scheme are given from Tab. 6.1 to Tab. 6.4. Several interesting conclusions were reached.

- The UUU/QF mesh gives third-order accurate results. This was naturally expected since the third-order extrapolation is built to be third-order accurate on this type of mesh.
- However, the UUU/QT, UUU/TF and UUU/TT meshes only give first-order accurate results. Indeed, the third-order extrapolation does not take into account the local metrics since the coefficients are constant and not dependent on the local metrics. On general unstructured grids, this scheme is absolutely not efficient.

- Looking at (6.14) and (6.15), it seems to be unavoidable to find a better evaluation of the cell centered and interface centered gradients. Including the metrics into the evaluation of these gradients could improve the accuracy of the extrapolation: this is the topic of the next chapter.

Tab. 6.1. *Mesh convergence analysis on the UUU/QF mesh for the third-order Roe scheme.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$6.25000000000 \cdot 10^{-2}$	$2.71283294809 \cdot 10^{0}$	–
$4.16666666667 \cdot 10^{-2}$	$2.21773714423 \cdot 10^{0}$	0.50
$3.12500000000 \cdot 10^{-2}$	$1.79068993330 \cdot 10^{0}$	0.74
$2.08333333333 \cdot 10^{-2}$	$1.10768881137 \cdot 10^{0}$	1.18
$1.56250000000 \cdot 10^{-2}$	$6.75289847793 \cdot 10^{-1}$	1.72
$1.04166666667 \cdot 10^{-2}$	$2.72056704277 \cdot 10^{-1}$	2.24
$7.81250000000 \cdot 10^{-3}$	$1.27660698051 \cdot 10^{-1}$	2.63
$5.20833333333 \cdot 10^{-3}$	$4.03960774836 \cdot 10^{-2}$	2.84
$3.90625000000 \cdot 10^{-3}$	$1.73612318685 \cdot 10^{-2}$	2.94
$2.60416666667 \cdot 10^{-3}$	$5.20401698177 \cdot 10^{-3}$	2.97
$1.95312500000 \cdot 10^{-3}$	$2.20601903504 \cdot 10^{-3}$	2.98

Tab. 6.2. *Mesh convergence analysis on the UUU/QT mesh for the third-order Roe scheme.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$6.25000000000 \cdot 10^{-2}$	$2.79403573837 \cdot 10^{0}$	–
$4.16666666667 \cdot 10^{-2}$	$2.26480472572 \cdot 10^{0}$	0.52
$3.12500000000 \cdot 10^{-2}$	$1.85311543384 \cdot 10^{0}$	0.70
$2.08333333333 \cdot 10^{-2}$	$1.17997017257 \cdot 10^{0}$	1.11
$1.56250000000 \cdot 10^{-2}$	$7.58129574205 \cdot 10^{-1}$	1.54
$1.04166666667 \cdot 10^{-2}$	$3.53798634672 \cdot 10^{-1}$	1.88
$7.81250000000 \cdot 10^{-3}$	$2.07508062465 \cdot 10^{-1}$	1.85
$5.20833333333 \cdot 10^{-3}$	$9.59772645293 \cdot 10^{-2}$	1.90
$3.90625000000 \cdot 10^{-3}$	$6.16577070161 \cdot 10^{-2}$	1.54
$2.60416666667 \cdot 10^{-3}$	$3.53783525826 \cdot 10^{-2}$	1.37
$1.95312500000 \cdot 10^{-3}$	$2.55318144096 \cdot 10^{-2}$	1.13

Tab. 6.3. *Mesh convergence analysis on the UUU/TF mesh for the third-order Roe scheme.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$3.84615384615 \cdot 10^{-2}$	$2.46382205125 \cdot 10^{0}$	–
$2.54493299280 \cdot 10^{-2}$	$1.86187018517 \cdot 10^{0}$	0.68
$1.94110729049 \cdot 10^{-2}$	$1.42262917879 \cdot 10^{0}$	0.99
$1.26836572358 \cdot 10^{-2}$	$8.45341763240 \cdot 10^{-1}$	1.22
$9.63768687807 \cdot 10^{-3}$	$5.39963756698 \cdot 10^{-1}$	1.63
$6.37965835292 \cdot 10^{-3}$	$3.15023367407 \cdot 10^{-1}$	1.31
$4.81236634590 \cdot 10^{-3}$	$2.17373336372 \cdot 10^{-1}$	1.32
$3.21130312701 \cdot 10^{-3}$	$1.25827094196 \cdot 10^{-1}$	1.35
$2.40105900452 \cdot 10^{-3}$	$9.42775595367 \cdot 10^{-2}$	0.99
$1.60253117779 \cdot 10^{-3}$	$6.07787443992 \cdot 10^{-2}$	1.09
$1.20086012388 \cdot 10^{-3}$	$4.52585499353 \cdot 10^{-2}$	1.02

Tab. 6.4. *Mesh convergence analysis on the UUU/TT mesh for the third-order Roe scheme.*

$h\ [-]$	Error $\varepsilon_V\ [m \cdot s^{-1}]$	Order $[m \cdot s^{-1}]$
$3.84615384615 \cdot 10^{-2}$	$6.32321543693 \cdot 10^{0}$	–
$2.54493299280 \cdot 10^{-2}$	$1.88118208562 \cdot 10^{0}$	2.94
$1.94110729049 \cdot 10^{-2}$	$1.50097781878 \cdot 10^{0}$	0.83
$1.26836572358 \cdot 10^{-2}$	$9.42932928481 \cdot 10^{-1}$	1.09
$9.63500241020 \cdot 10^{-3}$	$6.51643183789 \cdot 10^{-1}$	1.34
$6.37732276494 \cdot 10^{-3}$	$3.94538795755 \cdot 10^{-1}$	1.22
$4.81236634590 \cdot 10^{-3}$	$2.96214715129 \cdot 10^{-1}$	1.02
$3.21130312701 \cdot 10^{-3}$	$1.90014440499 \cdot 10^{-1}$	1.10
$2.40035335802 \cdot 10^{-3}$	$1.46879118265 \cdot 10^{-1}$	0.88
$1.60253117779 \cdot 10^{-3}$	$9.59953279797 \cdot 10^{-2}$	1.05
$1.20086012388 \cdot 10^{-3}$	$7.32851792311 \cdot 10^{-2}$	0.94

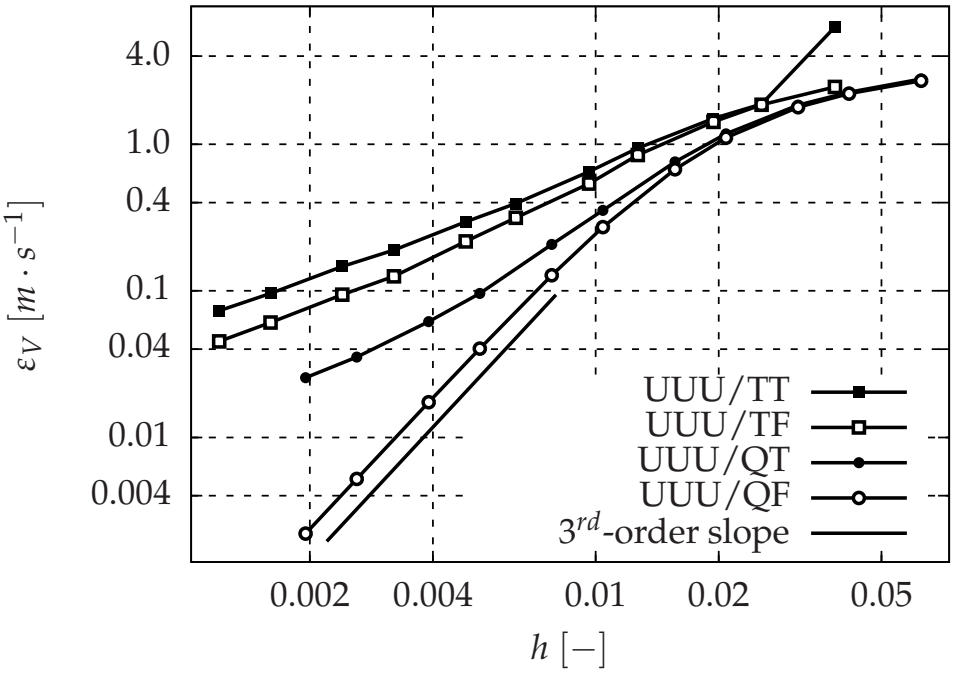


Fig. 6.1. *Mesh convergence for the third-order Roe scheme.*

7 The unstructured interface gradient

"If the facts don't fit the theory, change the facts."

— Albert Einstein (1879 - 1955)

Abstract. *In this chapter, a review of Finite Volume gradient computation is made. Several methods are reviewed among which the standard and enhanced Green-Gauss formulations, the Least-Squares Method and the diamond cell approach. Then, a new formulation for Unstructured Interface Gradient (UIG) computation is developed and leads to a second-order accurate scheme on general unstructured grids. This gradient can be used to compute the Laplacian operator which is first-order accurate on general unstructured grids. The numerical tests to validate this gradient formulation have not been performed yet, but upon achievement they will constitute the gist of paper in preparation for Journal of Scientific Computing [66].*

7.1 Finite Volume gradient review

Before reviewing the bibliography, let us introduce the general form of standard Partial Differential Equations (PDE) with a diffusion term:

$$\frac{\partial W}{\partial t} + \nabla \cdot \mathcal{D}(W, \nabla \boldsymbol{W}) = 0, \tag{7.1}$$

where W stands for the unknown and $\mathcal{D}$ is the general form of any operator involving both state variable W and its gradient $\nabla \boldsymbol{W}$. Since we are interested in solving (7.1) with a cell-centered Finite Volume approach, the computational domain is first split into several non overlapping volumes denoted Ω_i. The intersection between two volumes Ω_i and Ω_j is either one of the mesh interfaces denoted $F_{(i,j)}$ or empty $\varnothing$. For a volume Ω_i, the list of its direct neighbor indexes is denoted $\mathcal{F}_i$: for any index $j \in \mathcal{F}_i$, the face $F_{(i,j)}$ is a true mesh face that separates cells Ω_i and Ω_j. Then, integrating (7.1) over Ω_i and using Gauss formula, it comes:

$$\frac{\mathrm{d}}{\mathrm{d}t}(\mathcal{V}_i \overline{W}_i) + \sum_{j \in \mathcal{F}_i} \int_{F_{(i,j)}} \mathcal{D}(W, \nabla \boldsymbol{W}) \boldsymbol{n}_{(i,j)} \, \mathrm{d}\mathcal{S} = 0, \tag{7.2}$$

where $\mathcal{V}_i$ is the volume of cell Ω_i and $\overline{W}_i$ is the mean value of W over cell Ω_i. (7.2) shows that starting from the averaged quantity over any mesh cell, advancing the equation in time is possible once integrals on mesh interfaces are known, taking care that the unit normal vector $\boldsymbol{n}_{(i,j)}$ is directed outward (from Ω_i to Ω_j). It is therefore mandatory to define an interface gradient to compute any diffusion term of a PDE.

For industrial Computational Fluid Dynamics solvers, the time integration is only possible once a convection scheme is introduced. The standard way to deal with any hyperbolic equation follows information transport along characteristic curves and an exact or approximated Riemann solver is introduced. For a second order of accuracy, Van Leer introduced the notion of MUSCL reconstruction [65]: the flux is not computed directly using left and right states but using reconstructed quantities obtained as linear extrapolation of the cell-centered quantities.

Such an extrapolation involves a cell-centered gradient which must be computed first by taking into account the surrounding cells. In this section, several methods to define an interface gradient are introduced and it is shown that most of the diffusion schemes build the interface gradient using the cell-centered gradients as input.

Before entering into the details, we wish to remind one key point regarding the diffusion operator. The diffusion operator, as the convection operator, must satisfy the discrete maximum principle. Roughly speaking, the Discrete Maximum Principle (DMP) of the diffusion operator and the shape of its associated matrix on the computational domain are strongly linked. In practice, it is well-known that if the associated matrix is an $M-$Matrix, then all the desired properties hold. We recall that an $M-$matrix is a $Z-$matrix with eigenvalues whose real parts are positive and a $Z-$matrix is such that off-diagonal entries are less than or equal to zero. Even it won't be recalled in the following, the definition of a second-order discretization of the Laplacian with the discrete maximum property is still an open problem for general unstructured grids. Only few second-order diffusion schemes are monotonic on special grid shapes and the property does not hold in general. In [67, 68], a general form for diffusion schemes dedicated to 2D and 3D unstructured meshes is proposed, with a structure such that the Local Maximum Property (LMP) is guaranteed. However, most of diffusion schemes currently applied in CFD are not LMP on general unstructured grids. Finally, following the work of Le Potier and co-authors, the LMP can be recovered for consistent schemes [68, 69, 70, 71].

Remark: In the present bibliographic review, only schemes applied to Finite Volume are considered. A complete analysis may also include schemes for spectral discontinuous methods such as the Discontinuous Galerkin, Spectral Difference, Spectral Volume, Flux Reconstruction techniques.

7.1.1 Techniques based on Green-Gauss formulation on mesh elements

The standard Green-Gauss method (identified by GG in the following) uses the Green-Gauss formulation to compute the cell-centered averaged gradient using states at the interface. It is generally defined as an arithmetic average of quantities from both sides, using either a pure centered approximation or a weighted formulation taking into account cell volume. Following notations introduced in (7.2), the formulation becomes:

$$\int_{\Omega_i} \nabla \boldsymbol{W} \, \mathrm{d}\mathcal{V} = \sum_{j \in \mathcal{F}_i} \int_{F_{(i,j)}} W \boldsymbol{n}_{(i,j)} \, \mathrm{d}\mathcal{S}, \tag{7.3}$$

and (7.3) can be discretized as:

$$\mathcal{V}_i \overline{\nabla \boldsymbol{W}}_i = \sum_{j \in \mathcal{F}_i} \frac{\mathcal{V}_j \overline{W}_i + \mathcal{V}_i \overline{W}_j}{\mathcal{V}_i + \mathcal{V}_j} \mathcal{S}_{(i,j)} \boldsymbol{n}_{(i,j)}, \tag{7.4}$$

where $\mathcal{S}_{(i,j)}$ represents the area of the face $F_{(i,j)}$.

Standard Green-Gauss formulation.

The interface gradient is then defined as an arithmetic average of quantities of both sides, using either a pure centered approximation or a weighted formulation taking into account opposite element volume:

$$\nabla \boldsymbol{W}|_{F_{(i,j)}} = \frac{\mathcal{V}_j \overline{\nabla \boldsymbol{W}}_i + \mathcal{V}_i \overline{\nabla \boldsymbol{W}}_j}{\mathcal{V}_i + \mathcal{V}_j}. \tag{7.5}$$

The implementation is straightforward in a standard CFD code using a face-based data structure: the computation of the interface gradient only needs two loops on all mesh faces. For

the pure centered approach, the interface gradient involves two centered averages and such a scheme does not damp high frequency modes. Moreover, the scheme leads to an odd / even decoupling. Such a scheme was recently analyzed for unstructured grids composed of quadrangles, triangles and polygons [72].

The scheme defined by (7.4) and (7.5) is shown to be conservative and consistent on regular elements (equilateral triangles, squares or rectangles of a given size). It is stable and monotonic for equilateral triangles, squares, parallelograms and regular rectangles. Moreover, the method is found to be second order accurate for squares and equilateral triangles but since the schemes does not involve any metrics associated with cell shapes, its accuracy is reduced on irregular elements.

An alternative to (7.5) consists in averaging the gradients by a weighted formulation taking into account local element volume:

$$\nabla \boldsymbol{W}|_{F_{(i,j)}} = \frac{\mathcal{V}_i \overline{\nabla \boldsymbol{W}}_i + \mathcal{V}_j \overline{\nabla \boldsymbol{W}}_j}{\mathcal{V}_i + \mathcal{V}_j}. \tag{7.6}$$

This consists in defining the interface gradient as the mean gradient computed on a "composite" cell composed of the two cells that share the interface. Using standard Taylor expansion rules, basic arithmetic rules can show that (7.6) is not consistent with the interface gradient in 1D.

Enhanced Green-Gauss formulation.

In the context of vertex-centered Finite Volume approximation on dual grids, Galle [73] proposed to compute the gradients at mesh nodes by a GG formulation on dual cells as shown in Fig. 7.1 built around mesh nodes using cell centers and face centers in 2D (and mid-edges in 3D). The gradient on the interface located between nodes i and j is then the average of the gradients in nodes i and j.

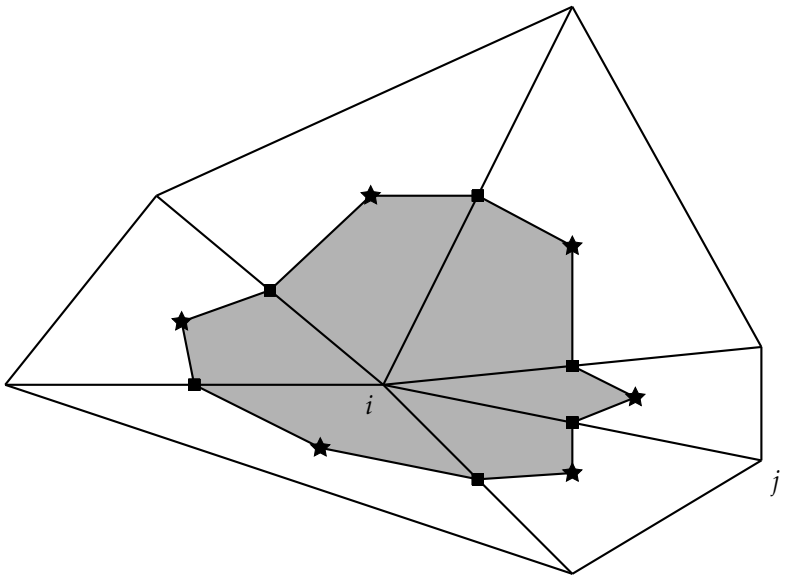


Fig. 7.1. *Definition of the dual volume around node i including face centers (squares) and cell centers (stars). In 2D, face center and mid-edges are located at the same place.*

In 1997, Crumpton [74] introduced a correction called CGG for Corrected Green-Gauss to improve the stability for meshes with a large cell aspect ratio. The correction consists in replacing the GG interface gradient by a centered finite difference gradient in the direction of the mesh edge ij. Such a correction can be easily introduced in the context of cell-centered Finite Volume approximation. It involves a correction of the interface gradient in the direction of the cell centers for both control volumes that share the considered face. The new interface

gradient $\overline{\nabla \boldsymbol{W}}\big|_{F_{(i,j)}}$ is now:

$$\overline{\nabla \boldsymbol{W}}\big|_{F_{(i,j)}} = \nabla \boldsymbol{W}|_{F_{(i,j)}} - \left[\nabla \boldsymbol{W}|_{F_{(i,j)}} \cdot \boldsymbol{C_i C_j} - \left(\overline{W_j} - \overline{W_i}\right) \right] \frac{\boldsymbol{C_i C_j}}{\left\| \boldsymbol{C_i C_j} \right\|^2}, \tag{7.7}$$

where C_i and C_j are the cell centers of Ω_i and Ω_j respectively, and the other notations are the same as in (7.3).

The CGG method correctly damps high frequencies and the CPU overhead is negligible compared with the standard GG technique. Moreover, the technique introduces a new coupling which prevents checkerboard instability found in the GG formulation. This scheme is shown to be conservative, consistent, monotonic and second-order accurate on rectangular meshes only.

An alternative to the CGG method was proposed by Cary, Dorgan and Mani [75] in order to correct the lack of accuracy of the cell-centered gradient for the drag prediction. They proposed to use the cell centered quantities to define a gradient in the direction of the cell centers and to introduce the cell-center value of a cell that shares a vertex with the considered face to account for a contribution in another direction. In such a way, linear solution can be recovered exactly and formally, the proposed scheme is second-order accurate. Unfortunately, such a method introduces a minimal amount of dissipation and the authors explain that the procedure was found to lack robust stability properties on practical arbitrary meshes. For this reason, it will not be considered in the following.

7.1.2 Least-Squares Method for cell-centered gradient

The second technique for computing the cell-centered gradient is based on the Least-Squares Method. Such a gradient [76] was first introduced to tackle the second order of accuracy for convection terms by means of linear extrapolation and slope limiter procedure [65] to satisfy the Total Variation-Diminishing (TVD) property. The Least-Squares Method was analyzed for the advection-diffusion equation by Ollivier-Gooch *et al.* in 2002 [77] and revisited in 2003 by Mavriplis [78].

A recent exhaustive analysis for the diffusion scheme was performed by Jalali, Sharbatdar and Ollivier-Gooch in 2014 [79]. The Least-Squares reconstruction was also analysed by Sozer, Brehm and Kiris [72] for the definition of the second-order hyperbolic flux. The Least-Squares Method was improved very recently by Sejekan *et al.* [80]. Here, the goal is to perform the Least-Squares Method in the H_1 norm instead of the standard L^2 norm. Such a change enables to control at the same time the reconstruction of the field and of the gradient and the resulting gradients are second-order accurate.

The Least-Squares cell-centered reconstruction is at the basis of the composite gradient introduced in [75] for computing the diffusion flux on the mesh interfaces. Cary *et al.* [75] introduced the interface gradient as the averaged gradient on the composite cell (as in (7.6)) consisting of the two cells that share the face which the gradient is desired. An additional term proportional to the face area and reconstructed discontinuity (generally called jump term) is introduced in the direction normal to the face. In contrary with (7.7) where the correction is based on the cell-centered quantities, the procedure proposed by Cary needs a correction term propositional to the jump of the extrapolated quantities at the face center.

The procedure to define the cell-centered gradient is simple. First, a Taylor expansion of the solution is performed locally around a cell Ω_i. It is of course required that the integral of the Taylor expansion over Ω_i recovers the average value $\overline{W}_i$, while the integral of the Taylor expansion over the adjacent cell Ω_j of Ω_i will recover $\overline{W}_j$.

Finally, the interface gradient is estimated using a simple averaging procedure or using an advanced treatment that depends on local metrics [79]. A first enhanced approach proposed by Shima *et al.* consists in blending the Green-Gauss approach and the Least-Squares reconstruction [81] for second-order convection scheme on polyhedral unstructured grids. Two different

improvements were proposed recently by Sejekan and Ollivier-Gooch [80]. The first modification concerns the definition of the local reference point inside the cell. Here, the authors propose to choose the containment center, the center of the smallest circle that can enclose a given mesh triangle. The introduction of an interface jump term is the second improvement. This jump term is computed at any Gauss quadrature point on the mesh interface and it is based on the standard linear extrapolation of unknowns plus a second-order correction based on the Hessian matrix in the direction normal to the interface.

Currently, the Least-Squares reconstruction approach is considered as the standard way to compute gradients. It requires the pre-computation of the coefficients of the pseudo-inverse matrix [82, 83] and these coefficients only depend on local metrics. In practice, the accuracy of the Least-Squares formulation strongly depends on the local accuracy of the polynomial reconstruction. In general, the considered stencil includes the direct neighbors and extra cells but here, we decided to consider the simplest stencil, involving only the direct neighbors.

7.1.3 Quasi-Green and Quasi-Least-Squares methods

Starting from the definition of the Green-Gauss and Least-Squares methods, it is clear that the cell-centered gradient can be seen as a linear combination of the averaged quantities over the chosen stencil. By performing Taylor expansion, taking care of local metrics and on the definition of the cell-centered quantities, it is possible to perform a local accuracy analysis. Both Green-Gauss and Least-Squares methods are found to be not consistent for any general unstructured grid. The principle of Quasi-Green and Quasi-Least-Squares methods is to correct the computed gradients in order to increase their accuracy [84, 85]. Such a technique is applied successfully for the $k-$exact reconstruction technique introduced in Chap. 8 and is derived in [86, 87] in the context of the successive correction algorithm. With the later, it is possible to build the Taylor expansion derivative per derivative using a compact stencil (the current cell and its direct neighbors only).

7.1.4 Diamond cell approaches

The third class of techniques to compute interface gradients follows the diamond path approach introduced by Coirier [88] for Cartesian grids and the cell-centered Finite Volume approximation. The principle of the proposed approach is to define a dual cell around any mesh interface and to compute an averaged gradient on it using the Green-Gauss rule as shown in Fig. 7.2. The dual cell is based on both face nodes and adjacent cell centers. The dual cell gradient is then chosen as interface gradient for the diffusion flux on the face. Following the

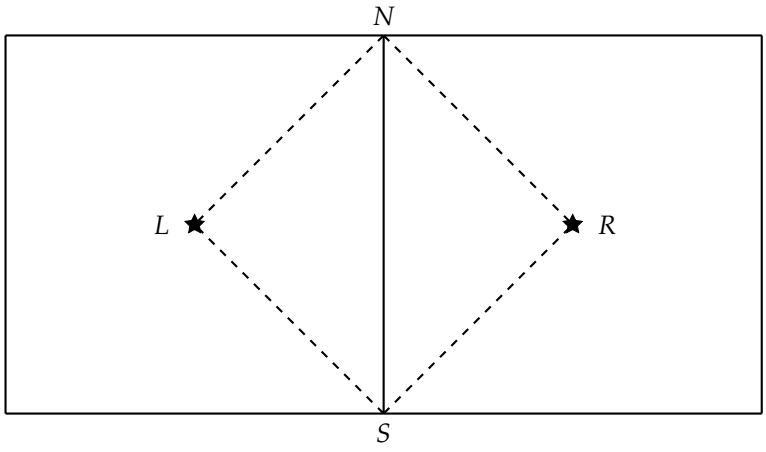


Fig. 7.2. *Definition of the dual volume around the face SN. Cell centers are represented by the black star.*

definition of the four nodes in Fig. 7.2 and introducing the normal vector $\boldsymbol{n}_{NL}$ (resp. $\boldsymbol{n}_{LS}$, $\boldsymbol{n}_{SR}$,

$\boldsymbol{n}_{RN}$) normal to the face NL (resp. LS, SR, RN) and directed toward the outside of the diamond cell, the final expression of the gradient over the diamond cell is simply:

$$\mathcal{V}_d \nabla \boldsymbol{W} = \frac{W_N + W_L}{2}\boldsymbol{n}_{NL} + \frac{W_L + W_S}{2}\boldsymbol{n}_{LS} + \frac{W_S + W_R}{2}\boldsymbol{n}_{SR} + \frac{W_R + W_N}{2}\boldsymbol{n}_{RN}, \tag{7.8}$$

with $\mathcal{V}_d$ the volume of the diamond cell. A diamond cell technique requires the definition of the solution at mesh nodes S and N and there are several ways to compute them.

Averaging procedure.

In the original approach proposed by Coirier [88] for Cartesian grids, the unknowns at the mesh node M are defined by an arithmetic average with constant weights of contributions from all cells built with the node M. For a Cartesian grid, the reconstruction stencil is compact and the number of elements sharing a node is known *a priori*. However, such a reconstruction presents two drawbacks for unstructured grids. First, all cells are assumed to have the same size, which is not the case on general unstructured grids. Moreover, the number of cells sharing a given node is assumed constant, due to the general form of the Cartesian grids. Extensions were proposed by Coirier [88] to correct the Cartesian grid limitations.

Least-Squares reconstruction.

In Coudière *et al.* [89], the reconstruction at a node is performed with a Least-Squares Method and involves any cell based on the considered node. For the sake of clarity, the Least-Squares formulation is described below in two-dimensional case. Let f_j be a linear function defined on $\mathcal{V}_j$, the set of mesh elements that share node j. Due to its definition, f_j takes the following form:

$$f_j(x, y) = a + bx + cy, \tag{7.9}$$

where a, b and c are the three unknowns. The Least-Squares interpolation consists in choosing the coefficients a, b and c such that the distance d_j between the reconstructed value using f_j and the averaged cell-centered values in the cells of $\mathcal{V}_j$ is minimized:

$$d_j = \sum_{\Omega_i \in \mathcal{V}_j} \left(\overline{W_i} - f_j(x_i, y_i) \right)^2, \tag{7.10}$$

where (x_i, y_i) represents the coordinates of the cell center of Ω_i. Moreover, since the gradient of any function is zero when the function is at its minimum, the procedure consists in computing the derivative of d_j as a function of a, b and c. It leads to a 3×3 linear system of equations. The system can be inverted upon condition on the cell center positions. For three different cell centers C_i, C_k and C_l of cells in $\mathcal{V}_j$, the vectors $\boldsymbol{C_iC_k}$ and $\boldsymbol{C_iC_l}$ must not be linearly connected.

Choice of the local stencil for reconstruction.

Le Potier [90] proposed a new approach to simplify the interpolation process. Instead of solving the whole Least-Squares system involving all cells surrounding the node, Le Potier proposed to take into account the minimum number of cells, closest to the mesh node on which interpolation must be performed. This enables the definition of the nodal quantity from a simple arithmetic averaging procedure using weights chosen as the inverse of the distance between the mesh node and the cell center.

Enhancement of the method.

An extension of the method was proposed. For instance, Hermeline [91], Domelevo and Omnès [92] proposed to introduce the same kind of dual grid as Le Potier and to solve the diffusion problem on both grids. Of course, the procedure establishes a coupling between the unknown on the original and dual grids. Such a procedure is not retained for the current analysis since the computational cost associated with such a technique is by nature larger than when only one problem is solved.

7.1.5 Multi Flux Point Approximation

The Multi Point Flux Approximation (MPFA) schemes assume that the solution is piecewise linear in some sub-cells around each vertex. They introduce additional edge unknowns around each vertex, and express the linear variation of the solution to compute gradients and thus fluxes in these sub-cells. The edge unknowns are eliminated by writing continuity equations for the solution and conservativity equations for the fluxes. Finally, the fluxes are consistent, conservative and expressed in terms of cell unknowns only.

A general review of MPFA schemes is proposed by Droniou [93] and many adaptations have been proposed in the past. A complete review is out of the scope of the current thesis but an interested reader can carefully read the first papers [94, 95, 96] and some recent extensions [97, 98, 99, 100, 101, 102]. Among them, the scheme of Droniou and Le Potier [100] is easy to implement, positive and consistent. It preserves the DMP. It was successfully applied to radiative hydrodynamics [103, 104].

Let us consider the notations introduced in Fig. 7.3. K and L define the cells that share the face i on which the flux needs to be estimated. The point $M_{K,i}$ (resp. $M_{L,i}$) is located on the line normal to the interface and issued from the cell center of cell K (resp. L). The values of the unknowns in $M_{K,i}$ or $M_{L,i}$ are obtained as a convex combination of the cell-centered unknowns with weights precomputed using geometric properties. A simple Finite Difference approximation using the unknowns in K and $M_{K,i}$ and in L and $M_{L,i}$ is a good approximation of the gradient on these lines. In other words, two upwind approximations of the flux can be defined. The final flux is then written as a combination of both fluxes using weights based on the local unknowns. This is the reason why the approach is non-linear. Such a scheme preserves the Discrete Maximum Property but is first-order accurate.

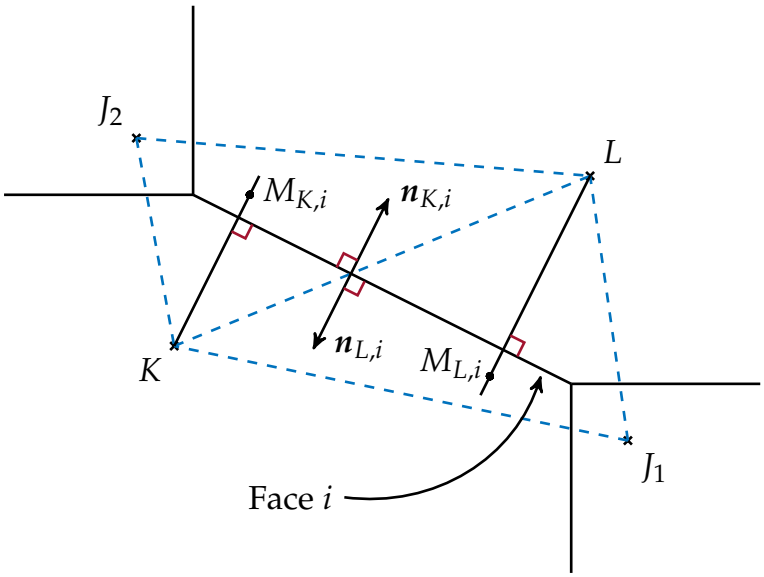


Fig. 7.3. *Definition of $M_{K,i}$ and $M_{L,i}$ points. The rays $[KM_{K,i})$ and $[LM_{L,i})$ are orthogonal to the face i. $M_{K,i}$ (resp. $M_{L,i}$) belongs to the triangle KLJ_2 (resp. KLJ_1).*

7.1.6 Diffusion on curvilinear meshes

It is easily demonstrated that standard diffusion scheme loose their accuracy on curvilinear meshes. Two techniques are proposed, the first one for CFD and the later one for magnetic fields.

The Curvilinear Gradient Method (CGM).
An alternative way to compute the gradients follows a simple idea: a gradient can be defined accurately on structured grids by simple relations built using finite differences. With the curvilinear gradient method [72], the unstructured (polygonal) grid is mapped to a uniform Cartesian mesh and the gradient is computed using a structured-grid approach on this Cartesian grid.

For this mapping, a four-point stencil (six-point stencil in 3D) must be selected from the available cell neighbors (Fig. 7.4). The selection procedure evaluates every possible combination of stencil point pairs to pick the curvilinear directions ξ and η (Fig. 7.4). To do so, the mapping $\mathcal{M}$ between the original mesh and the Cartesian grid is first computed and the Jacobian matrix $\mathcal{J}$ of the mapping is computed. The largest determinant of the Jacobian gives the best choice in the computational domain. In the reference domain, a central differencing formula gives:

$$\begin{cases} \dfrac{\partial \varphi}{\partial \xi} = \varphi_{i+1} - \varphi_{i-1} \\ \dfrac{\partial \varphi}{\partial \eta} = \varphi_{j+1} - \varphi_{j-1}. \end{cases} \tag{7.11}$$

Finally, the gradient in the computational domain follows:

$$\begin{pmatrix} \dfrac{\partial \varphi}{\partial x} \\ \dfrac{\partial \varphi}{\partial y} \end{pmatrix} = \mathcal{J}^{-1} \begin{pmatrix} \dfrac{\partial \varphi}{\partial \xi} \\ \dfrac{\partial \varphi}{\partial \eta} \end{pmatrix}. \tag{7.12}$$

It should be highlighted that since four-point stencil is the smaller stencil to consider in 2D, it seems *a priori* impossible to apply such a method to meshes composed of triangles. On triangles, the method can be applied without modification but the stencil pairs are then allowed to share points (Fig. 7.5). This procedure seems attractive and since the definition of the local stencil is purely geometric, the treatment can be performed one time at the beginning of the time loop. To our knowledge, such a procedure is not applied in many CFD codes and only one paper addresses the analysis of such a method. Finally, the curvilinear gradient method needs the smallest stencil among the considered techniques.

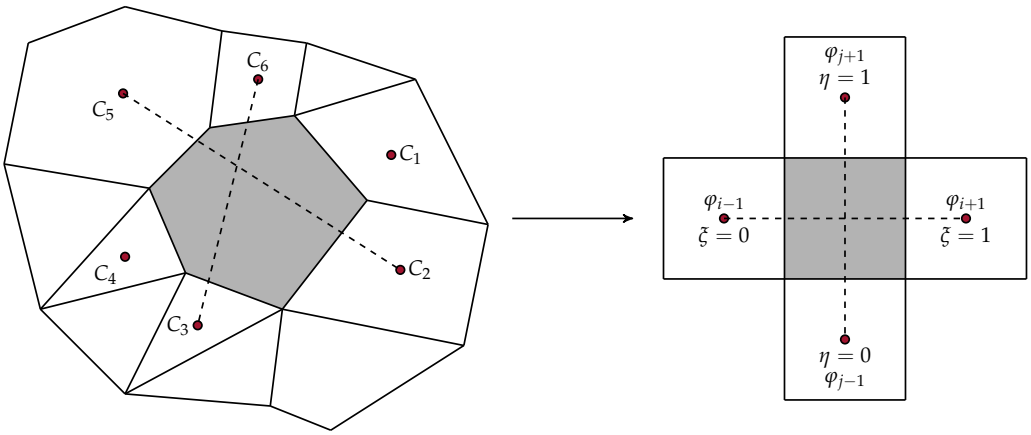


Fig. 7.4. *Curvilinear gradient stencil and mapping.*

The Magnetic Field Aligned Meshes (MFAM).
In the context of electric propulsion devices, the standard plasma transport theory states that the perpendicular electron transport coefficient is inversely proportional to the square of the

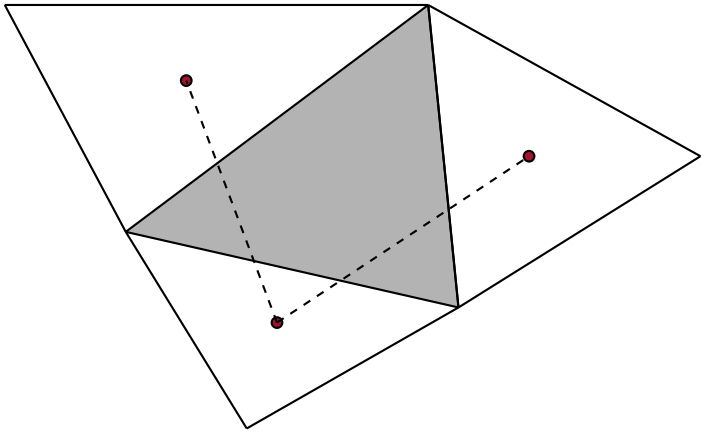


Fig. 7.5. *Shared-point curvilinear gradient stencil for a triangle.*

magnetic field strength B^{-2} and is always smaller, up to various orders of magnitude, than the parallel transport coefficient. This anisotropy of the transport coefficients may induce a numerical error when numerically solving the flow transport equations, the error becoming more significant if the computational mesh is misaligned with the principal magnetic directions, leading to numerical diffusion. In [105], the authors proposed to define first a new mesh generation algorithm in order to align mesh lines with diffusion direction. The generation of this type of meshes presents some challenges and several meshing strategies are implemented and analyzed in order to provide insight into achieving acceptable mesh regularity. Once the mesh generation is performed, the second analysis in [105] deals with the comparison of standard diffusion schemes on such grids. And as expected, standard diffusion schemes perform better when mesh lines and physical variations are aligned. In many aspects, the MFAM method can be seen as a mesh adaptation procedure and the mesh is locally optimized in order to minimize numerical errors.

7.1.7 Other diffusion schemes

Diffusion scheme dedicated to anisotropic operator.

Eymard, Gallouët and Herbin [106] proposed a solver for diffusion equation with anisotropic operators. Diffusive flux of the isotropic operator is solved by Finite Difference and the anisotropic operator is solved by a dedicated solver proposed by the authors. This method requires that the "fictitious edge" between two neighboring cell centers and the interface are orthogonal. It is possible if the cell center is assumed to be the center of the circumscribed circle. However, the technique cannot be extended straightforwardly to any element shape.

Method based on the Finite Element approximation.

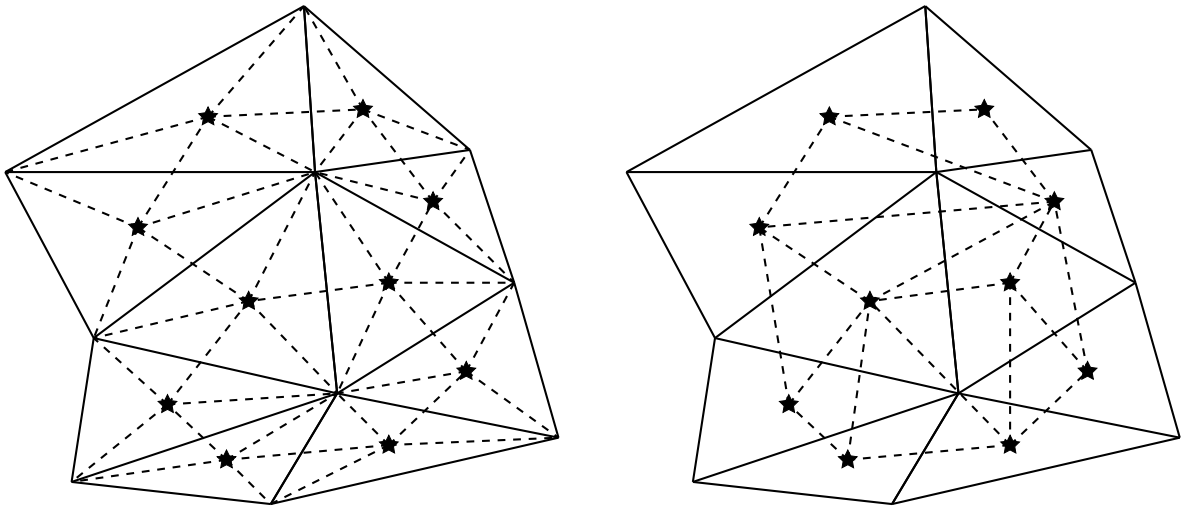
For a cell-vertex Finite Volume approximation, dual cells are built around mesh nodes using cell centers, face centers and edge mid-points as shown in Fig. 7.1. The $\mathcal{P}^1$ Finite Element method provides a simple framework for the computation of gradients on triangular meshes. In fact, an unknown is expressed on the basis of shape functions and by linearity, gradients are computed from the gradient of the shape functions. Moreover, an extension was proposed to handle general unstructured elements using new gradient still based on the $\mathcal{P}^1$ Finite Element approximation [107].

Flandrin *et al.* [108, 109] proposed to blend the cell-centered Finite Volume approximation for convection and the vertex-based Finite Element approximation for diffusion in an unstructured cell-centered Finite Volume solver. The motivation behind the approach is to use the high flexibility of the Finite Element technique to define the gradient by means of shape function gradient. To do so, the initial computational domain is covered by a new triangulation and

there are two ways to build this triangulation.

In the first approach, triangles involve both mesh nodes and cell centers. In this case, the diffusion flux on a two-dimensional face is always divided into two contributions with two different gradients as shown in Fig. 7.6. This approach can be seen as an extension of the diamond cell approach of Coirier, the main difference being in the way gradients are computed. Such an approach maintains the need for an interpolation to mesh nodes, something that is difficult to achieve in a parallel framework using a cell-centered Finite Volume solver with a face-based data structure. Hence, this approach will not be considered in the current study.

In the second approach, the triangulation only considers the cell centers. The new triangle grid intersects the initial mesh faces and produces facets as shown in Fig. 7.6. A gradient is then computed and applied for the flux integral computation on each facet. The definition of local facets is quite obvious in a two-dimensional framework. In a three-dimensional case, the situation is not straightforward and in general, the intersection of a mesh element with a tetrahedron can produce complex polyhedra. This approach is also not considered in this chapter due to its inherent complexity and high CPU cost since it requires to compute one gradient per facet.



(a) *Definition of the triangulation for the first approach.*

(b) *Definition of the triangulation for the second approach.*

Fig. 7.6. *Definition of the triangulation for both approaches. Mesh edges are represented with solid line, cell centers with stars and the triangulation by dashed lines.*

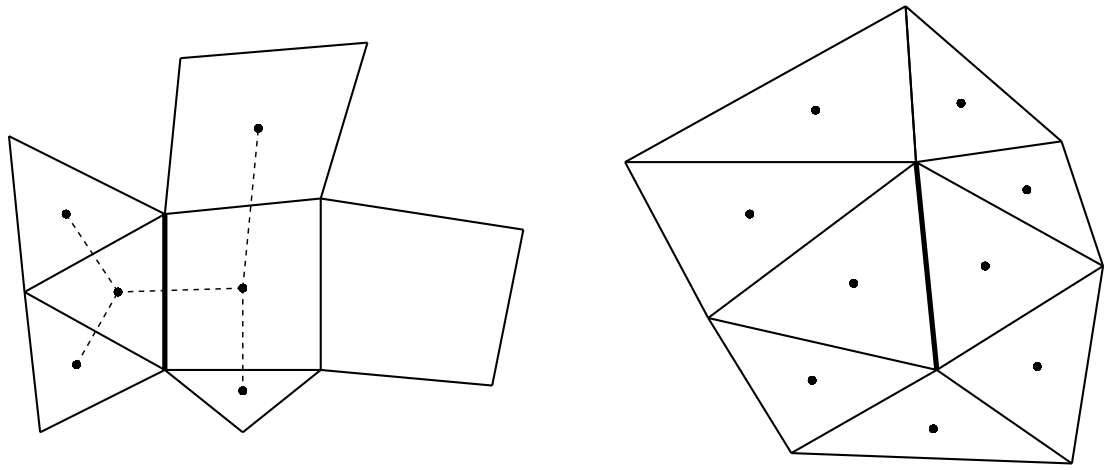
Least-Squares Method for interface gradient.

The previous Least-Squares approach in Sec. 7.1.2 consists in computing first a cell-centered gradient. This cell-centered gradient then serves to define the interface gradient for the evaluation of the viscous flux. In 1991, Dubois [110, 111] proposed to use a Least-Squares approach to define directly the interface gradient. The idea is the following. Any gradient is a local and linear function of the solutions W_i in a predefined neighborhood V_Σ around a face Σ of the mesh. As a consequence, the gradient is evaluated using:

$$\nabla \boldsymbol{W}_\Sigma = \sum_{i \in V_\Sigma} \alpha_{\Sigma,i} W_i, \tag{7.13}$$

where W_i is the averaged solution in cell i of the neighborhood V_Σ and $\alpha_{\Sigma,i}$ is a weighted coefficient. (7.13) can be seen as a linear system of unknowns $\alpha_{\Sigma,i}$. Defining the origin of the local reference frame as the center of the face, an interpolation Lagrange polynomial is first defined on the stencil V_Σ and then differentiated to obtain the weights. Fig. 7.7 gives two

examples of neighborhood reconstruction using mesh faces or mesh nodes. Of course, the quality of this approach depends on the stencil for the polynomial reconstruction and this technique requires the storage of the interpolation coefficients and the cell indices for the face neighborhood.



(a) *Neighborhood based on faces.* **(b)** *Neighborhood based on nodes.*

Fig. 7.7. *Dubois' approach.*

7.1.8 Partial conclusion

Several classes of diffusion schemes were introduced:

- The Green-Gauss formulation in standard or corrected formulations,
- The Least-Squares technique for cell-centered gradient,
- The diamond cell approaches,
- The Finite Element approaches from cell-centered quantities with/without mesh-node quantities,
- A diffusion operator dedicated to anisotropic diffusion,
- The Least-Squares Method coupled with Lagrange interpolation for interface gradient.

Both Least-Squares and Green-Gauss formulations for computing cell-centered gradients and then interface gradients are widely used. The other techniques are not very popular. From our point of view, the method of Dubois using least square technique and Lagrange interpolation is a viable alternative to standard methods due to its compact stencil and simple algorithm. Since the standard methods always follow a two-step procedure, a new interface gradient will be introduced. This interface gradient has a very compact stencil, is compatible with a cell-centered formulation on unstructured multi-element-shape grids and is easy to implement.

7.2 The new interface gradient scheme

The standard Finite Volume approach for diffusion involves an integral of the gradient over a mesh interface (7.2). In this section, we propose now a new technique to directly compute an interface gradient to be used in the flux computation. This method requires the definition of a new computational cell, called dual cell in this section, around the interface and the averaged gradient on the dual cell serves as interface gradient. In some aspects, the new diffusion scheme tends to take the benefit of the simplicity of diamond cell formulation, but avoids the interpolation to mesh nodes. This algorithm is optimized for a face-based data structure, as generally encountered in unstructured Finite Volume solvers.

7.2.1 Building a new dual cell including the interface

Let $F_{(i,j)}$ be the interface between cells Ω_i and Ω_j: $F_{(i,j)} = \Omega_i \cap \Omega_j$. $(\mathcal{L}_i)_{(i,j)}$ (resp. $(\mathcal{L}_j)_{(i,j)}$) denotes the list of faces of the volume Ω_i (resp. Ω_j) that share a node (resp. an edge) with $F_{(i,j)}$ in a two-dimensional (resp. in a three-dimensional) case. Note that the face $F_{(i,j)}$ does not belong to $(\mathcal{L}_i)_{(i,j)}$ nor $(\mathcal{L}_j)_{(i,j)}$. Let's take two examples. For a tetrahedron, $(\mathcal{L}_i)_{(i,j)}$ contains all the triangular faces but $F_{(i,j)}$. For a prism, if $F_{(i,j)}$ is a triangle, $(\mathcal{L}_i)_{(i,j)}$ contains the three quadrangular faces.

The new dual cell surrounding the interface can now be defined for two-dimensional and three-dimensional faces.

Remark: The procedure to define the new dual cell looks like the one applied in the vertex-centered finite-volume approach. It takes ingredients of the well-known extension of the definition of the median dual volume from simplex to primitive elements introduced by Dervieux [112] for the Euler equations and by Rostand and Stoufflet [113] for the Navier–Stokes equations.

Two-dimensional dual cell.

A general view of the dual volume for a mesh composed of triangles is shown in Fig. 7.8. The dual cell is limited by the face nodes and the midpoints for all edges in $(\mathcal{L}_i)_{(i,j)}$ and $(\mathcal{L}_j)_{(i,j)}$. These facets define a local region around $F_{(i,j)}$ which is partially closed. In Ω_i, the closure involves these new midpoints and the cell center. The same approach is applied on Ω_j.

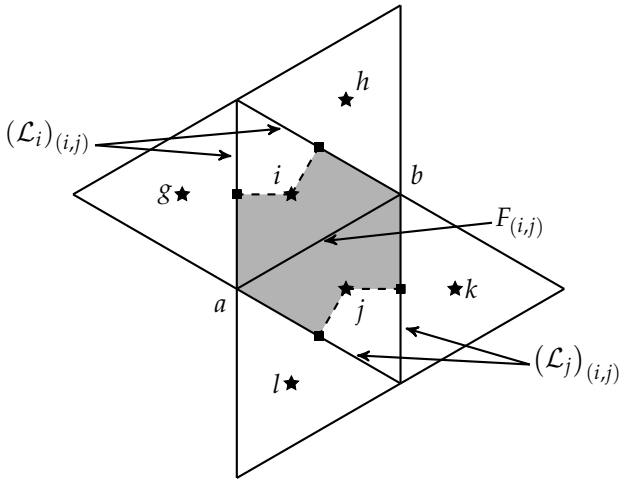


Fig. 7.8. *Definition of the dual volume around the face $F_{(i,j)}$ for a two-dimensional mesh composed of triangles. Squares (resp. stars) represent the face centers (resp. cell centers). The indexes g, h, l and k refer to the direct neighbors of the volumes i and j in a face-based framework.*

Three-dimensional dual cell.

The three-dimensional dual cell definition is an extension of the two-dimensional approach. Now, the dual cell is limited by two kinds of facets. The first kind of facets is built using the faces in $(\mathcal{L}_i)_{(i,j)}$ (resp. $(\mathcal{L}_j)_{(i,j)}$) for Ω_i (resp. Ω_j). The facets are limited by the face nodes, the face center and the midpoints for edges that share one node with the face. This first kind of facets is introduced in Fig. 7.9. The second kind of facets is illustrated on Fig. 7.10. It is composed of triangles only. Any new triangle is based on:

1. The center of faces of $(\mathcal{L}_i)_{(i,j)}$ and $(\mathcal{L}_j)_{(i,j)}$,
2. The edge midpoints of the faces belonging to $(\mathcal{L}_i)_{(i,j)}$ and $(\mathcal{L}_j)_{(i,j)}$,
3. The centers of the volume Ω_i or Ω_j.

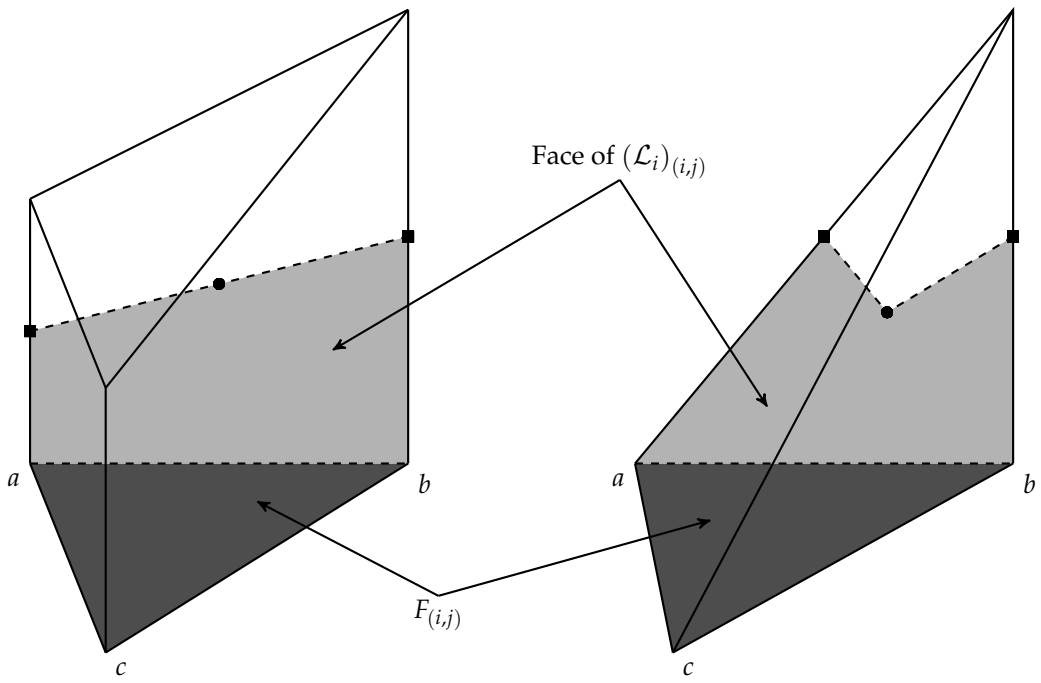


Fig. 7.9. *Part of the dual cell for the face (a, b, c) of a prism and a tetrahedron. The (background) face of interest is dark gray and the facet of the dual cell is in light grey. The square symbol represent edge midpoint while the circle represent the face center.*

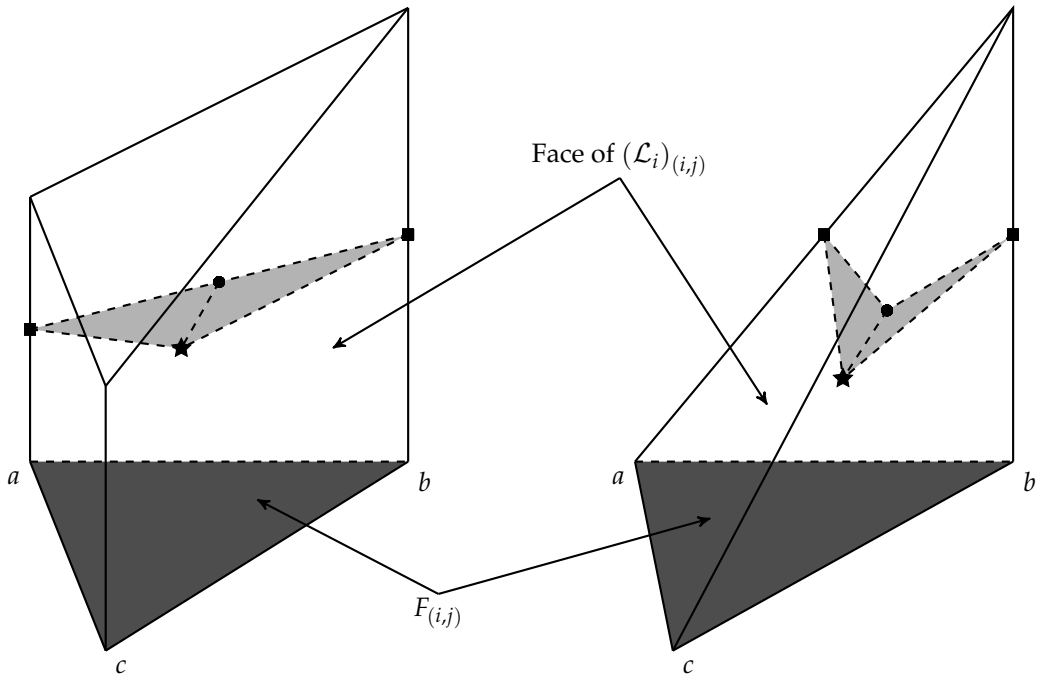


Fig. 7.10. *Part of the dual cell for the face (a, b, c) of a prism and a tetrahedron. The (background) face of interest is dark gray and the facets of the dual cell are in light gray. The square symbol represent edge midpoint while the circle represent the face center and the star is the cell center.*

7.2.2 Computation of the new gradient for the diffusive flux

Naive definition.

The procedure follows the one introduced for the diamond cell approach. A constant gradient is obtained over the dual cell and this gradient is chosen as interface gradient to be introduced in the surface integral issued from the diffusion term. The gradient is computed using a standard Green-Gauss approach. Such an approach involves the definition of the quantities on the different facets of the dual cell introduced previously. For the facets defined with the use of the cell centers, the interface unknown is the mean quantity over the cell. Other facets are defined from the mesh faces that separate two cells. The interface value is then defined as an average of the mean quantities on both sides of the interface. In 2D, following the notations introduced in Fig. 7.8:

$$\begin{aligned} \mathcal{V}_d^* \, \nabla \boldsymbol{W}|_{F_{(i,j)}} &= \overline{W}_i \, \boldsymbol{n}^i_{(i,j)} + \overline{W}_j \, \boldsymbol{n}^j_{(i,j)} \\ &+ \frac{\overline{W}_i + \overline{W}_g}{2} \frac{\mathcal{S}_{(i,g)}}{2} \boldsymbol{n}_{(i,g)} + \frac{\overline{W}_i + \overline{W}_h}{2} \frac{\mathcal{S}_{(i,h)}}{2} \boldsymbol{n}_{(i,h)} \\ &+ \frac{\overline{W}_j + \overline{W}_k}{2} \frac{\mathcal{S}_{(j,k)}}{2} \boldsymbol{n}_{(j,k)} + \frac{\overline{W}_j + \overline{W}_l}{2} \frac{\mathcal{S}_{(j,l)}}{2} \boldsymbol{n}_{(j,l)} \end{aligned} \tag{7.14}$$

where $\mathcal{V}_d^*$ is the volume of the dual cell. $\boldsymbol{n}^i_{(i,j)}$ is the outward vector defined as the sum of the normal vectors of the facets represented in dashed lines in Fig. 7.8. Following the notations introduced in Fig. 7.8, we have $\boldsymbol{n}^i_{(i,j)} = -\frac{1}{2}\boldsymbol{n}_{(i,j)}$ and $\boldsymbol{n}^j_{(i,j)} = \frac{1}{2}\boldsymbol{n}_{(i,j)}$ for a triangle.

It can be highlighted that this very simple procedure will not be accurate. Actually, a second-order gradient definition needs a third-order extrapolation of the unknowns in the face centers. It is clear that the definition of interface quantities for the facets in $(\mathcal{L}_i)_{(i,j)}$ is not third-order accurate. The same situation occurs for facets based on the cell-center node. It can be shown that, used for the Laplacian operator computation, this gradient formulation leads to a non consistent Laplacian operator. Consequently, this formulation must be corrected but this naive approach is helpful to explain the basics of the procedure.

Correction procedure.

The general formulation of the naive gradient definition can be seen as a linear combination of cell-centered averaged quantities over a predefined stencil based on cells that share a face (or a part of face) with the dual control volume. Our idea is now to keep the stencil unchanged but to allow a new definition of the weights in order to attain both consistence and desired accuracy.

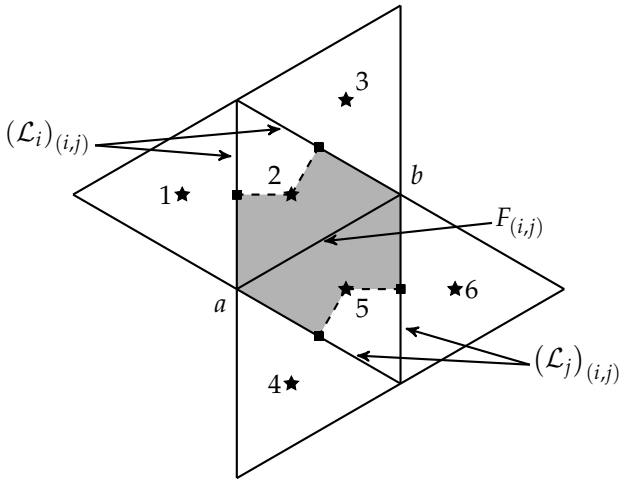


Fig. 7.11. *Definition of six neighboring cells of the face $F_{(i,j)}$ for the correction procedure.*

Let us consider a mesh composed of triangles only, as in Fig. 7.11. For a triangle, the formulation involves the six states of neighboring cells of the face $F_{(i,j)}$ The formulation of the Unstructured Interface Gradient (UIG) is then simply corrected by writing a linear combination of these states:

$$\nabla \boldsymbol{W}|_{F_{(i,j)}} = \sum_{i=1}^{6} W_i \boldsymbol{\xi_i}, \tag{7.15}$$

where $\boldsymbol{\xi_i} = (\alpha_i, \beta_i)^\top$ are two-dimensional vectors. Considering the Taylor series expansion, it is possible to write two linear systems, one for the x-component to find the α_i coefficients, one for the y-component to find the β_i coefficients, to ensure the second-order of accuracy:

$$\sum_{i=1}^{6} \alpha_i = 0, \tag{7.16}$$

$$\sum_{i=1}^{6} \alpha_i (x_i - x_F) = 1, \tag{7.17}$$

$$\sum_{i=1}^{6} \alpha_i (y_i - y_F) = 0, \tag{7.18}$$

$$\sum_{i=1}^{6} \alpha_i (x_i - x_F)^2 = 0, \tag{7.19}$$

$$\sum_{i=1}^{6} \alpha_i (y_i - y_F)^2 = 0, \tag{7.20}$$

$$\sum_{i=1}^{6} \alpha_i (x_i - x_F)(y_i - y_F) = 0, \tag{7.21}$$

and

$$\sum_{i=1}^{6} \beta_i = 0, \tag{7.22}$$

$$\sum_{i=1}^{6} \beta_i (x_i - x_F) = 0, \tag{7.23}$$

$$\sum_{i=1}^{6} \beta_i (y_i - y_F) = 1, \tag{7.24}$$

$$\sum_{i=1}^{6} \beta_i (x_i - x_F)^2 = 0, \tag{7.25}$$

$$\sum_{i=1}^{6} \beta_i (y_i - y_F)^2 = 0, \tag{7.26}$$

$$\sum_{i=1}^{6} \beta_i (x_i - x_F)(y_i - y_F) = 0, \tag{7.27}$$

where the point of coordinates (x_F, y_F) is the center of the face $F_{(i,j)}$. Both systems are then solved using the Moore-Penrose pseudo-inverse technique [82, 83]. By definition, a consistent second-order gradient evaluation is obtained on general unstructured grids since the metrics is taken into account. The conservation property is straightforward due to the definition of an unique gradient on any face. This gradient can now be used to compute the Laplacian operator which is first-order accurate on general unstructured grids. Furthermore, it will not lead to checkerboard instability since it involves all the stencil states and in particular, the nearest neighbors.

General comments.

For a face shared by two quadrangles, the new gradient method needs six cells to construct the value at the interface. Green-Gauss and Least-Square Methods have a larger stencil composed of eight cells. The proposed technique needs a smaller stencil. The same situation occurs in 3D. For instance, for a face shared by two hexahedra, the new gradient method needs ten cells whereas other methods require twelve cells.

Moreover, the computation of the weights needs only local information, especially in agreement with the basic formulation of a cell-centered Finite Volume solver with a face-based data structure. In particular, such a scheme is designed to avoid a complex stencil definition, especially in a parallel environment when all cells do not have the same shape.

In addition, the proposed technique is applied to faces based on 3 points (triangles) or 4 points (quadrangles). The procedure can be easily extended to polygonal facets. In this case, depending on the definitions of $(\mathcal{L}_i)_{(i,j)}$ and $(\mathcal{L}_j)_{(i,j)}$, the stencil is extended but the general framework for the solution, based on Moore-Penrose pseudo-inverse method, can be kept.

Actually, this procedure can also be applied on an incomplete dual cell for the computation of the boundary gradient, when a Dirichlet condition is considered. This is of strong importance since a simple copy of the averaged gradient in the cell adjacent to a Dirichlet boundary face is seen as a zeroth-order recopy. Coupled with a second-order scheme, it makes the solution lose one order of accuracy: a first-order solution is recovered.

Finally, it should be highlighted that the proposed procedure guarantees a second-order accurate gradient. Such a procedure can also be identified as a $1-$exact reconstruction of the gradient on the face. The idea of the definition of the local correction uses ideas introduced in Chap. 8.

Numerical tests were performed first using the naive definition of the schemes. With the naive reconstruction, the proposed scheme performs as well as the standard Green-Gauss formulation and it is more accurate than the least-square formulation on regular quadrangles. The results are not provided here since the conclusions will change according to the proposed correction based on local metrics. Today, the numerical tests using the enhanced formulation have not been carried out to validate this gradient formulation yet. It is planned to complete the analysis by numerical experiments and to submit then a paper to the Journal of Scientific Computing [66].

8 *k*-exact schemes

"It is vain to do with more what can be done with less."

— William of Ockham (1287 - 1347)

Abstract. *In this chapter, a review of Finite Volume high-order convection schemes is made that includes the k-exact reconstruction with the Successive Corrections Method (SCM), the Essentially Non-Oscillatory (ENO) or Weighted Essentially Non-Oscillatory (WENO) schemes and the Multi-dimensional Optimal Order Detection (MOOD) approach. It is shown that all these methods are more of less equivalent since for a given set of points, it only exists one polynomial. Basically, only these schemes lead to high-order schemes in an unstructured cell-centered Finite Volume paradigm. Then, a simple example is described to learn how the SCM works. Several conclusions are then drawn, in particular about the efficiency of the method and the CPU time.*

8.1 Bibliography

In Sec. 7.2, a new Unstructured Interface Gradient was proposed. It was not discussed in this manuscript but basically, it could be imagined that this gradient could be used to define a new convection scheme and inserted into (6.14) and (6.15). It is then obvious that the accuracy of this convection scheme is strongly related to the accuracy of the Unstructured Interface Gradient. This means that this convection scheme would be written as a linear combination of states and their gradients for each cell. But why should the reconstruction be limited to an expression including only the states and their gradients? Why not adding the Hessian matrix in the formulation? This is exactly the principle of the k-exact approach. Of course, several approaches were developed but are all based on the k-exact approach. They follow the same idea of polynomial reconstruction and can be denoted as k-exact.

The principle of the k-exact reconstruction was introduced in the pioneering work of Barth and Frederickson [114] and developed by [115, 116, 117]. It consists in representing the local variations of the unknowns as polynomials of degree k. The stencil is chosen large enough to couple the monomials with the cell-centered unknowns through the resolution of a linear system of equations. Such a system can be solved directly but generally, the stencil is larger than the number of unknowns and the system is solved in a Least-Squares sense. Jalali and Ollivier-Gooch [118] use the Least-Squares Method over a large stencil to define high order polynomial approximation in the $k-$exact framework. They apply successfully the procedure to RANS simulations on anisotropic meshes and introduce several local corrections to remove (or at least reduce) the least-square method sensitivity to space directions.

An efficient implementation of the k-exact procedure was proposed by Haider *et al.* [87, 119, 120] with the Successive Corrections Method (SCM), Caraeni *et al.* [121, 122] and Yang *et al.* [123]. Here, the computation of the monomials of the k-reconstruction follows a successive differentiation technique. It is done by applying recursively a first-order accurate gradient on general unstructured grids. Such a procedure allows to consider only local reduced stencil to compute the p-th order derivative from the $(p-1)$-th order derivative and some corrections are introduced in order to keep the accuracy at each step in agreement with the requirements. The

k-exact reconstruction can be chosen to define a high-order representation of the cell-centered gradient in order to produce a high-order approximation of the interface state. Then, a Riemann solver computes the interface fluxes from the left and right states. Although this approach does not reduce the size of the stencil necessary to reconstruct the interface state, it is compact in terms of communications.

The Essentially Non-Oscillatory scheme (ENO) was first proposed by Harten *et al.* [124] and developed in [125, 126]. This approach is closely related to k-exact reconstruction. The ENO approach builds several polynomials by using different stencils in an unstructured Finite Volume framework. The smoothest polynomial is then selected to compute fluxes at the interface to prevent oscillations due to discontinuities. Unfortunately, the stencil is composed of several tens of cells for a fourth- or fifth-order reconstruction which may induce a large computational cost. Further details about this approach can be found in [127]. Similarly to the ENO scheme, the Weighted Essentially Non-Oscillatory scheme, firstly introduced in [128] and developed in [129, 130], defines several polynomials. Instead of choosing the smoothest polynomial however, the final polynomial is a weighted average of all polynomials.

The MOOD reconstruction [131, 132, 133] differs from the k-exact technique essentially by the form of the polynomial reconstruction. The polynomial is by nature able to recover the local averaged quantity over the considered cell. Such a result is not guaranteed by the k-exact reconstruction since the monomials are found by solving a least-squares problem.

8.2 One-dimensional Finite Difference example on a non-uniform mesh

In this section, the third-order Successive Corrections Method (SCM3) is applied on a one-dimensional linear advection equation over a non-uniform mesh with a constant advective velocity $c = 1.0\ [m/s]$ to discretize the convective term simply equal to the first derivative. f is the unknown and corresponds to the convected variable. The first derivative is noted f' and the second derivative f''. The stencil is shown in Fig. 8.1. Let us choose the Finite Difference framework to avoid the necessary distinction needed in a Finite Volume framework between the nodal value at cell center and the mean value over the cell.

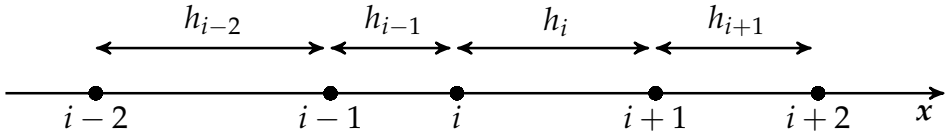


Fig. 8.1. *Stencil for the third-order Successive Corrections Method (SCM3).*

As stated in [85], to be able to build a third-order reconstruction, one needs a second-order derivative and a first-order second derivative. First of all, a first-order operator $\mathcal{D}^{(1)}$ for the first derivative is required. Considering the Taylor expansion for f_{i-1} and f_{i+1}, one obtains:

$$f_{i-1} = f_i - h_{i-1} f_i' + \frac{h_{i-1}^2}{2} f_i'' + \mathcal{O}\left(h^3\right), \tag{8.1}$$

$$f_{i+1} = f_i + h_i f_i' + \frac{h_i^2}{2} f_i'' + \mathcal{O}\left(h^3\right). \tag{8.2}$$

From both these Taylor expansions, the operator $\mathcal{D}^{(1)}$ is deduced:

$$\mathcal{D}^{(1)} f_i = \frac{h_{i-1} f_{i+1} - \left(h_{i-1} - h_i\right) f_i - h_i f_{i-1}}{2 h_i h_{i-1}}, \tag{8.3}$$

$$= f_i' + \frac{h_i - h_{i-1}}{4} f_i'' + \mathcal{O}\left(h^2\right), \tag{8.4}$$

which gives indeed a first-order derivative operator in the case of non-uniform meshes. On uniform meshes, it gives a second-order derivative operator but it will never be the case for general unstructured grids. With this operator, f'_{i-1}, f'_i and f'_{i+1} are evaluated. Then, the second derivative is computed by:

$$\mathcal{D}^{(1)} f'_i = \frac{h_{i-1} f'_{i+1} - (h_{i-1} - h_i) f'_i - h_i f'_{i-1}}{2 h_i h_{i-1}}, \tag{8.5}$$

$$= \frac{h_i^2 + 4 h_{i-1} h_i + h_{i-2} h_i + h_{i-1}^2 + h_{i-1} h_{i+1}}{8 h_i h_{i-1}} f''_i + \mathcal{O}(h). \tag{8.6}$$

This expression gives a non consistent formulation of the second derivative which must be corrected by multiplying by α^{-1} where the metric dependent coefficient α is defined by:

$$\alpha = \frac{h_i^2 + 4 h_{i-1} h_i + h_{i-2} h_i + h_{i-1}^2 + h_{i-1} h_{i+1}}{8 h_i h_{i-1}}, \tag{8.7}$$

and the first-order second derivative is obtained. Let us notice that the metric dependent coefficient α does not have a compact formulation since it depends on h_{i-2}, h_{i-1}, h_i and h_{i+1} This first-order second derivative is then used to correct the derivative to obtain a second-order derivative. Indeed,

$$\mathcal{D}^{(1)} f_i - \frac{h_i - h_{i-1}}{4} \alpha^{-1} \mathcal{D}^{(1)} f'_i = f'_i + \mathcal{O}\left(h^2\right). \tag{8.8}$$

Finally, f'_i is second-order accurate and f''_i is first-order accurate. This gives a second-order Finite Difference convection scheme even on non-uniform meshes. Consequently, the third-order Successive Corrections Method (SCM3) gives a second-order convection scheme on general non-uniform meshes.

8.3 One-dimensional Finite Volume example on a non-uniform mesh

In this section, the third-order Successive Corrections Method (SCM3) is applied on a one-dimensional linear advection equation over a non-uniform mesh with a constant advective velocity $c = 1.0\ [m/s]$ to discretize the convective term. f is the unknown and corresponds to the convected variable. The first derivative is noted $f^{(1)}$, the second derivative $f^{(2)}$ and the third derivative $f^{(3)}$. The stencil is shown in Fig. 8.2. h_i is the size of cell i and $d_i = (h_i + h_{i+1})/2$ is the distance between the center of cell i and the center of cell $i+1$. Finally, let us assume that it exists $\forall i \in [1, N]$, $(a_i, b_i) \in \mathbb{R}^2$ such as $h_i = a_i h = \mathcal{O}(h)$ and $d_i = b_i h = \mathcal{O}(h)$ where N is the number of cell of the considered mesh.

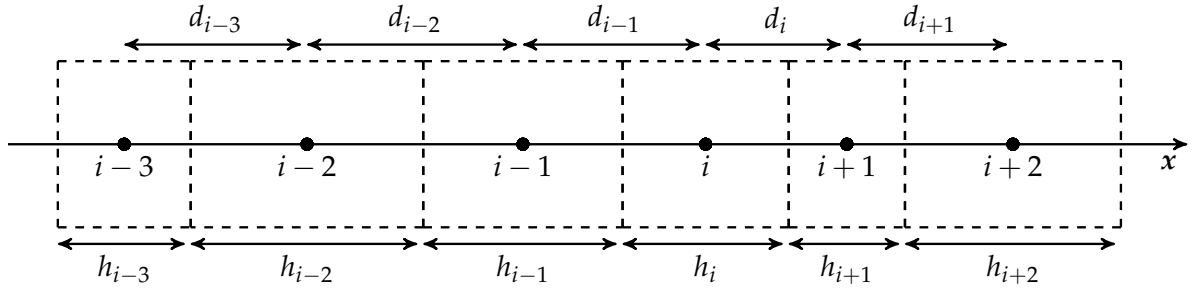


Fig. 8.2. *Stencil for the third-order Successive Corrections Method (SCM3). The black dots are the cell centers. h_i is the size of cell i and $d_i = (h_i + h_{i+1})/2$ is the distance between the center of cell i and the center of cell $i+1$.*

According to (2.4), the Finite Volume formulation for a one-dimensional linear advection with a constant advective velocity c is given by:

$$\frac{\mathrm{d}}{\mathrm{d}t} \bar{f}_i + c \cdot \frac{f_{i+1/2} - f_{i-1/2}}{h_i} = 0, \tag{8.9}$$

where $\overline{f}_i$ is the mean value of f over the cell i.

8.3.1 Link between nodal value and averaged value

The link between nodal value and averaged value is established. For sake of clarity, let us consider the cell i of size h. Let us note $\delta h = h/2$. The center x_i of cell i is taken equal to 0. Considering the Taylor expansion of $f(x)$ at $x = 0$:

$$f(x) = \sum_{k=0}^{n-1} f_i^{(k)}(x) \frac{x^k}{k!} + \mathcal{O}(x^n). \tag{8.10}$$

Taking the mean value $\overline{f}_i$ of $f(x)$ over the cell i, one obtains:

$$\begin{aligned}
\overline{f}_i &= \frac{1}{h} \int_{-\delta h}^{+\delta h} f(x)\, \mathrm{d}x, && (8.11)\\
&= \frac{1}{h} \int_{-\delta h}^{+\delta h} \sum_{k=0}^{n-1} f_i^{(k)}(x) \frac{x^k}{k!}\, \mathrm{d}x + \mathcal{O}(h^n), && (8.12)\\
&= \frac{1}{h} \sum_{k=0}^{n-1} \frac{f_i^{(k)}(x)}{k!} \int_{-\delta h}^{+\delta h} x^k\, \mathrm{d}x + \mathcal{O}(h^n), && (8.13)\\
&= \frac{1}{h} \sum_{k=0}^{n-1} \frac{f_i^{(k)}(x)}{k!} \left[\frac{x^{k+1}}{k+1}\right]_{-\delta h}^{+\delta h} + \mathcal{O}(h^n), && (8.14)\\
&= \frac{1}{h} \sum_{k=0}^{n-1} \frac{f_i^{(k)}(x)}{(k+1)!} \left[\delta h^{k+1} - (-\delta h)^{k+1}\right] + \mathcal{O}(h^n), && (8.15)\\
&= \sum_{k=0}^{n-1} \frac{f_i^{(k)}(x)}{2^{k+1}(k+1)!} \left[h^k - (-1)^{k+1} h^k\right] + \mathcal{O}(h^n). && (8.16)
\end{aligned}$$

Besides, $h^k - (-1)^{k+1} h^k = 0$ if k is odd and $h^k - (-1)^{k+1} h^k = 2h^k$ if k is even. Finally, a general formula of order $2n+2$ is obtained:

$$\overline{f}_i = \sum_{k=0}^{n} \frac{f_i^{(2k)}(x)}{(2k+1)!} \left(\frac{h}{2}\right)^{2k} + \mathcal{O}\left(h^{2n+2}\right), \tag{8.17}$$

from which one deduces the fourth- and sixth-order formulas:

$$\overline{f}_i = f_i + \frac{h^2}{24} f_i^{(2)} + \mathcal{O}\left(h^4\right), \tag{8.18}$$

$$\overline{f}_i = f_i + \frac{h^2}{24} f_i^{(2)} + \frac{h^4}{1920} f_i^{(4)} + \mathcal{O}\left(h^6\right). \tag{8.19}$$

8.3.2 *k*-exact principle

Like all Finite Volume schemes, the *k*-exact schemes aim to reconstruct an approximation for the interface fluxes. In particular, a 2-exact scheme builds an approximation of interface fluxes by a a polynomial of degree 2. It then gives a third-order approximation of interface fluxes. The interface flux at $i + 1/2$ for an advection equation with a positive advective velocity is given by $f_{i+1/2}^L$ where the superscript L means left. Then, the third-order upwind approximation of $f_{i+1/2}^L$ is extrapolated by:

$$f_{i+1/2}^L = f_i + \frac{h_i}{2} f_i^{(1)} + \frac{h_i^2}{8} f_i^{(2)} + \mathcal{O}\left(h^3\right). \tag{8.20}$$

This third-order upwind extrapolation formula requires a second-order first derivative and a first-order second derivative. But, first of all, a first-order first derivative operator $\mathcal{D}^{(1)}$ is built. Then, this operator is recursively applied to the first-order first derivative. This leads to an inconsistent second derivative. By taking into account the metrics, this inconsistent second derivative is corrected to obtain a first-order second derivative. Finally, this first-order second derivative is used to correct the first-order first derivative to obtain a second-order first derivative.

8.3.3 First-order operator for the first derivative

Let us build the first-order operator $\mathcal{D}^{(1)}$ to approximate the first derivative. Considering the Taylor expansion of f_{i-1} and f_{i+1} at the center of cell i, one obtains:

$$f_{i-1} = f_i - d_{i-1} f_i^{(1)} + \frac{d_{i-1}^2}{2} f_i^{(2)} - \frac{d_{i-1}^3}{6} f_i^{(3)} + \mathcal{O}\left(h^4\right), \tag{8.21}$$

$$f_{i+1} = f_i + d_i f_i^{(1)} + \frac{d_i^2}{2} f_i^{(2)} + \frac{d_i^3}{6} f_i^{(3)} + \mathcal{O}\left(h^4\right). \tag{8.22}$$

Using the relation between f_i and $\overline{f}_i$, one finds:

$$f_{i-1} = \overline{f}_i - \frac{h_i^2}{24} f_i^{(2)} - d_{i-1} f_i^{(1)} + \frac{d_{i-1}^2}{2} f_i^{(2)} - \frac{d_{i-1}^3}{6} f_i^{(3)} + \mathcal{O}\left(h^4\right), \tag{8.23}$$

$$f_{i+1} = \overline{f}_i - \frac{h_i^2}{24} f_i^{(2)} + d_i f_i^{(1)} + \frac{d_i^2}{2} f_i^{(2)} + \frac{d_i^3}{6} f_i^{(3)} + \mathcal{O}\left(h^4\right), \tag{8.24}$$

and

$$f_{i-1} = \overline{f}_i - d_{i-1} f_i^{(1)} + \left(\frac{d_{i-1}^2}{2} - \frac{h_i^2}{24}\right) f_i^{(2)} - \frac{d_{i-1}^3}{6} f_i^{(3)} + \mathcal{O}\left(h^4\right), \tag{8.25}$$

$$f_{i+1} = \overline{f}_i + d_i f_i^{(1)} + \left(\frac{d_i^2}{2} - \frac{h_i^2}{24}\right) f_i^{(2)} + \frac{d_i^3}{6} f_i^{(3)} + \mathcal{O}\left(h^4\right). \tag{8.26}$$

Using the relation between f_{i-1} and $\overline{f}_{i-1}$ and f_{i+1} and $\overline{f}_{i+1}$, it comes:

$$\overline{f}_{i-1} - \frac{h_{i-1}^2}{24} f_{i-1}^{(2)} = \overline{f}_i - d_{i-1} f_i^{(1)} + \left(\frac{d_{i-1}^2}{2} - \frac{h_i^2}{24}\right) f_i^{(2)} - \frac{d_{i-1}^3}{6} f_i^{(3)} + \mathcal{O}\left(h^4\right), \tag{8.27}$$

$$\overline{f}_{i+1} - \frac{h_{i+1}^2}{24} f_{i+1}^{(2)} = \overline{f}_i + d_i f_i^{(1)} + \left(\frac{d_i^2}{2} - \frac{h_i^2}{24}\right) f_i^{(2)} + \frac{d_i^3}{6} f_i^{(3)} + \mathcal{O}\left(h^4\right). \tag{8.28}$$

Considering the Taylor expansion of $f_{i-1}^{(2)}$ and $f_{i+1}^{(2)}$ at the center of cell i, one obtains:

$$f_{i-1}^{(2)} = f_i^{(2)} - d_{i-1} f_i^{(3)} + \mathcal{O}\left(h^2\right), \tag{8.29}$$

$$f_{i+1}^{(2)} = f_i^{(2)} + d_i f_i^{(3)} + \mathcal{O}\left(h^2\right). \tag{8.30}$$

Substituting (8.29) in (8.27) and (8.30) in (8.28), it comes:

$$\begin{aligned} \overline{f}_{i-1} = \overline{f}_i - d_{i-1} f_i^{(1)} &+ \left(\frac{d_{i-1}^2}{2} - \frac{h_i^2}{24} + \frac{h_{i-1}^2}{24}\right) f_i^{(2)} \\ &- \left(\frac{d_{i-1}^3}{6} + \frac{h_{i-1}^2 d_{i-1}}{24}\right) f_i^{(3)} + \mathcal{O}\left(h^4\right), \end{aligned} \tag{8.31}$$

$$\overline{f}_{i+1} = \overline{f}_i + d_i f_i^{(1)} + \left(\frac{d_i^2}{2} - \frac{h_i^2}{24} + \frac{h_{i+1}^2}{24}\right) f_i^{(2)} + \left(\frac{d_i^3}{6} + \frac{h_{i+1}^2 d_i}{24}\right) f_i^{(3)} + \mathcal{O}\left(h^4\right). \tag{8.32}$$

Computing $d_{i-1}\overline{f}_{i+1} - (d_{i-1} - d_i)\,\overline{f}_i - d_i\overline{f}_{i-1}$, one finds:

$$\begin{aligned} d_{i-1}\overline{f}_{i+1} - (d_{i-1} - d_i)\,\overline{f}_i - d_i\overline{f}_{i-1} = {} & 2d_{i-1}d_i f_i^{(1)} \\ & + \left(\frac{d_{i-1}d_i^2}{2} - \frac{h_i^2 d_{i-1}}{24} + \frac{h_{i+1}^2 d_{i-1}}{24}\right) f_i^{(2)} \\ & - \left(\frac{d_{i-1}^2 d_i}{2} - \frac{h_i^2 d_i}{24} + \frac{h_{i-1}^2 d_i}{24}\right) f_i^{(2)} \\ & + \left(\frac{d_{i-1}d_i^3}{6} + \frac{h_{i+1}^2 d_{i-1}d_i}{24}\right) f_i^{(3)} \\ & + \left(\frac{d_{i-1}^3 d_i}{6} + \frac{h_{i-1}^2 d_{i-1}d_i}{24}\right) f_i^{(3)} + \mathcal{O}\left(h^5\right). \end{aligned} \tag{8.33}$$

From (8.33), one can deduce the first-order operator $\mathcal{D}^{(1)}$ for the first derivative:

$$\mathcal{D}^{(1)}\overline{f}_i = \frac{d_{i-1}\overline{f}_{i+1} - (d_{i-1} - d_i)\,\overline{f}_i - d_i\overline{f}_{i-1}}{2d_{i-1}d_i}. \tag{8.34}$$

Indeed, simplifying (8.33), one finally finds:

$$\begin{aligned} \mathcal{D}^{(1)}\overline{f}_i = {} & f_i^{(1)} \\ & + \left(\frac{d_i}{4} - \frac{h_i^2}{48d_i} + \frac{h_{i+1}^2}{48d_i} - \frac{d_{i-1}}{4} + \frac{h_i^2}{48d_{i-1}} - \frac{h_{i-1}^2}{48d_{i-1}}\right) f_i^{(2)} \\ & + \left(\frac{d_i^2}{12} + \frac{h_{i+1}^2}{48} + \frac{d_{i-1}^2}{12} + \frac{h_{i-1}^2}{48}\right) f_i^{(3)} + \mathcal{O}\left(h^3\right). \end{aligned} \tag{8.35}$$

Thereafter, for sake of clarity, let us note $\forall i \in [1, N]$:

$$\alpha_i = \frac{d_i}{4} - \frac{h_i^2}{48d_i} + \frac{h_{i+1}^2}{48d_i} - \frac{d_{i-1}}{4} + \frac{h_i^2}{48d_{i-1}} - \frac{h_{i-1}^2}{48d_{i-1}} = \frac{h_{i+1} - h_{i-1}}{6}, \tag{8.36}$$

$$\beta_i = \frac{d_i^2}{12} + \frac{h_{i+1}^2}{48} + \frac{d_{i-1}^2}{12} + \frac{h_{i-1}^2}{48}. \tag{8.37}$$

The first-order operator $\mathcal{D}^{(1)}$ for the first derivative is then given by:

$$\mathcal{D}^{(1)}\overline{f}_i = f_i^{(1)} + \alpha_i f_i^{(2)} + \beta_i f_i^{(3)} + \mathcal{O}\left(h^3\right). \tag{8.38}$$

Let us notice that $\alpha_i = \mathcal{O}(h)$ and $\beta_i = \mathcal{O}\left(h^2\right)$.

8.3.4 First-order operator for the second derivative

Let us recursively apply the first-order operator $\mathcal{D}^{(1)}$ to approximate the second derivative:

$$\mathcal{D}^{(2)}\overline{f}_i = \mathcal{D}^{(1)}\left(\mathcal{D}^{(1)}\overline{f}_i\right) \tag{8.39}$$

$$= \frac{d_{i-1}\mathcal{D}^{(1)}\overline{f}_{i+1} - (d_{i-1} - d_i)\,\mathcal{D}^{(1)}\overline{f}_i - d_i\mathcal{D}^{(1)}\overline{f}_{i-1}}{2d_{i-1}d_i}. \tag{8.40}$$

Substituting (8.38) in (8.40), one obtains:

$$\begin{aligned} 2d_{i-1}d_i\mathcal{D}^{(2)}\overline{f}_i = {} & d_{i-1}\left(f_{i+1}^{(1)} + \alpha_{i+1}f_{i+1}^{(2)} + \beta_{i+1}f_{i+1}^{(3)}\right) \\ & - (d_{i-1} - d_i)\left(f_i^{(1)} + \alpha_i f_i^{(2)} + \beta_i f_i^{(3)}\right) \\ & - d_i\left(f_{i-1}^{(1)} + \alpha_{i-1}f_{i-1}^{(2)} + \beta_{i-1}f_{i-1}^{(3)}\right) + \mathcal{O}\left(h^4\right). \end{aligned} \tag{8.41}$$

Considering the Taylor expansion of $f_{i-1}^{(1)}$, $f_{i-1}^{(2)}$, $f_{i-1}^{(3)}$, $f_{i+1}^{(1)}$, $f_{i+1}^{(2)}$ and $f_{i+1}^{(3)}$ at the center of cell i, one obtains:

$$f_{i-1}^{(1)} = f_i^{(1)} - d_{i-1} f_i^{(2)} + \frac{d_{i-1}^2}{2} f_i^{(3)} + \mathcal{O}\left(h^3\right) \tag{8.42}$$

$$f_{i-1}^{(2)} = f_i^{(2)} - d_{i-1} f_i^{(3)} + \mathcal{O}\left(h^2\right) \tag{8.43}$$

$$f_{i-1}^{(3)} = f_i^{(3)} + \mathcal{O}\left(h\right) \tag{8.44}$$

$$f_{i+1}^{(1)} = f_i^{(1)} + d_i f_i^{(2)} + \frac{d_i^2}{2} f_i^{(3)} + \mathcal{O}\left(h^3\right) \tag{8.45}$$

$$f_{i+1}^{(2)} = f_i^{(2)} + d_i f_i^{(3)} + \mathcal{O}\left(h^2\right) \tag{8.46}$$

$$f_{i+1}^{(3)} = f_i^{(3)} + \mathcal{O}\left(h\right). \tag{8.47}$$

Substituting (8.42), (8.43), (8.44), (8.45), (8.46) and (8.47) in (8.41), one finds:

$$\begin{aligned} 2d_{i-1}d_i\mathcal{D}^{(2)}\overline{f}_i = {} & d_{i-1}\left(f_i^{(1)} + d_i f_i^{(2)} + \frac{d_i^2}{2} f_i^{(3)}\right) \\ & + d_{i-1}\alpha_{i+1}\left(f_i^{(2)} + d_i f_i^{(3)}\right) \\ & + d_{i-1}\beta_{i+1} f_i^{(3)} \\ & - \left(d_{i-1} - d_i\right)\left(f_i^{(1)} + \alpha_i f_i^{(2)} + \beta_i f_i^{(3)}\right) \\ & - d_i\left(f_i^{(1)} - d_{i-1} f_i^{(2)} + \frac{d_{i-1}^2}{2} f_i^{(3)}\right) \\ & - d_i\alpha_{i-1}\left(f_i^{(2)} - d_{i-1} f_i^{(3)}\right) \\ & - d_i\beta_{i-1} f_i^{(3)} + \mathcal{O}\left(h^4\right). \end{aligned} \tag{8.48}$$

Gathering the terms, one obtains:

$$\begin{aligned} 2d_{i-1}d_i\mathcal{D}^{(2)}\overline{f}_i = {} & \left(2d_{i-1}d_i + d_{i-1}\alpha_{i+1} - \alpha_i\left(d_{i-1} - d_i\right) - d_i\alpha_{i-1}\right) f_i^{(2)} \\ & + \left(\frac{d_{i-1}d_i^2}{2} + d_{i-1}d_i\alpha_{i+1} + d_{i-1}\beta_{i+1} - \beta_i\left(d_{i-1} - d_i\right)\right. \\ & \left. - \frac{d_{i-1}^2 d_i}{2} + d_{i-1}d_i\alpha_{i-1} - d_i\beta_{i-1}\right) f_i^{(3)} + \mathcal{O}\left(h^4\right). \end{aligned} \tag{8.49}$$

Finally, the approximation of the second derivative is given by:

$$\begin{aligned} \mathcal{D}^{(2)}\overline{f}_i = {} & \left(1 + \frac{\alpha_{i+1}}{2d_i} - \frac{\alpha_i\left(d_{i-1} - d_i\right)}{2d_{i-1}d_i} - \frac{\alpha_{i-1}}{2d_{i-1}}\right) f_i^{(2)} \\ & + \left(\frac{d_i}{4} + \frac{\alpha_{i+1}}{2} + \frac{\beta_{i+1}}{2d_i} - \frac{\beta_i\left(d_{i-1} - d_i\right)}{2d_{i-1}d_i}\right. \\ & \left. - \frac{d_{i-1}}{4} + \frac{\alpha_{i-1}}{2} - \frac{\beta_{i-1}}{2d_{i-1}}\right) f_i^{(3)} + \mathcal{O}\left(h^2\right). \end{aligned} \tag{8.50}$$

Thereafter, for sake of clarity, let us note $\forall i \in [1, N]$:

$$\gamma_i = 1 + \frac{\alpha_{i+1}}{2d_i} - \frac{\alpha_i\left(d_{i-1} - d_i\right)}{2d_{i-1}d_i} - \frac{\alpha_{i-1}}{2d_{i-1}}, \tag{8.51}$$

$$\delta_i = \frac{d_i}{4} + \frac{\alpha_{i+1}}{2} + \frac{\beta_{i+1}}{2d_i} - \frac{\beta_i\left(d_{i-1} - d_i\right)}{2d_{i-1}d_i} - \frac{d_{i-1}}{4} + \frac{\alpha_{i-1}}{2} - \frac{\beta_{i-1}}{2d_{i-1}}. \tag{8.52}$$

The approximation of the second derivative is then given by:

$$\mathcal{D}^{(2)}\overline{f}_i = \gamma_i f_i^{(2)} + \delta_i f_i^{(3)} + \mathcal{O}\left(h^2\right). \tag{8.53}$$

Let us notice that $\gamma_i = \mathcal{O}(1)$ and $\delta_i = \mathcal{O}(h)$. However, one can notice that the operator $\mathcal{D}^{(2)}$ gives an inconsistent approximation for the second derivative because γ_i is different from 1 for general unstructured grids. To obtain a first-order second derivative, this operator must be corrected by multiplying by γ_i^{-1} and one obtains:

$$f_i^{(2)} = \gamma_i^{-1}\mathcal{D}^{(2)}\overline{f}_i - \gamma_i^{-1}\delta_i f_i^{(3)} + \mathcal{O}\left(h^2\right). \tag{8.54}$$

8.3.5 Correction procedure: second-order operator for the first derivative

At this stage, a first-order second derivative and a first-order first derivative are available. However, according to (8.20), a second-order first derivative is required. It is obtained by correcting the first-order first derivative with the first-order second derivative:

$$\begin{aligned} f_i^{(1)} &= \mathcal{D}^{(1)}\overline{f}_i - \alpha_i f_i^{(2)} - \beta_i f_i^{(3)} + \mathcal{O}\left(h^3\right) && (8.55)\\ &= \mathcal{D}^{(1)}\overline{f}_i - \alpha_i\left(\gamma_i^{-1}\mathcal{D}^{(2)}\overline{f}_i - \gamma_i^{-1}\delta_i f_i^{(3)}\right) - \beta_i f_i^{(3)} + \mathcal{O}\left(h^3\right). && (8.56)\end{aligned}$$

Consequently, the second-order first derivative is obtained by:

$$f_i^{(1)} = \mathcal{D}^{(1)}\overline{f}_i - \alpha_i\gamma_i^{-1}\mathcal{D}^{(2)}\overline{f}_i + \left(\alpha_i\gamma_i^{-1}\delta_i - \beta_i\right) f_i^{(3)} + \mathcal{O}\left(h^3\right). \tag{8.57}$$

Indeed, the operator $\mathcal{D}^{(1)}\overline{f}_i - \alpha_i\gamma_i^{-1}\mathcal{D}^{(2)}\overline{f}_i$ is second-order accurate to approximate the first derivative $f_i^{(1)}$ since $\alpha_i\gamma_i^{-1}\delta_i - \beta_i = \mathcal{O}\left(h^2\right)$.

8.3.6 Extrapolation formula

Combining (8.57) and (8.54) into (8.20), one obtains:

$$\begin{aligned} f_{i+1/2}^L = f_i + \frac{h_i}{2}&\left(\mathcal{D}^{(1)}\overline{f}_i - \alpha_i\gamma_i^{-1}\mathcal{D}^{(2)}\overline{f}_i + \left(\alpha_i\gamma_i^{-1}\delta_i - \beta_i\right) f_i^{(3)}\right)\\ &+ \frac{h_i^2}{8}\left(\gamma_i^{-1}\mathcal{D}^{(2)}\overline{f}_i - \gamma_i^{-1}\delta_i f_i^{(3)}\right) + \mathcal{O}\left(h^4\right). \end{aligned} \tag{8.58}$$

Using the relation between f_i and $\overline{f}_i$ and (8.54), one finds:

$$\begin{aligned} f_i &= \overline{f}_i - \frac{h_i^2}{24} f_i^{(2)} + \mathcal{O}\left(h^4\right) && (8.59)\\ &= \overline{f}_i - \frac{h_i^2}{24}\left(\gamma_i^{-1}\mathcal{D}^{(2)}\overline{f}_i - \gamma_i^{-1}\delta_i f_i^{(3)}\right) + \mathcal{O}\left(h^4\right) && (8.60)\\ &= \overline{f}_i - \frac{h_i^2\gamma_i^{-1}}{24}\mathcal{D}^{(2)}\overline{f}_i + \frac{h_i^2\gamma_i^{-1}\delta_i}{24} f_i^{(3)} + \mathcal{O}\left(h^4\right). && (8.61)\end{aligned}$$

Using (8.58) and (8.61), one obtains:

$$\begin{aligned} f_{i+1/2}^L = \overline{f}_i &- \frac{h_i^2\gamma_i^{-1}}{24}\mathcal{D}^{(2)}\overline{f}_i + \frac{h_i^2\gamma_i^{-1}\delta_i}{24} f_i^{(3)}\\ &+ \frac{h_i}{2}\left(\mathcal{D}^{(1)}\overline{f}_i - \alpha_i\gamma_i^{-1}\mathcal{D}^{(2)}\overline{f}_i + \left(\alpha_i\gamma_i^{-1}\delta_i - \beta_i\right) f_i^{(3)}\right)\\ &+ \frac{h_i^2}{8}\left(\gamma_i^{-1}\mathcal{D}^{(2)}\overline{f}_i - \gamma_i^{-1}\delta_i f_i^{(3)}\right) + \mathcal{O}\left(h^4\right). \end{aligned} \tag{8.62}$$

Finally, the third-order upwind approximation of $f^L_{i+1/2}$ is extrapolated by:

$$\begin{aligned} f^L_{i+1/2} = \overline{f}_i + \frac{h_i}{2}\mathcal{D}^{(1)}\overline{f}_i + \left(\frac{h_i^2\gamma_i^{-1}}{12} - \frac{h_i\alpha_i\gamma_i^{-1}}{2}\right)\mathcal{D}^{(2)}\overline{f}_i \\ + \left(\frac{h_i}{2}\left(\alpha_i\gamma_i^{-1}\delta_i - \beta_i\right) - \frac{h_i^2\gamma_i^{-1}\delta_i}{12}\right) f_i^{(3)} + \mathcal{O}\left(h^4\right), \end{aligned} \tag{8.63}$$

Indeed, one has:

$$\frac{h_i}{2}\left(\alpha_i\gamma_i^{-1}\delta_i - \beta_i\right) - \frac{h_i^2\gamma_i^{-1}\delta_i}{12} = \mathcal{O}\left(h^3\right). \tag{8.64}$$

Let us remark that the Finite Volume operator $\mathcal{X}_{FV}$ defined by:

$$\mathcal{X}_{FV}\left(\overline{f}_i\right) = \overline{f}_i + \frac{h_i}{2}\mathcal{D}^{(1)}\overline{f}_i + \left(\frac{h_i^2\gamma_i^{-1}}{12} - \frac{h_i\alpha_i\gamma_i^{-1}}{2}\right)\mathcal{D}^{(2)}\overline{f}_i, \tag{8.65}$$

is only a linear combination of $\overline{f}_{i+2}$, $\overline{f}_{i+1}$, $\overline{f}_i$, $\overline{f}_{i-1}$ and $\overline{f}_{i-2}$, which are the only quantities available in a Finite Volume code.

8.3.7 Truncation error

Following (8.63), a similar expression can be found for $f^L_{i-1/2}$. The truncation error $\mathcal{E}$ for the flux balance $f^L_{i+1/2} - f^L_{i-1/2}$ is then given by:

$$\begin{aligned} \mathcal{E}\left(f^L_{i+1/2} - f^L_{i-1/2}\right) = \left(\frac{h_i}{2}\left(\alpha_i\gamma_i^{-1}\delta_i - \beta_i\right) - \frac{h_i^2\gamma_i^{-1}\delta_i}{12}\right) f_i^{(3)} \\ - \left(\frac{h_{i-1}}{2}\left(\alpha_{i-1}\gamma_{i-1}^{-1}\delta_{i-1} - \beta_{i-1}\right) - \frac{h_{i-1}^2\gamma_{i-1}^{-1}\delta_{i-1}}{12}\right) f_{i-1}^{(3)} \\ + \mathcal{O}\left(h^4\right). \end{aligned} \tag{8.66}$$

Let us remark that $f_{i-1}^{(3)} = f_i^{(3)} + \mathcal{O}(h)$ and note:

$$\varepsilon_i = \frac{h_i}{2}\left(\alpha_i\gamma_i^{-1}\delta_i - \beta_i\right) - \frac{h_i^2\gamma_i^{-1}\delta_i}{12}, \tag{8.67}$$

it comes:

$$\mathcal{E}\left(f^L_{i+1/2} - f^L_{i-1/2}\right) = \left(\varepsilon_i - \varepsilon_{i-1}\right) f_i^{(3)} + \mathcal{O}\left(h^4\right). \tag{8.68}$$

Finally, the truncation error of the Finite Volume discretization is:

$$\mathcal{E}\left(\frac{f^L_{i+1/2} - f^L_{i-1/2}}{h_i}\right) = \frac{\varepsilon_i - \varepsilon_{i-1}}{h_i} f_i^{(3)} + \mathcal{O}\left(h^3\right). \tag{8.69}$$

Noticing that $\varepsilon_i = \mathcal{O}\left(h^3\right)$, this proves that this Finite Volume scheme is second-order accurate on general unstructured grids. More generally, as established in App. D, the truncation error of the spatial discretization is $p-1$-th order accurate if the extrapolation formula for the interface fluxes is p-th order accurate.

8.3.8 Specific case: uniform mesh

In the specific case of uniform meshes, several simplifications appear. First of all, $\forall i \in [1, N]$, the different coefficients previously introduced reads:

$$h_i = d_i = h, \tag{8.70}$$
$$\alpha_i = \delta_i = 0, \tag{8.71}$$
$$\beta_i = \frac{5h^2}{24}, \tag{8.72}$$
$$\gamma_i = 1. \tag{8.73}$$

Consequently, this simplifies the operator $\mathcal{D}^{(1)}$:

$$\mathcal{D}^{(1)}\overline{f}_i = \frac{\overline{f}_{i+1} - \overline{f}_{i-1}}{2h}, \tag{8.74}$$
$$= f_i^{(1)} + \frac{5h^2}{24} f_i^{(3)} + \mathcal{O}\left(h^3\right). \tag{8.75}$$

This offers a free second-order operator for the first derivative. No correction with the first-order operator for the second derivative is needed. Similarly, this also simplifies the operator $\mathcal{D}^{(2)}$:

$$\mathcal{D}^{(2)}\overline{f}_i = \frac{\overline{f}_{i+2} - 2\overline{f}_i + \overline{f}_{i-2}}{4h^2}, \tag{8.76}$$
$$= f_i^{(2)} + \mathcal{O}\left(h^2\right). \tag{8.77}$$

This offers a free second-order operator for the second derivative. Finally, the third-order upwind approximation of $f^L_{i+1/2}$ is simply extrapolated by:

$$f^L_{i+1/2} = \overline{f}_i + \frac{h}{2}\mathcal{D}^{(1)}\overline{f}_i + \frac{h^2}{12}\mathcal{D}^{(2)}\overline{f}_i - \frac{5h^3}{48} f_i^{(3)} + \mathcal{O}\left(h^4\right) \tag{8.78}$$
$$= \overline{f}_i + \frac{\overline{f}_{i+1} - \overline{f}_{i-1}}{4} + \frac{\overline{f}_{i+2} - 2\overline{f}_i + \overline{f}_{i-2}}{48} - \frac{5h^3}{48} f_i^{(3)} + \mathcal{O}\left(h^4\right) \tag{8.79}$$
$$= \frac{48\overline{f}_i + 12\overline{f}_{i+1} - 12\overline{f}_{i-1} + \overline{f}_{i+2} - 2\overline{f}_i + \overline{f}_{i-2}}{48} - \frac{5h^3}{48} f_i^{(3)} + \mathcal{O}\left(h^4\right) \tag{8.80}$$
$$= \frac{\overline{f}_{i+2} + 12\overline{f}_{i+1} + 46\overline{f}_i - 12\overline{f}_{i-1} + \overline{f}_{i-2}}{48} - \frac{5h^3}{48} f_i^{(3)} + \mathcal{O}\left(h^4\right). \tag{8.81}$$

Analogously, the third-order upwind approximation of $f^L_{i-1/2}$ is simply extrapolated by:

$$f^L_{i-1/2} = \overline{f}_{i-1} + \frac{h}{2}\mathcal{D}^{(1)}\overline{f}_{i-1} + \frac{h^2}{12}\mathcal{D}^{(2)}\overline{f}_{i-1} - \frac{5h^3}{48} f_{i-1}^{(3)} + \mathcal{O}\left(h^4\right) \tag{8.82}$$
$$= \frac{\overline{f}_{i+1} + 12\overline{f}_i + 46\overline{f}_{i-1} - 12\overline{f}_{i-2} + \overline{f}_{i-3}}{48} - \frac{5h^3}{48} f_{i-1}^{(3)} + \mathcal{O}\left(h^4\right). \tag{8.83}$$

In the specific case of uniform meshes, the Finite Volume operator $\mathcal{X}_{FV}$ becomes:

$$\mathcal{X}_{FV}\left(\overline{f}_i\right) = \frac{\overline{f}_{i+2} + 12\overline{f}_{i+1} + 46\overline{f}_i - 12\overline{f}_{i-1} + \overline{f}_{i-2}}{48} - \frac{5h^3}{48} f_i^{(3)}. \tag{8.84}$$

The truncation error $\mathcal{E}$ for the flux balance $f^L_{i+1/2} - f^L_{i-1/2}$ is then given by:

$$\mathcal{E}\left(f^L_{i+1/2} - f^L_{i-1/2}\right) = \frac{5h^3}{48}\left(f_{i-1}^{(3)} - f_i^{(3)}\right) + \mathcal{O}\left(h^4\right). \tag{8.85}$$

Let us remark that $f_{i-1}^{(3)} = f_i^{(3)} + \mathcal{O}(h)$ and it comes:

$$\mathcal{E}\left(f_{i+1/2}^L - f_{i-1/2}^L\right) = \mathcal{O}\left(h^4\right). \tag{8.86}$$

Finally, the truncation error of the Finite Volume discretization is:

$$\mathcal{E}\left(\frac{f_{i+1/2}^L - f_{i-1/2}^L}{h}\right) = \mathcal{O}\left(h^3\right). \tag{8.87}$$

This proves that this Finite Volume scheme is third-order accurate but only on uniform grids and one can compute $f_{i+1/2}^L - f_{i-1/2}^L$:

$$f_{i+1/2}^L - f_{i-1/2}^L = \frac{\overline{f}_{i+2} + 11\overline{f}_{i+1} + 34\overline{f}_i - 58\overline{f}_{i-1} + 13\overline{f}_{i-2} - \overline{f}_{i-3}}{48} + \mathcal{O}\left(h^4\right). \tag{8.88}$$

8.4 HPC efficiency

Pont [84] computed the number of points per wavelength necessary to fulfill a given error on dispersion and dissipation for second-order, third-order, fourth-order and fifth-order schemes. Moreover, he also evaluated the CPU time for each scheme. Based on both these considerations, it seems that increasing the order of accuracy of the k-exact scheme may not be so attractive. In other words, in certain cases, it seems to be more efficient in terms of CPU time to use the second-order scheme with more degrees of freedom than to use the third-order scheme with less degrees of freedom to achieve the same level of accuracy. For this reason, our choice was to study more promising high-order methods such as the Spectral Difference Method which belongs to the family of Spectral Discontinuous Methods.

V The Spectral Difference Method

9 Spectral analysis 129
9.1 Introduction
9.2 Spectral analysis for high-order Finite Difference methods
9.3 Effect of time integration on the spectral analysis
9.4 Spectral analysis of the Spectral Difference method
9.5 Extension to high wavenumbers for the Spectral Difference method
9.6 Comparison with standard high-order schemes
9.7 Conclusion

9 Spectral analysis

"Science is not everything, but science is very beautiful."

— Julius Robert Oppenheimer (1904 - 1967)

Abstract. *The spectral analysis is a basic tool to characterize the behavior of any convection scheme. By nature, the solution projected onto the Fourier basis enables to estimate the dissipation and the dispersion associated with the spatial discretization of the hyperbolic linear problem. In this chapter, we wish to revisit such analysis, focusing the attention on two key points. The first point concerns the effects of time integration on the spectral analysis. It is shown with standard high-order Finite Difference schemes dedicated to aeroacoustics that the time integration has an effect on the required number of points per wavelength. The situation depends on the choice of the coupled schemes (one for time integration, one for space derivative and one for the filter) and here, the compact scheme with its eighth-order filter seems to have a better spectral accuracy than the considered dispersion-relation preserving scheme with its associated filter, especially in terms of dissipation. Secondly, such a coupled space-time approach is applied to the new class of high-order Spectral Discontinuous approaches, focusing especially on the Spectral Difference method. A new way to address the specific spectral behavior of the scheme is introduced first for wavenumbers in* $[0, \pi]$*, following the Matrix Power method. For wavenumbers above* π*, an aliasing phenomenon always occurs but it is possible to understand and to control the aliasing of the signal. It is shown that aliasing depends on the polynomial degree and on the number of time steps. A new way to define dissipation and dispersion is introduced and applied to wavenumbers larger than* π*. Since the new criteria recover the previous results for wavenumbers below* π*, the new proposed approach is an extension of all the previous ones dealing with dissipation and dispersion errors. At last, since the standard Finite Difference schemes can serve as reference solution for their capability in aeroacoustics, it is shown that the Spectral Difference method is as accurate as (or even more accurate than) the considered Finite Difference schemes. This chapter led to a paper published in the Journal of Computational Physics [134] and to two conference papers presented at the TILDA - Symposium & Workshop on Industrial LES & DNS [135, 136].*

9.1 Introduction

Because of the continuous growth of available computational resources during the last decade, there was an increased interest in performing Large Eddy Simulation -LES- to solve industrial problems. Among these problems, aeroacoustics requires to compute and to transport accurately pressure waves around complex geometries and over a long distance. Many classes of schemes were proposed to perform LES during the last 30 years, depending on the underlying mathematical framework considered to discretized the Navier-Stokes equations.

First, in the context of Finite Difference -FD- formalism, high-order centered schemes for structured grids were built following the Taylor's expansion technique and their accuracy was compared with the one of spectral methods [137]. Any (high) order of accuracy can be attained but the number of degrees of freedom to update the solution at one point can be large. Two optimizations of FD schemes were introduced: the compact formulation of Lele [39] that leads to a spectral-like resolution and the Dispersion-Relation-Preserving -DRP- technique of Tam

and Webb [45] dedicated to aeroacoustics. The compact approach links several derivatives with unknowns located closely. By this way, a linear (implicit) system of equations links all derivatives with unknowns. For a given accuracy, the stencil of DRP schemes is larger than for the standard FD approach and the extra unknowns enable to control the numerical properties of the scheme: dissipation and dispersion. Both approaches being centered, they are non dissipative and a filter stabilizes the computations by dissipating wavenumbers. In this chapter, we consider compact and DRP schemes as standard ingredients for LES and it is assumed that they will provide reference results.

More recently, a new generation of high-order techniques denoted as *spectral discontinuous* emerged. Following the pioneering work of Reed and Hill [138], the Discontinuous Galerkin -DG- formulation was first applied to hyperbolic equations by Cockburn, Shu and co-authors [139, 140, 141] and opened many years of research and papers (see [142] as an example of reference book on DG method). The idea is to solve problems defined in the weak form inside any mesh cell, without requiring the solution to be continuous at the mesh interfaces. At the interface, the fluxes are computed using standard Riemann solvers, as in Finite Volume -FV- formalism. Therefore, the FV flux computation enables the coupling of the weak problems in surrounding cells and the FV fluxes make information going across mesh interfaces. Several alternative high-order methods have been recently introduced. Following the staggered-grid multidomain spectral method [143] for structured grids, Liu, Vinokur and Wang [144, 145] introduced the Spectral Difference -SD- method aiming at a simpler to implement and more efficient method than the current state of the art for the DG method. The approach was then extended to mixed elements [146]. The SD method takes benefit of the resolution of the strong differential form of the equations, as in FD, but does not assume that the solution is continuous on the whole mesh, as in FV. Another way to define a high-order polynomial reconstruction follows the definition of averaged quantities, as in FV. With the Spectral Volume -SV- approach [147, 148, 149, 150, 151], a polynomial reconstruction is defined inside any cell using the averaged quantities over sub-cells built by subdivision of the initial mesh elements. As before, several Riemann problems are solved on mesh boundaries since the solution polynomials are not required to be continuous at mesh interfaces. Finally, the Flux Reconstruction method introduced in 2007 by Huynh [152] solves the strong form of the equation. It can be seen as a collocated Spectral Difference scheme but the main difference occurs in the definition of the flux polynomial: now, a lifting operator [153, 154, 155, 156] is introduced to increase the polynomial degree of the initial flux polynomial by one. This is mandatory to recover the required polynomial degree after the computation of the divergence (hyperbolic) term. The main advantage of FR method is its ability to recover SD, SV and DG approaches for the linear advection equation, depending on the lifting operator [157]. Compared to the standard schemes for structured grids, the major advantage of DG, SV, SD and FR methods lies in their natural ability to handle unstructured meshes, which is a prerequisite to treat complex geometries. Moreover, such schemes use a very compact stencil, defined locally inside any mesh cell. This is also an advantage in terms of high performance computing required by massively-parallel LES.

When dealing with aeroacoustics, the first question to answer concerns the spectral accuracy of the chosen scheme: the key point concerns the required number of grid points per wavelength. The spectral analysis [5] consists of dealing with the space derivative and in comparing the numerical spatial derivative with the theoretical derivative, after projection onto the Fourier basis. This analysis is performed on the linear advection equation in a periodic domain with a harmonic initial solution:

$$\begin{cases} \dfrac{\partial u}{\partial t} + c\dfrac{\partial u}{\partial x} = \dfrac{\partial u}{\partial t} + c\,\mathbb{D}(u) = 0 \\ u_0(x) = u(0,x) = \exp(jkx) \text{ with } j^2 = -1, \end{cases} \tag{9.1}$$

where the function $u(x,t)$ is the unknown, c the constant advective velocity, k the constant wavenumber and $\mathbb{D}$ represents the spatial derivative operator. As a consequence, (9.1) will play an important role and for sake of clarity, the notation for the unknowns will be kept unchanged along the whole document.

The formulation of the Fourier spectral analysis makes the analysis simple for standard schemes based on Finite Difference paradigm. This is due to the fact that the degrees of freedom are coupled by the numerics and not by the method itself. As introduced by Hu *et al.* in 1999 [158] for the DG method, the situation is more complex for spectral discontinuous methods. For DG method, the authors show that the degrees of freedom are coupled by the definition of the local polynomial -inside any mesh cell-. Finally, the Fourier analysis can be performed as for FD approach but the final equation changes. Instead of one equation giving the complex-valued numerical wavenumber, one obtains, even for a scalar equation, a set of linear equations. Depending on the polynomial degree p, $p+1$ waves are solutions of the linear set of equations. In [158], it is then explained that among the $p+1$ waves traveling at different phase speeds, one mode is the *physical mode* as its frequency approximates the exact dispersion relation for a range of wavenumbers, while the others are the *parasite modes* due to the scheme. In other words, the solution is the superposition of one physical mode and p parasite modes. Such results were also obtained by Zhang *et al.* for three different formulations of the DG method [159]. The same kind of analysis with $p+1$ waves, one physical mode and p parasite modes, was also proposed for the SV approach [160], for the SD approach [161] and for the FR technique [154, 162, 163]. Finally, even if the occurrence of parasite modes is demonstrated, the proposed analysis does not explain the role of these parasite modes. Moreover, even if it can be proved that the eigenvalues of the system are periodic (with a period of 2π), the spectral analysis presented in [154, 158, 159, 160, 161, 162, 163] shows that the spectral behavior, playing with the wavenumber and the order p of the polynomials, is no longer 2π-periodic. This point is clearly not in agreement with mathematical requirements.

In this chapter, we introduce a new way to perform the Fourier spectral analysis for polynomial discontinuous methods and we compare the accuracy of the SD technique with two standard centered and stabilized FD schemes, the compact scheme of Lele [39] and a DRP scheme developed by Bogey and Bailly [44]. The remainder of this chapter unfolds as follows. In Sec. 9.3, the space-time Fourier spectral analysis is applied to the considered spatial FD schemes coupled with a low-storage second-order Runge Kutta scheme. In Sec. 9.4, a new analysis of the spectral Fourier approach for spectral discontinuous methods is introduced and applied to the SD method. In Sec. 9.5, we explicitly address the case of a wavenumber larger than π and we give the analysis in terms of number of points per wavelength for the SD method. Before concluding, standard FD schemes and SD scheme are finally compared in Sec. 9.6.

9.2 Spectral analysis for high-order Finite Difference methods

In this section, the standard Fourier analysis is applied to two FD schemes and standard results are recovered. They will play an important role for comparison with the new results shown in Sec. 9.3.

9.2.1 Spatial discretization

Let us introduce a one-dimensional domain decomposed of elements with uniform length Δx. In 1D, any convection equation with a constant advection velocity is linear and the key point concerns the definition of the numerical derivative. Let u_i be the discrete unknown at the point i and u'_i its derivative at the same location. The length of the computational domain is not taken into consideration: the goal is to derive scheme expressions for a given point far from the boundary (in order to have access to the whole scheme stencil). In the following, two schemes

frequently applied to aeroacoustics simulations are considered.

The first scheme is the sixth-order compact scheme of Lele [39] denoted CS6. This scheme is purely centered and for stabilization, damping high frequency waves is mandatory. Here, the compact filter denoted CF8 is the eighth-order filter designed by Visbal and Gaitonde [164]. The Lele's sixth-order compact scheme CS6 approximates the first-order spatial derivative with:

$$\frac{1}{3}u'_{i-1} + u'_i + \frac{1}{3}u'_{i+1} = \frac{14}{9}\frac{u_{i+1} - u_{i-1}}{2\Delta x} + \frac{1}{9}\frac{u_{i+2} - u_{i-2}}{4\Delta x}, \tag{9.2}$$

while the eighth-order compact filter CF8 designed by Visbal and Gaitonde is defined by:

$$\alpha_f u^f_{i-1} + u^f_i + \alpha_f u^f_{i+1} = \sum_{l=0}^{4} b_l \frac{u_{i+l} + u_{i-l}}{2}, \tag{9.3}$$

where the superscript f means filtered. The b_l coefficients in (9.3) are summarized in Tab. 9.1. α_f is the filter coefficient and it is equal to 0.47. This numerical setup is used by, *e.g.*, Aikens *et al.* [165] or Le Bras *et al.* [166].

Tab. 9.1. *CF8 compact filter coefficients.*

b_0	b_1	b_2	b_3	b_4
$\frac{93+70\alpha_f}{128}$	$\frac{7+18\alpha_f}{16}$	$\frac{-7+14\alpha_f}{32}$	$\frac{1-2\alpha_f}{16}$	$\frac{-1+2\alpha_f}{128}$

The second scheme of interest is the optimized fourth-order DRP scheme designed by Bogey and Bailly [44], denoted FDo11p. It uses a symmetric stencil with 11 points and its expression is:

$$u'_i = \frac{1}{\Delta x}\sum_{l=-5}^{5} a_l u_{i+l}. \tag{9.4}$$

The scheme is stabilized by an optimized sixth-order filter denoted SFo11p [167] which is defined on the same stencil:

$$u^f_i = u_i - \sigma_d \sum_{l=-5}^{5} d_l u_{i+l}. \tag{9.5}$$

The optimized coefficients for the FDo11p scheme (a_l) and for the SFo11p filter (d_l) are summarized in Tab. 9.2. σ_d in (9.5) is generally chosen such that $\sigma_d = 1.0$ [167]. To illustrate the need of high-order schemes, the first- and second-order upwind scheme were added to this study. The first-order upwind scheme (UP1) evaluates the spatial first derivative with:

$$u'_i = \frac{u_i - u_{i-1}}{\Delta x}. \tag{9.6}$$

The second-order upwind scheme (UP2) evaluates the spatial first derivative with:

$$u'_i = \frac{3u_i - 4u_{i-1} + u_{i-2}}{2\Delta x}. \tag{9.7}$$

The full space and time integration is as follows. First, the spatial derivative u'_i is computed thanks to the numerical scheme (FDo11p or CS6). Then, the standard Runge-Kutta time integration is performed and finally, the filter (SFo11p or CF8) is applied to the solution.

Tab. 9.2. *FDo11p scheme and SFo11p filter coefficients.*

(a) *Coefficients of the fourth-order scheme FDo11p using 11 points with $a_{-j} = -a_j$ and $a_0 = 0$.*

a_1	0.872756993962
a_2	−0.286511173973
a_3	0.090320001280
a_4	−0.020779405824
a_5	0.002484594688

(b) *Coefficients of the sixth-order filter SFo11p using 11 points with $d_{-j} = d_j$.*

d_0	0.234810479761700
d_1	−0.199250131285813
d_2	0.120198310245186
d_3	−0.049303775636020
d_4	0.012396449873964
d_5	−0.001446093078167

9.2.2 Spectral analysis

The principle of the spectral analysis consists in comparing analytical and numerical derivatives for a harmonic function $u(x) = \exp(jkx)$ with $j^2 = -1$ and k the constant wavenumber. The exact derivative of u is $u'(x) = jk\exp(jkx) = jku(x)$ and the wavenumber can be defined *a posteriori* by

$$k = \frac{u'(x)}{ju(x)}. \tag{9.8}$$

Using the discretized value of u at a point i denoted $u_i = \exp(ji\,k\Delta x)$, the numerical derivative u'_i is computed with the scheme and the ratio of discrete quantities $u'_i/(ju_i)$ can be also obtained. This complex-valued ratio represents the modification of the wavenumber due to numerics and is called the modified wavenumber $k_m = u'_i/(ju_i)$. The dispersive behavior appears if the real part of k_m and the exact wavenumber k differ, while amplification (resp. dissipation) occurs when the imaginary part of k_m is larger (resp. lower) than 0. Moreover, due to the sampling theorem of Nyquist-Shannon [168], at least two points are mandatory to discretize a wave and the wavenumber behaves in $[0, \pi]$.

Both CS6 and FDo11p schemes are centered and thus not dissipative. On the contrary, the filters only introduce dissipation and do not create any dispersion. By applying a spatial Fourier transform to (9.2) and (9.4), the modified wavenumber k_m is obtained for the Lele's sixth-order compact schemes CS6 in (9.9) and for the fourth-order explicit scheme FDo11p in (9.10), respectively:

$$[k_m\Delta x]_{\text{CS6}} = \frac{\frac{14}{9}\sin(k\Delta x) + \frac{1}{18}\sin(2k\Delta x)}{1 + \frac{2}{3}\cos(k\Delta x)}, \tag{9.9}$$

$$[k_m\Delta x]_{\text{FDo11p}} = 2\sum_{l=1}^{5} a_j \sin(l\,k\Delta x)\,. \tag{9.10}$$

A perfect scheme recovers exactly the theoretical wavenumber $k_m = k$ and dispersion is measured either by $k_m\Delta x$ defined from $k\Delta x$, or by the difference $|k\Delta x - k_m\Delta x|$. The modified wavenumber k_m is obtained for the first-order upwind scheme UP1 in (9.11) and for the second-order upwind scheme UP2 in (9.12), respectively:

$$[k_m\Delta x]_{UP1} = \sin(k\Delta x) + j\left[\cos(k\Delta x) - 1\right], \tag{9.11}$$

$$[k_m\Delta x]_{UP2} = \left[2\sin(k\Delta x) - \frac{1}{2}\sin(2k\Delta x)\right] + j\left[2\cos(k\Delta x) - \frac{1}{2}\cos(2k\Delta x) - \frac{3}{2}\right]. \tag{9.12}$$

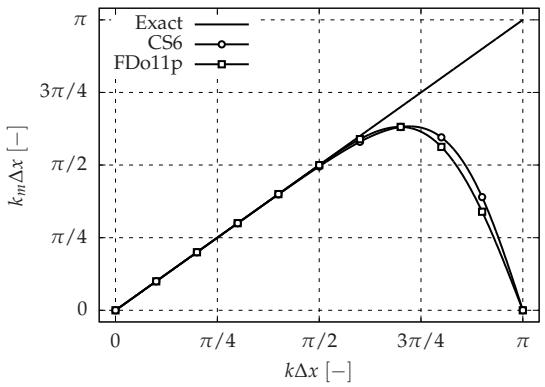
The same Fourier transform is now applied to the filters defined in (9.3) and (9.5). Here, the analysis is immediately performed by comparing the unfiltered data with the filtered ones. On

the contrary, the filters only introduce dissipation and do not create any dispersion. When a Fourier mode is considered as an initial solution, the dissipation rates $\mathcal{F} = u_i^f / u_i$ obtained for the eighth-order compact filter CF8 and for the sixth-order explicit filter SFo11p are defined in (9.13) and in (9.14) respectively:

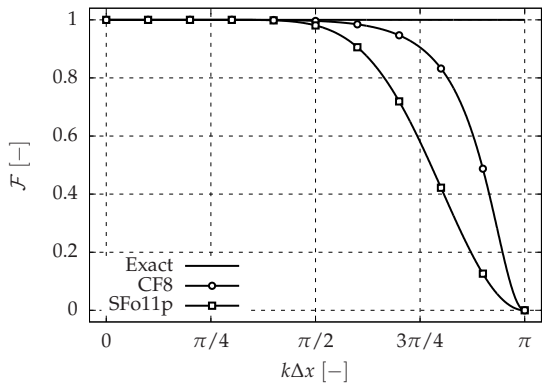
$$[\mathcal{F}(k\Delta x)]_{\text{CF8}} = \frac{1}{1 + 2\alpha_f \cos(k\Delta x)} \sum_{l=0}^{4} b_l \cos(l\, k\Delta x), \tag{9.13}$$

$$[\mathcal{F}(k\Delta x)]_{\text{SFo11p}} = 1 - \sigma_d d_0 - 2\sigma_d \sum_{l=1}^{5} d_l \cos(l\, k\Delta x). \tag{9.14}$$

A filter amplifies (resp. dissipates) when $\mathcal{F}$ is larger (resp. lower) than 1. Dispersion and dissipation for both schemes and their filters are shown in Fig. 9.1a and Fig. 9.1b respectively. To draw a comparison, dispersion and dissipation for both upwind schemes are shown in Fig. 9.2a and Fig. 9.2b respectively. For wavenumbers below $\pi/2$ (equivalent to four points per wavelength), numerical dispersion (resp. dissipation) fits very well with the exact dispersion (resp. dissipation). In order to analyze more accurately the difference between numerics and theory, dispersion and dissipation deviations are also shown in Fig. 9.3a, Fig. 9.3b, Fig. 9.4a and Fig. 9.4b.

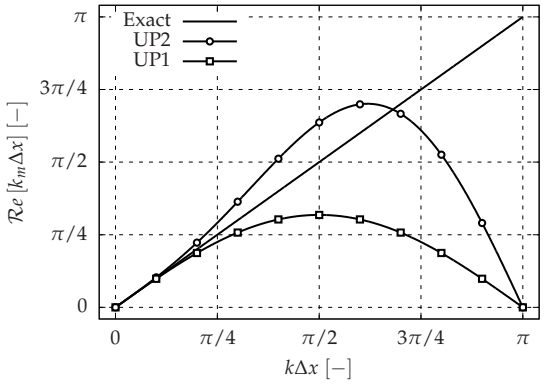


(a) *Modified wavenumber for the sixth-order compact schemes CS6 and the fourth-order explicit schemes using 11 points FDo11p.*

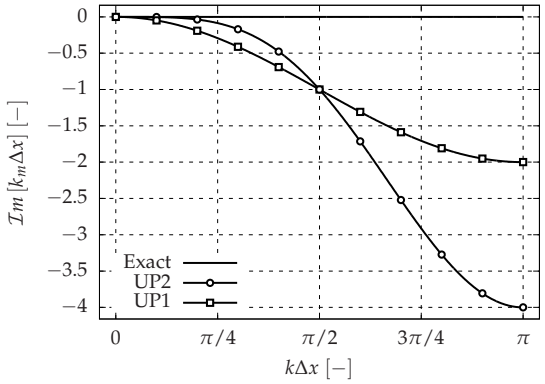


(b) *Dissipation rate for the eighth-order compact filter CF8 and the sixth-order explicit filter using 11 points with $\sigma_d = 1.0$ SFo11p.*

Fig. 9.1. *Spectral analysis of standard schemes used in Computational AeroAcoustics (CAA).*

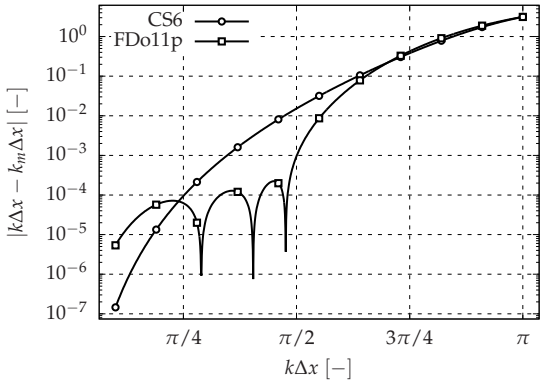


(a) *Real part of the modified wavenumber for the second-order upwind scheme UP2 and the first-order upwind scheme UP1.*

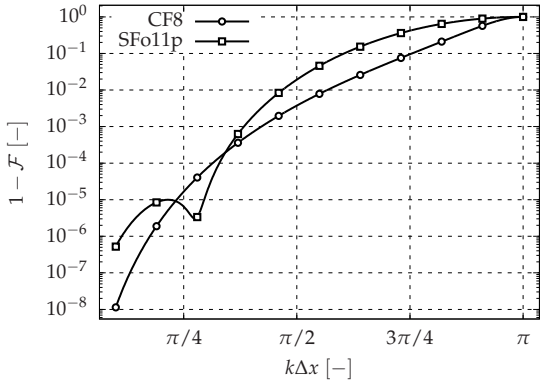


(b) *Imaginary part of the modified wavenumber for the second-order upwind scheme UP2 and the first-order upwind scheme UP1.*

Fig. 9.2. *Spectral analysis of the second-order upwind scheme UP2 and the first-order upwind scheme UP1.*

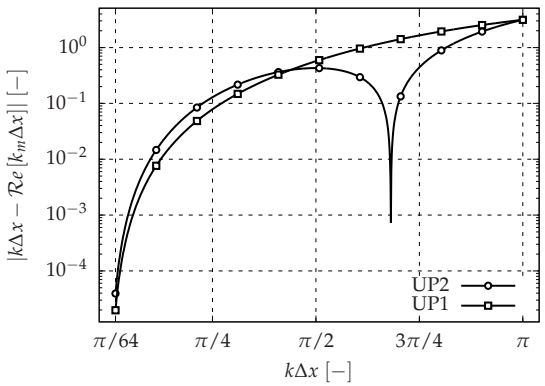


(a) *Dispersion error for the sixth-order compact schemes CS6 and the fourth-order explicit schemes using 11 points FDo11p in logarithmic scale.*

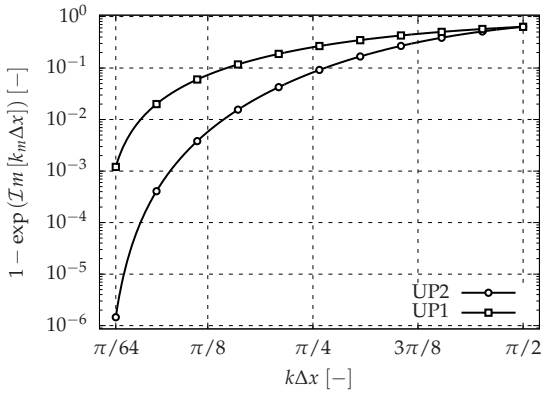


(b) *Dissipation error for the eighth-order compact filter CF8 and the sixth-order explicit filter using 11 points with $\sigma_d = 1.0$ SFo11p in logarithmic scale.*

Fig. 9.3. *Spectral analysis of standard schemes used in Computational AeroAcoustics (CAA) in logarithmic scale to focus on differences between numerics and theory.*



(a) *Dispersion error for the second-order upwind scheme UP2 and the first-order upwind scheme UP1.*



(b) *Dissipation error for the second-order upwind scheme UP2 and the first-order upwind scheme UP1.*

Fig. 9.4. *Spectral analysis of the second-order upwind scheme UP2 and the first-order upwind scheme UP1.*

9.2.3 Points Per Wavelength (PPW)

The number of points per wavelength is representative of the resolution power of any numerical scheme. The criteria proposed by Bogey and Bailly [44] are chosen to measure the maximum wavenumber properly (resp. accurately), defined with subscript p (resp. with subscript a) and calculated for both dispersion and dissipation (superscript ϕ and E respectively). The four criteria are expressed as an optimization problem. For the dispersion, the optimization problems are:

$$\text{Look for } k_p^\phi \Delta x \text{ solution of: } \max_{k\Delta x} |k\Delta x - k_m \Delta x| \leqslant 5\pi \times 10^{-4},$$

$$\text{Look for } k_a^\phi \Delta x \text{ solution of: } \max_{k\Delta x} |k\Delta x - k_m \Delta x| \leqslant 5\pi \times 10^{-5}.$$

Similarly, the optimization problems for dissipation are:

$$\text{Look for } k_p^E \Delta x \text{ solution of: } \max_{k\Delta x} |1 - \mathcal{F}(k\Delta x)| \leqslant 2.5 \times 10^{-3},$$

$$\text{Look for } k_a^E \Delta x \text{ solution of: } \max_{k\Delta x} |1 - \mathcal{F}(k\Delta x)| \leqslant 2.5 \times 10^{-4}.$$

Tab. 9.3a and Tab. 9.3b summarize the number of points per wavelength for both schemes and filters. These values are the same as those proposed by their respective authors, which shows that our analysis is in agreement with previous works. To draw a comparison, Tab. 9.4a and Tab. 9.4b summarize the number of points per wavelength for both upwind schemes.

Tab. 9.3. *Number of points per wavelength (PPW) for the centered CS6 and FDo11p schemes and their corresponding filters CF8 and SFo11p.*

(a) *Accuracy in term of dispersion measured by the scheme.*

	$\lambda_p^\phi/\Delta x$	$\lambda_a^\phi/\Delta x$
CS6	5.42	7.45
FDo11p	3.93	4.65

(b) *Accuracy in term of dissipation measured by the filter.*

	$\lambda_p^E/\Delta x$	$\lambda_a^E/\Delta x$
CF8	4.21	5.66
SFo11p	4.85	5.76

Tab. 9.4. *Number of points per wavelength (PPW) for the first-order upwind UP1 and second-order upwind UP2 scheme.*

(a) *Accuracy in term of dispersion.*

	$\lambda_p^\phi/\Delta x$	$\lambda_a^\phi/\Delta x$
UP1	29.72	64.07
UP2	37.35	80.68

(b) *Accuracy in term of dissipation.*

	$\lambda_p^E/\Delta x$	$\lambda_a^E/\Delta x$
UP1	88.78	280.97
UP2	19.78	35.29

Up to now, it should be noticed that the time integration was not taken into account yet. It means that the time integration discretization has no effect on the equivalent wavenumber for a given scheme / filter. This is only true if the time integration is performed exactly. Moreover, the CF8 filter leads to a slightly better resolution power than the SFo11p filter. Finally, it seems that the FDo11p scheme is better than the CS6 scheme in terms of dispersion.

9.3 Effect of time integration on the spectral analysis

As an introduction, we briefly explain how the space-time spectral analysis is performed. This advanced method is then applied to two Finite Difference schemes. The results obtained in this section will notably play an important role later for purpose of comparison with the new results shown in Sec. 9.4.

9.3.1 Time discretization

The advection equation (9.1) is now time-marched using a standard explicit time integration procedure based on the Runge-Kutta Method. In the whole chapter, the low-storage DRP second-order six-stage Runge-Kutta (RKo6s) scheme of Bogey and Bailly [44] is chosen. For completeness, the Runge-Kutta coefficients are given in Tab. 9.5.

Tab. 9.5. *Coefficients of the low-storage second-order six-stage Runge-Kutta (RKo6s).*

γ_1	1.000000000000
γ_2	0.500000000000
γ_3	0.165919771368
γ_4	0.040919732041
γ_5	0.007555704391
γ_6	0.000891421261

The full space and time integration is as follows. First, the spatial derivative u'_i is computed thanks to the numerical scheme (either FDo11p or CS6). Then, the standard Runge-Kutta time integration is performed and finally, the filter (either SFo11p or CF8) is applied to the updated solution.

The analysis being performed in both space and time, $u(x,t)$ is discretized by $u_i^n = u(i\Delta x, n\Delta t)$, where the index i (resp. n) refers to space (resp. time) position and Δt to the time step. The fully discrete space-time scheme then reads:

$$u_i^{n+1} = u_i^n + \sum_{l=1}^{6} \gamma_l \left(-c\Delta t \mathbb{D}(u_i^n)\right)^l, \tag{9.15}$$

where $\mathbb{D}$ is the derivative operator introduced in (9.1).

Introducing the Courant-Friedrichs-Lewy -CFL- number defined by $\nu = c\Delta t/\Delta x$ and the discretized normal mode

$$u_i^n = \exp\left(-jn\,\omega\Delta t + ji\,k\Delta x\right)$$

into (9.15) and applying the filter transfer function, the expression of the dispersion relation of the space-time scheme with filtering is:

$$\exp\left(-j\omega\Delta t\right) = \mathcal{F}\left(k\Delta x\right)\left(1 + \sum_{l=1}^{6} \gamma_l \left(-j\nu\, k_m \Delta x\right)^l\right). \tag{9.16}$$

(9.16) contains two new quantities. k_m represents the modification of the spatial mode k when the spatial scheme is applied. Moreover, $\mathcal{F}(k\Delta x)$ represents the transfer function associated to either CF8 or SFo11p filters. The analytical expressions for $\mathcal{F}$ and the analytical relations between k_m and k are given in Sec. 9.2.2.

Such a dispersion relation must be compared with the exact one $\omega = kc$. The corresponding dimensionless dispersion relation is $\omega\Delta t = \nu k\Delta x$. Let us now introduce:

$$\mathcal{G}\left(k\Delta x\right) = \mathcal{F}\left(k\Delta x\right)\left(1 + \sum_{l=1}^{6} \gamma_l \left(-j\nu\, k_m \Delta x\right)^l\right), \tag{9.17}$$

$$\varphi = -\arg\left(\mathcal{G}\right)/\nu \in]-\pi, \pi], \tag{9.18}$$

$$\rho = |\mathcal{G}|. \tag{9.19}$$

For a non dispersive scheme, φ is equal to $\omega\Delta t/\nu = k\Delta x$. For a non dissipative scheme, ρ is equal to 1. $\rho > 1$ gives amplification, whereas $\rho < 1$ gives dissipation. Such a coupled space-time approach has been previously introduced in [169, 170, 171].

Remark: The quantities ρ and φ are defined for one iteration of the time integration process.

9.3.2 Stability analysis based on CFL criterion

The space-time discretization is stable under a CFL condition. Before performing the Fourier analysis, the maximum CFL number for stable computation is computed. Starting from (9.15), the iterative process to find u_i^n is convergent if ρ is strictly lower than 1. Looking for the largest time step to maintain ρ strictly lower than 1 for all $k\Delta x$ is an optimization problem and the stability limits (that we obtained) are summarized in Tab. 9.6. Stability limits are also given for both upwind schemes in Tab. 9.7.

Tab. 9.6. *CFL stability for the RKo6s-SFo11p-FDo11p and RKo6s-CF8-CS6 combinations.*

Runge-Kutta	Filter	Scheme	ν
RKo6s	SFo11p	FDo11p	2.053740
RKo6s	CF8	CS6	1.997980

Tab. 9.7. *CFL stability for the RKo6s-UP1 and RKo6s-UP2 combinations.*

Runge-Kutta	Scheme	ν
RKo6s	UP1	2.077149
RKo6s	UP2	1.038574

9.3.3 Spectral analysis

Three CFL numbers (0.1, 1.0 and 1.9, respectively) are chosen in agreement with the stability constraints in Tab. 9.6 and both dispersion and dissipation are estimated from (9.17). Numerical results are shown in Fig. 9.5 and Fig. 9.6 for dispersion and dissipation, respectively. These curves show that the spectral behavior changes with the CFL number and as highlighted in the next section, this change in spectral behavior has a strong impact on the number of grid points per wavelength.

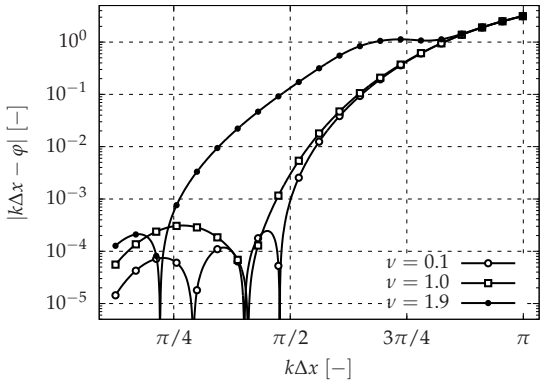
9.3.4 Number of Points Per Wavelength (PPW)

The definition of the number of points per wavelength follows the approach introduced in Sec. 9.2.3. For the coupled space-time analysis, the results may now depend on the CFL number ν. The results obtained for the considered schemes are summarized in Tab. 9.8 and Tab. 9.9. It can be found that, when the CFL number ν tends to 0, the relative error on dispersion and dissipation due to the time integration procedure decreases and the (standard) results for the (standard) spatial spectral analysis are obviously recovered. In terms of dispersion, the required number of points per wavelength does not necessary increase when the CFL number ν increases since the dispersion curves are non-monotonic. In contrast, in terms of dissipation, the required number of points per wavelength increases when the CFL number ν increases.

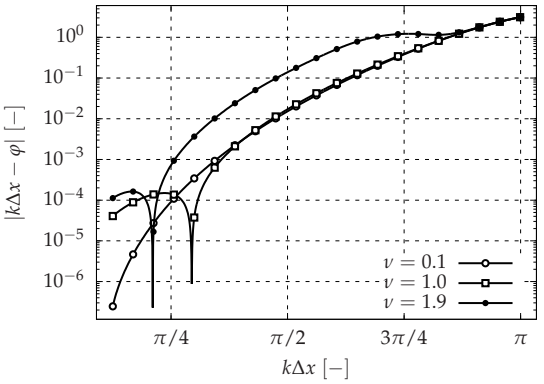
9.3.5 Summary

It has been shown that the time integration procedure has an effect on the spectral behavior of the scheme. Of course, standard results are recovered when the CFL number tends to 0. For the CS6 compact scheme, accounting for the time integration procedure does not really change the required number of points per wavelength up to $\nu = 1$. However, the situation differs for the DRP scheme FDo11p since the number of points per wavelength must be increased. Both filters CF8 and SFo11p are not specifically sensitive to the CFL number ν up to $\nu = 1$. As a consequence, this analysis shows that the coupled space-time approach is a prerequisite to deduce the number of grid points per wavelength.

Moreover, a general method to study the spectral behavior of space-time discretization has been recalled. As shown next in Sec. 9.4, such a coupled approach is mandatory to describe the spectral behavior of spectral discontinuous techniques.

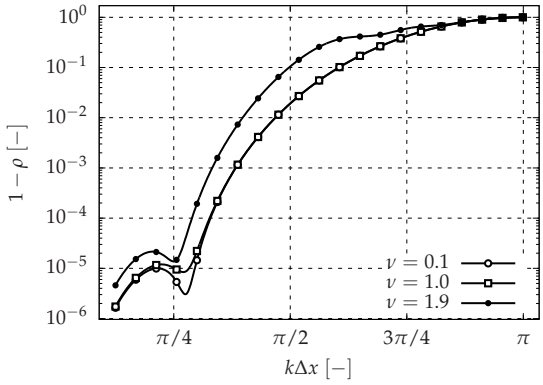


(a) *Dispersion error using RKo6s-SFo11p-FDo11p combination.*

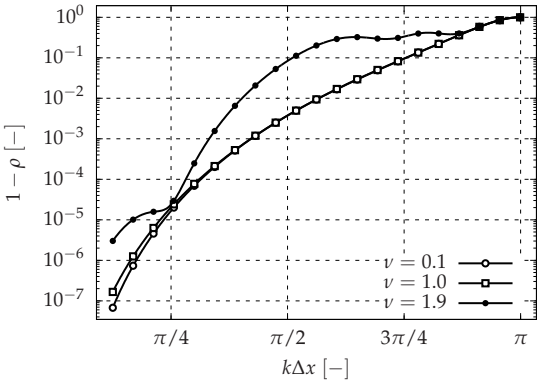


(b) *Dispersion error using RKo6s-CF8-CS6 combination.*

Fig. 9.5. *Effect of time integration on dispersion for the proposed FD schemes.*



(a) *Dissipation error using RKo6s-SFo11p-FDo11p combination.*



(b) *Dissipation error using RKo6s-CF8-CS6 combination.*

Fig. 9.6. *Effect of time integration on dissipation for the proposed FD schemes.*

Tab. 9.8. *Number of points per wavelength for the space-time coupled approach using the RKo6s-SFo11p-FDo11p combination.*

ν	$\lambda_p^\phi/\Delta x$	$\lambda_a^\phi/\Delta x$	$\lambda_p^E/\Delta x$	$\lambda_a^E/\Delta x$
0.01	3.93	4.65	4.85	5.76
0.1	3.93	4.68	4.85	5.76
0.5	3.89	5.26	4.85	5.76
0.7	3.90	9.64	4.85	5.76
0.9	4.01	10.75	4.85	5.77
1.0	4.15	11.23	4.85	5.77
1.5	5.95	13.28	4.94	5.80
1.9	7.27	14.50	5.63	6.57

Tab. 9.9. *Number of points per wavelength for the space-time coupled approach using the RKo6s-CF8-CS6 combination.*

ν	$\lambda_p^\phi/\Delta x$	$\lambda_a^\phi/\Delta x$	$\lambda_p^E/\Delta x$	$\lambda_a^E/\Delta x$
0.01	5.42	7.45	4.21	5.66
0.1	5.42	7.42	4.21	5.66
0.5	5.31	6.86	4.22	5.67
0.7	5.27	6.54	4.22	5.68
0.9	5.28	6.38	4.22	5.69
1.0	5.33	6.39	4.22	5.70
1.5	6.20	11.84	4.61	5.77
1.9	7.38	12.96	5.61	6.66

9.4 Spectral analysis of the Spectral Difference method

This section is devoted to the Fourier analysis of the SD method. As before, the analysis is performed on the advection equation (9.1).

9.4.1 Description of the Spectral Difference method

The spectral difference method solves in any cell the strong form of the equation per direction, as in the FD approach. The concept of the SD approach is simple: if the solution u evolves as a polynomial of degree p, the divergence of the flux must also be a polynomial of degree p and therefore the flux polynomial must be a polynomial of degree $p+1$. Instead of defining any polynomial by its monomial coefficients, polynomials are defined by their values on a set of points and by Lagrange interpolation. We introduce the set of $p+1$ points called *solution points* denoted as $(SP_l)_{1\leq l\leq p+1}$ and the $p+2$ points called *flux points* denoted as $(FP_l)_{1\leq l\leq p+2}$. In the following, the explanation related to the position of the degrees of freedom is performed in $[-1,1]$, since any 1D cell can be cast into $[-1,1]$ by an isoparametric transformation.

The unknown polynomial for u is totally defined by its values at the *solution points* and here, the solution points are selected as the roots of the Chebyshev polynomial (of the first kind) of degree p defined by:

$$SP_l = -\cos\left(\frac{2l-1}{2p+2}\pi\right) \text{ for } 1 \leqslant l \leqslant p+1. \tag{9.20}$$

It is shown in [161] that the SD approach is independent of the solution point location and our choice is given for completeness.

The flux polynomial is defined by $p+2$ *flux points* (1 point more than the solution). Following Kopriva and Kolias [143], two flux points are located at the segment boundaries and any other p flux point is located between two solution points. The approach is staggered. In this chapter, we choose the p flux points as the roots of the Legendre polynomials of degree p and the two segment end points, as Huynh [152]. We found that this choice is the best one for stability and in particular, the other standard choice (Chebyshev-Gauss-Lobatto) is proved to be weakly unstable for $p > 2$ [161, 172].

Both solution and flux polynomials are based on the Lagrange polynomials using the solution points SP_l and the flux points FP_l respectively. The Lagrange polynomials $\mathcal{L}_l$ of degree p based on the solution point index l is:

$$\mathcal{L}_l(x) = \prod_{\substack{r=1\\ r\neq l}}^{p+1} \frac{x - SP_r}{SP_l - SP_r} \tag{9.21}$$

and the solution in cell i $u_i(x,t)$ can be expanded over this basis as:

$$u_i(x,t) = \sum_{l=1}^{p+1} u_i(SP_l,t)\ \mathcal{L}_l(x). \tag{9.22}$$

In the same way, the Lagrange polynomials $\mathcal{T}_l$ of order $p+1$ based on the flux point index l is:

$$\mathcal{T}_l(x) = \prod_{\substack{r=1 \\ r\neq l}}^{p+2} \frac{x - FP_r}{FP_l - FP_r} \tag{9.23}$$

and the flux function polynomial $f_i(x,t)$ in cell i ($f_i(x,t) = cu_i(x,t)$ for the advection equation) can be expanded over the basis as:

$$f_i(x,t) = \sum_{l=1}^{p+2} f_i(FP_l,t)\,\mathcal{T}_l(x). \tag{9.24}$$

The SD procedure is now as follows. First, the solution is extrapolated at the flux point locations using (9.22). At the interior flux points, the flux is computed immediately from the solution. The flux on border flux points is the solution of a Riemann problem since the two extrapolated quantities (one at each side) can differ. Finally, the divergence of the flux polynomial is evaluated at the solution point locations by differentiation of (9.24) as:

$$\frac{\partial f_i}{\partial x}(x,t) = \sum_{l=1}^{p+2} f_i(FP_l,t)\,\mathcal{T}_l'(x), \quad \text{where } \mathcal{T}_l'(x) = \frac{d\mathcal{T}_l}{dx}(x). \tag{9.25}$$

9.4.2 Matrix form of the SD procedure

Since the polynomials are defined on a Lagrange basis, any polynomial is defined by its values at the control points (either solution points or flux points). We introduce $\overline{\boldsymbol{U}}_i(t)$, the column vector of size $p+1$ (whose components are the solutions at solution points) as:

$$\overline{\boldsymbol{U}}_i(t) = \left[u_i(SP_l,t)\right]_{1\leqslant l\leqslant p+1}^{\top}, \tag{9.26}$$

where $^{\top}$ means the transpose operation.

The first step is the extrapolation of the solution at the flux points using the form of polynomials, leading to the column vector $\overline{\boldsymbol{V}}_i(t)$ of size $p+2$ defined by:

$$\overline{\boldsymbol{V}}_i(t) = \left[u_i(FP_l,t)\right]_{1\leqslant l\leqslant p+2}^{\top} = \left[\sum_{r=1}^{p+1} u_i(SP_r,t)\,\mathcal{L}_r(FP_l)\right]_{1\leqslant l\leqslant p+2}^{\top}. \tag{9.27}$$

Since a Riemann problem must be solved on the cell boundaries, informations from neighboring cells $i-1$ and $i+1$ are required. Solutions at solution points $\widetilde{\boldsymbol{U}}_i(t)$ and solutions at flux points $\widetilde{\boldsymbol{V}}_i(t)$ including solutions on the border flux points from adjacent cells are now defined as column vectors of size $3\times(p+1)$ and $p+4$, respectively :

$$\widetilde{\boldsymbol{U}}_i(t) = \begin{bmatrix} \overline{\boldsymbol{U}}_{i-1}(t) \\ \overline{\boldsymbol{U}}_i(t) \\ \overline{\boldsymbol{U}}_{i+1}(t) \end{bmatrix}, \tag{9.28}$$

$$\widetilde{\boldsymbol{V}}_i(t) = \begin{bmatrix} u_{i-1}\left(FP_{p+2},t\right) \\ \overline{\boldsymbol{V}}_i(t) \\ u_{i+1}(FP_1,t) \end{bmatrix}. \tag{9.29}$$

Defining $O_{m,n}$ as the zero matrix of dimension $m \times n$ and I_n the identity matrix of size n, the extrapolation matrix E of size $(p+4) \times (3 \times (p+1))$ such that $\widetilde{\boldsymbol{V}}_i(t) = E\widetilde{\boldsymbol{U}}_i(t)$ reads as:

$$E = \begin{bmatrix} \left[\mathcal{L}_l\left(FP_{p+2}\right)\right]_{1\leqslant l\leqslant p+1} & O_{1,p+1} & O_{1,p+1} \\ O_{1,p+1} & \left[\mathcal{L}_l\left(FP_1\right)\right]_{1\leqslant l\leqslant p+1} & O_{1,p+1} \\ O_{1,p+1} & \left[\mathcal{L}_l\left(FP_2\right)\right]_{1\leqslant l\leqslant p+1} & O_{1,p+1} \\ \vdots & \vdots & \vdots \\ O_{1,p+1} & \left[\mathcal{L}_l\left(FP_{p+2}\right)\right]_{1\leqslant l\leqslant p+1} & O_{1,p+1} \\ O_{1,p+1} & O_{1,p+1} & \left[\mathcal{L}_l\left(FP_1\right)\right]_{1\leqslant l\leqslant p+1} \end{bmatrix}. \tag{9.30}$$

The second step consists of computing the flux. Here, the Riemann problem is solved using the (upwind) Godunov scheme:

$$\mathcal{F}_{Riemann}(u_L, u_R) = c\left(\frac{1+sign(c)}{2}u_L + \frac{1-sign(c)}{2}u_R\right), \tag{9.31}$$

where $sign(c) = c/|c|$. The computation of the flux at the flux points can be defined by the matrix F of size $(p+2) \times (p+4)$:

$$F = c\begin{bmatrix} \frac{1+sign(c)}{2} & \frac{1-sign(c)}{2} & O_{1,p} & 0 & 0 \\ O_{p,1} & O_{p,1} & I_p & O_{p,1} & O_{p,1} \\ 0 & 0 & O_{1,p} & \frac{1+sign(c)}{2} & \frac{1-sign(c)}{2} \end{bmatrix}. \tag{9.32}$$

For the last step, the flux polynomial is differentiated and its divergence is computed at the solution points. The derivative matrix D of size $(p+1) \times (p+2)$ is then

$$D = \left[\mathcal{T}_l'\left(SP_r\right)\right]_{\substack{1\leqslant r\leqslant p+1 \\ 1\leqslant l\leqslant p+2}}.$$

The overall process for computing the divergence term from the solution points can now be written in the following compact matrix form as:

$$M_1 = DFE, \tag{9.33}$$

with M_1 a matrix of size $(p+1) \times (3(p+1))$.

9.4.3 Matrix form for the spectral analysis

The generating pattern for a one-dimensional problem is given by one cell, so let us introduce the discretized normal mode within this period $\overline{\boldsymbol{U}}_i(t) = \widehat{\boldsymbol{U}}_i(t)\exp(ji\,k\Delta x)$ such that $\widetilde{\boldsymbol{U}}_i(t) = L\widehat{\boldsymbol{U}}_i(t)$. Introducing the complex-valued matrix L of size $3(p+1) \times (p+1)$:

$$L = \begin{bmatrix} \exp(-j\,k\Delta x)\, I_{p+1} \\ I_{p+1} \\ \exp(j\,k\Delta x)\, I_{p+1} \end{bmatrix}, \tag{9.34}$$

it comes easily:

$$\frac{\partial \widehat{\boldsymbol{U}}_i(t)}{\partial t} = M\widehat{\boldsymbol{U}}_i(t), \tag{9.35}$$

where $M = -DFEL = -M_1 L$.

9.4.4 Time discretization

(9.35) is a system of linear differential equations whose integration is carried out using the low-storage second-order six-stage Runge-Kutta scheme of Bogey and Bailly [44], as in Sec. 9.3. $\widehat{\boldsymbol{U}}_i(t)$ is discretized by $\widehat{\boldsymbol{U}}_i^n = \widehat{\boldsymbol{U}}_i(n\Delta t)$ and the discrete solution is advanced in time using

$$\widehat{\boldsymbol{U}}_i^{n+1} = \left[I_{p+1} + \sum_{l=1}^{6} \gamma_l \Delta t^l M^l \right] \widehat{\boldsymbol{U}}_i^n = G \widehat{\boldsymbol{U}}_i^n. \tag{9.36}$$

G is a square complex-valued matrix of size $p+1$ which accounts for both space and time integration. G depends on the polynomial degree p, on the wavenumber k and of course on the CFL number $\nu = c\Delta t/\Delta x$. It is important to note that we keep here the standard definition of the CFL number for an advection equation.

9.4.5 Stability analysis based on CFL criterion

The SD method with the Runge-Kutta time integration is stable under a CFL condition, similarly as the FD method. Starting from (9.36), the matrix geometric progression between $\widehat{\boldsymbol{U}}_i^{n+1}$ and $\widehat{\boldsymbol{U}}_i^n$ is convergent if the spectral radius ρ of matrix G is strictly lower than 1. The stability limits found are summarized in Tab. 9.10. To be able to compare with the FD method, let us define a new CFL number as $\widehat{\nu} = (c\Delta t/\Delta x)(p+1) = \nu(p+1)$ with a length scale which corresponds to the mean distance between two adjacent degrees of freedom.

Tab. 9.10. *CFL stability bounds for the SD method with Runge-Kutta time integration (RKo6s).*

p	ν	$\widehat{\nu}$
2	0.542304	1.626913
3	0.337879	1.351515
4	0.233186	1.165928
5	0.172017	1.032102

9.4.6 Application of the Nyquist-Shannon sampling theorem

The Nyquist-Shannon sampling theorem states that at least two points are mandatory to capture a given frequency. Such an approach is routinely applied in standard schemes such as FD. Here, let us consider a vector $\widehat{\boldsymbol{U}}_i^n$ sampled over a cell of size Δx. The sampling frequency f_s is therefore $f_s = 1/\Delta x$. It is assumed that the vector $\overline{\boldsymbol{U}}_i$ represents the normal mode as in Sec. 9.4.3:

$$\overline{\boldsymbol{U}}_i(t,k) = \widehat{\boldsymbol{U}}_i(t) \exp(ji\,k\Delta x). \tag{9.37}$$

Computing now the same kind of relation for a new normal mode with $k' = k + m(2\pi/\Delta x) = k + m(2\pi f_s)$ $(m \in \mathbb{Z})$, one finds:

$$\begin{aligned}
\overline{\boldsymbol{U}}_i(t,k') &= \widehat{\boldsymbol{U}}_i(t) \exp\left(ji\,k'\Delta x\right), \\
&= \widehat{\boldsymbol{U}}_i(t) \exp\left(ji\ (k + 2m\pi/\Delta x)\,\Delta x\right), \\
&= \widehat{\boldsymbol{U}}_i(t) \exp(ji\,k\Delta x + ij\,2m\pi), \\
&= \overline{\boldsymbol{U}}_i(t,k),
\end{aligned}$$

because $im \in \mathbb{Z}$. As a consequence, any normal mode $\overline{\boldsymbol{U}}_i(t, k + 2m\pi/\Delta x)$ with $m \in \mathbb{Z}$ has the *same representation* after sampling at the solution points and we note that, in order to avoid aliasing, $k\Delta x$ should belong to $[0, \pi]$. This is a key point and in all previous studies on spectral accuracy (for DG, SD, SV and FR), such a property was always lost [154, 158, 159, 160, 161, 162, 163]. An illustration of the aliasing phenomenon for $k\Delta x > \pi$ will be shown in Sec. 9.5.

9.4.7 A naive approach for the spectral analysis

(9.36) introduces the transfer matrix G between time steps n and $n+1$:

$$\widehat{\boldsymbol{U}}_i^{n+1} = G\widehat{\boldsymbol{U}}_i^n. \tag{9.38}$$

Introducing a normal mode projected onto the SD basis

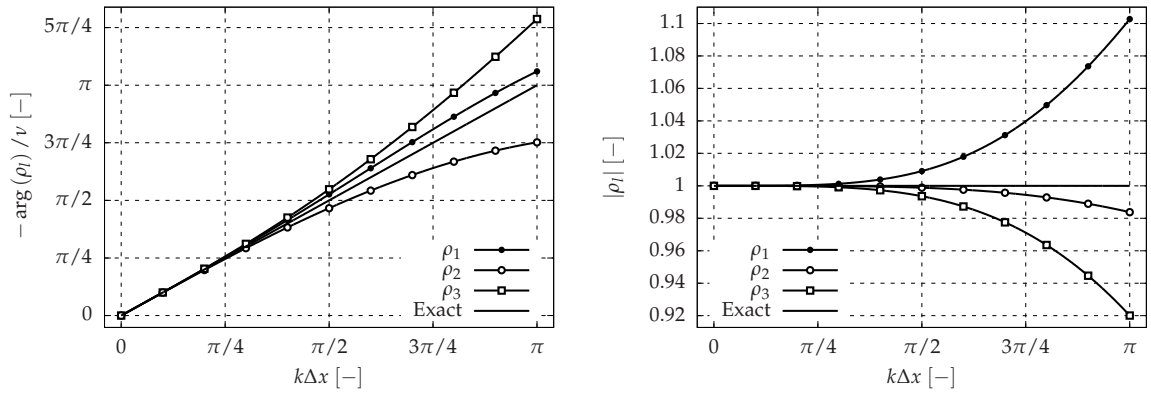
$$\widehat{\boldsymbol{U}}_i^n = \exp\left(-jn\,\omega\Delta t\right)\left[\exp\left(j\,k\left(SP_l + i\Delta x\right)\right)\right]_{1\leqslant l\leqslant p+1}^{\top} \tag{9.39}$$

and injecting it into (9.41), it comes:

$$\exp\left(-j\omega\Delta t\right)\left[\exp\left(jk\left(SP_l + i\Delta x\right)\right)\right]_{1\leqslant l\leqslant p+1}^{\top} = G\left[\exp\left(jk\left(SP_l + i\Delta x\right)\right)\right]_{1\leqslant l\leqslant p+1}^{\top}. \tag{9.40}$$

(9.40) represents a system of relations between the components of two column vectors. The naive approach for the spectral analysis consists of performing the analysis component by component. As a consequence, one can introduce for any solution point l the ratio ρ_l of the corresponding l-th components of the two vectors. ρ_l is a complex number which approximates $\exp\left(-j\omega\Delta t\right)$ for each solution point and such approximation allows to evaluate the dispersion relation for each solution point. $|\rho_l|$ gives the amplification or the dissipation at the l-th solution point: $|\rho_l| > 1$ gives amplification and $|\rho_l| < 1$ dissipation. The argument $\varphi = -\arg\left(\rho_l\right)/\nu$ gives the phase delay or the dispersion. It is of great importance to remind that since $p+1$ solution points were introduced, the spectral behavior is characterized by $p+1$ curves. These curves must be compared with the exact dispersion relation $\omega = kc$ or with its dimensionless expression $\omega\Delta t = \nu k\Delta x$.

Fig. 9.7a and Fig. 9.7b give the dispersion and the amplification for $(p,\nu) = (2, 0.1)$. It numerically seems that the scheme is unstable since $|\rho_1| > 1$ for $k\Delta x > 0$. It is clear in Fig. 9.7b that $|\rho_1| > 1$ for $k\Delta x > \pi/2$.



(a) *Dispersion.* **(b)** *Amplification.*

Fig. 9.7. *Spectral analysis for* $(p,\nu) = (2, 0.1)$*: naive approach.*

However, even if $|\rho_1| > 1$ for $k\Delta x > 0$, the scheme is stable. Fig. 9.8 gives an illustration of the spectral analysis previously achieved. Any black cross defines the position of one solution point and any full black circle represents an initial solution defined at the solution point locations. After time integration (9.41), the new solution is defined at the same solution points using the empty black circles. The new solution is obtained from the initial curve by adding the basic ingredients of numerical schemes: dissipation and dispersion. Thanks to this illustration, it can be seen that at some solution points, the solution is amplified whereas it is dissipated at the other solution points even if the solution is *globally dissipated*. Fig. 9.8 justifies the fact that a purely local analysis, based on values at solution points, is not the right way

to analyze the spectral behavior. The spectral analysis requires a global trend and the naive analysis looses the coupling of the solutions at solution points through the definition of the local polynomial. *In other words, the solution polynomial plays the most important role, not the data used to define it.*

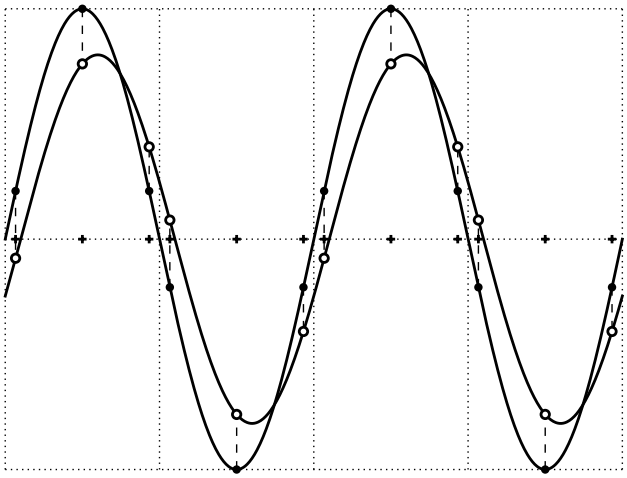


Fig. 9.8. *Illustration of the spectral analysis. The mesh is composed of four cells delimited by vertical dotted lines. Any black cross (+) represents one solution point. The initial solution at one solution point is after time integration at the solution points.*

9.4.8 Matrix Power Method for the spectral analysis

(9.36) introduces the transfer matrix G between time steps n and $n+1$:

$$\widehat{\boldsymbol{u}}_i^{n+1} = G\widehat{\boldsymbol{u}}_i^n. \tag{9.41}$$

and using properties of geometric progressions and (9.41), one obtains, raising G to the power n:

$$\widehat{\boldsymbol{u}}_i^n = G^n\widehat{\boldsymbol{u}}_i^0. \tag{9.42}$$

Carrying out an unsteady computation (for the advection equation) with the Spectral Difference method simply consists of computing the n-th power of the matrix G in order to obtain the solution at the n-th time iteration from the initial solution. The computation of the n-th power of any matrix is at the core of the Matrix Power Method (MPM) also called Power Iteration or Power Method [173, 174]. The main properties of the MPM are summarized below for completeness.

The MPM gives a well-known algorithm for the computation of the spectral radius of a matrix, *i.e.* the eigenvalue with the largest modulus. The proof of the theorem can be found in [173] and it is introduced hereafter since it enables to establish a general result for the spectral analysis of the SD Method. Since (9.41) is the expression of a matrix geometric progression, its behavior is given by the eigenvalue with the largest modulus when n tends to infinity because among all the modes, the considered one has the least dissipation. This is the reason why, even if all the eigenvalues were computed, only the one with the largest modulus was retained according to the MPM.

The Matrix Power Method for SD

G is a $(p+1)\times(p+1)$ matrix with complex coefficients. Consider the eigenvalue problem $G\widehat{\boldsymbol{u}}_i = \lambda\widehat{\boldsymbol{u}}_i$, where $\widehat{\boldsymbol{u}}_i \neq 0$, $\widehat{\boldsymbol{u}}_i \in \mathbb{C}^{(p+1)\times 1}$ and $\lambda \in \mathbb{C}$. $\widehat{\boldsymbol{u}}_i$ is indeed a column vector.

Assumptions: *We assume that the eigenvalue problem* $G\boldsymbol{v}_l = \lambda_l \boldsymbol{v}_l$ *with* $\lambda_l \in \mathbb{C}$ *and* $\boldsymbol{v}_l \in \mathbb{C}^{(p+1)\times 1}$ *admits a complete normalized eigenvector space* $(\boldsymbol{v}_1, \boldsymbol{v}_2, ..., \boldsymbol{v}_{p+1})$ *spanning* $\mathbb{C}^{(p+1)}$ *with corresponding eigenvalues satisfying* $|\lambda_1| < |\lambda_l|$, $l \neq 1$ *and* $|\lambda_l| < \left|\lambda_{p+1}\right|$, $l \neq p+1$.

From now on, let the initial condition $\widehat{\boldsymbol{U}}_i^0$ be a given vector, for which

$$\widehat{\boldsymbol{U}}_i^0 = \sum_{l=1}^{p+1} \alpha_l^{(0)} \boldsymbol{v}_l, \text{ with } \alpha_{p+1}^{(0)} \neq 0, \tag{9.43}$$

and let

$$\forall n \in \{1, 2, ...\}, \ \widehat{\boldsymbol{U}}_i^n = G\widehat{\boldsymbol{U}}_i^{n-1}, \tag{9.44}$$

be the basic recursive sequence. From (9.43) and (9.44), it comes immediately:

$$\widehat{\boldsymbol{U}}_i^n = \sum_{l=1}^{p+1} \alpha_l^{(0)} G^n \boldsymbol{v}_l.$$

Furthermore, $G^n \boldsymbol{v}_l = \lambda_l^n \boldsymbol{v}_l$, for all $l \in \{1, 2, ..., p+1\}$, and as a consequence:

$$\widehat{\boldsymbol{U}}_i^n = \sum_{l=1}^{p+1} \alpha_l^{(0)} \lambda_l^n \boldsymbol{v}_l$$

or equivalently:

$$\widehat{\boldsymbol{U}}_i^n = \alpha_{p+1}^{(0)} \lambda_{p+1}^n \left(\boldsymbol{v}_{p+1} + \sum_{l=1}^{p} \frac{\alpha_l^{(0)}}{\alpha_{p+1}^{(0)}} \left(\frac{\lambda_l}{\lambda_{p+1}} \right)^n \boldsymbol{v}_l \right).$$

Hence, using $\left|\frac{\lambda_l}{\lambda_{p+1}}\right| = 1 - \frac{\left|\lambda_{p+1}\right| - |\lambda_l|}{\left|\lambda_{p+1}\right|}$, it comes:

$$\left\| \boldsymbol{v}_{p+1} - \frac{1}{\alpha_{p+1}^{(0)} \lambda_{p+1}^n} \widehat{\boldsymbol{U}}_i^n \right\|_\infty \leqslant \sum_{l=1}^{p} \left| \frac{\alpha_l^{(0)}}{\alpha_{p+1}^{(0)}} \right| \left(1 - a_{p+1}\right)^n, \tag{9.45}$$

where

$$a_{p+1} = \min_{\substack{l \in \{1,...,p\} \\ l \neq p+1}} \left| \frac{\left|\lambda_{p+1}\right| - |\lambda_l|}{\lambda_{p+1}} \right|.$$

Noticing that

$$\lim_{n \to +\infty} \left(1 - a_{p+1}\right)^n = 0,$$

one can easily deduce from (9.45) that

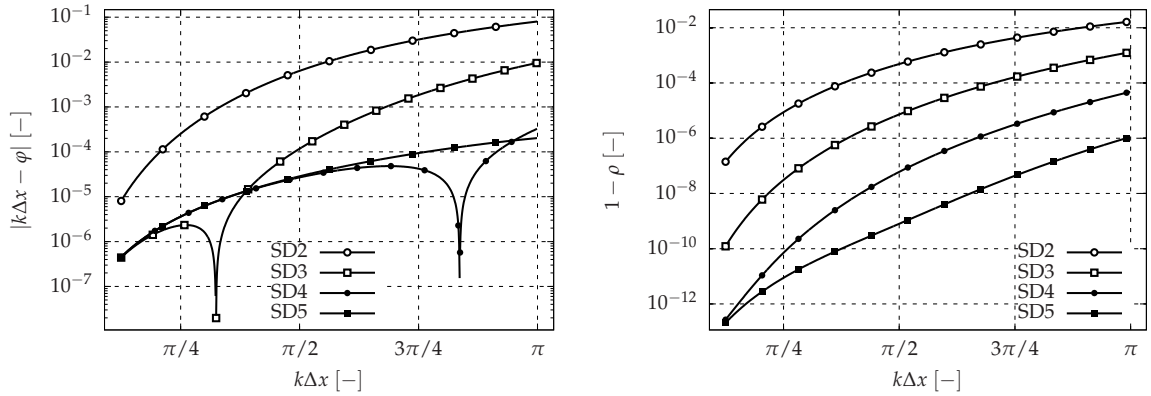
$$\widehat{\boldsymbol{U}}_i^n \underset{n \to +\infty}{\sim} \alpha_{p+1}^{(0)} \lambda_{p+1}^n \boldsymbol{v}_{p+1}. \tag{9.46}$$

So, $\widehat{\boldsymbol{U}}_i^n$ behaves as $\alpha_{p+1}^{(0)} \lambda_{p+1}^n \boldsymbol{v}_{p+1}$ when the number of iterations of the time integration n is large. Moreover, the difference between the true behavior and its asymptotic approximation depends on how the ratios $\left|\lambda_l / \lambda_{p+1}\right|$ for $l \in \{1, ..., p\}$ decay to 0. For a large number of iterations and for any guess $\widehat{U}_i^0$ with $\alpha_{p+1}^{(0)} \neq 0$, $\widehat{U}_i^n$ behaves in the direction of the dominant eigenvector $\boldsymbol{v}_{p+1}$.

Results of the MPM for SD

The Matrix Power Method gives a way to perform the spectral analysis of the Spectral Difference method. The dispersion and the amplification are simply given by the spectral radius $|\lambda_{p+1}|$ of the matrix G. Between the iterations n and $n+1$, the dispersion is given by the argument $\varphi = -\arg\left(\lambda_{p+1}\right)/\nu$ and the amplification is given by $\rho = \left|\lambda_{p+1}\right|$.

Remark: *In all the cases presented in this chapter, we found numerically that the eigenvalue problem $G\boldsymbol{v}_l = \lambda_l \boldsymbol{v}_l$ always satisfied the assumptions given above. We did not succeed in demonstrating mathematically such a result for any value of the polynomial degree p. This is left as a future line of research.*



(a) *Dispersion error.* **(b)** *Dissipation error.*

Fig. 9.9. *Spectral analysis for the RKo6s-SD schemes with CFL condition $\nu = 0.1$: effect of the order of the solution reconstruction.*

Remark: It should be noted that the Matrix Power Method can be applied to any kind of high-order spectral discontinuous method, since the discrete space-time integration can be written in the form of (9.44) for an advection equation.

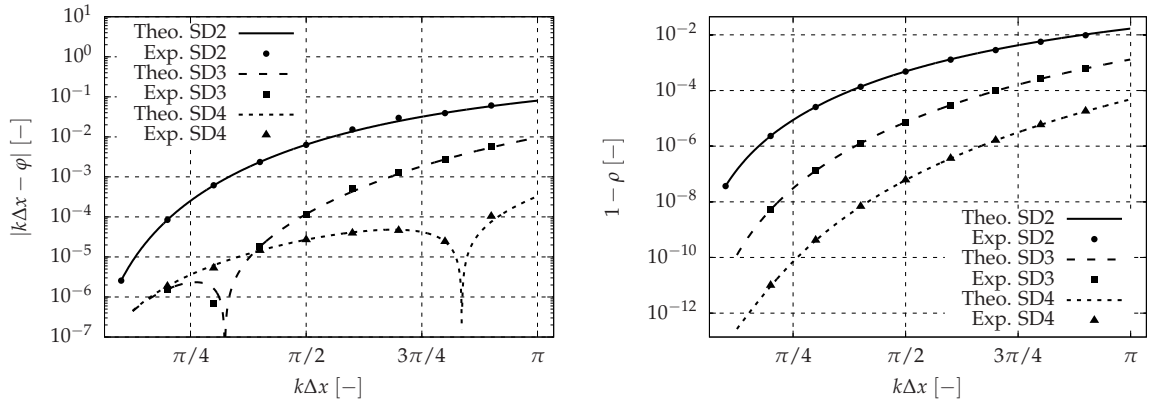
9.4.9 Comparison with numerical results

Numerical solutions in 1D are now computed to support the theoretical results on dispersion and dissipation in Fig. 9.9a and Fig. 9.9b. The advection equation (9.1) with $c = 1.0$ [m/s] is solved using the Spectral Difference method previously mentioned with $\nu = 0.1$. The initial solution is $u_0(x) = \sin(kx)$. The one-dimensional mesh is composed of 40 regular cells with a cell length $\Delta x = 2.0$. The system is closed with periodic boundary conditions. Indeed, numerical computations are performed with $k\Delta x = 0.1\pi n$ with $n \in \mathbb{N}$, $1 \leqslant n \leqslant 9$ for $p = 2$ and $2 \leqslant n \leqslant 9$ for $p = 3$ and $p = 4$. One obtains $2n$ periods on this specific mesh.

The initial solution is transported over a sufficient number of discrete time instants to measure the dissipation and the dispersion. The amplification and the phase shift are identified by a minimization process using the least squares method to solve:

$$\min_{\substack{A\in\mathbb{R}^{+*}\\ \phi\in[0,2\pi[}} \left\| f(x,t) - A\sin(kx+\phi)\right\|_2, \tag{9.47}$$

where $\|\cdot\|_2$ is the standard L^2 norm for functions. It is shown in Fig. 9.10a and Fig. 9.10b that theoretical and numerical behaviors for dispersion and dissipation are in a very good agreement, for $p = 2$, $p = 3$ and $p = 4$ respectively.



(a) *Dispersion error.*

(b) *Dissipation error.*

Fig. 9.10. *Spectral analysis for t he RKo6s-SD schemes with CFL condition $\nu = 0.1$: comparison of theoretical spectral behaviors with numerical solutions and effect of different polynomial orders.*

9.5 Extension to high wavenumbers for the Spectral Difference method

It was shown in Sec. 9.4.6 that the standard spectral analysis, with Fourier modes, cannot be applied to wavenumbers greater than π. This is a consequence of the Nyquist-Shannon sampling theorem. However, the spectral analysis published in the literature allows to capture wavenumbers greater than π [154, 158, 159, 160, 161, 162, 163]. This section is devoted to the analysis of this phenomenon when the wavenumber is larger than π.

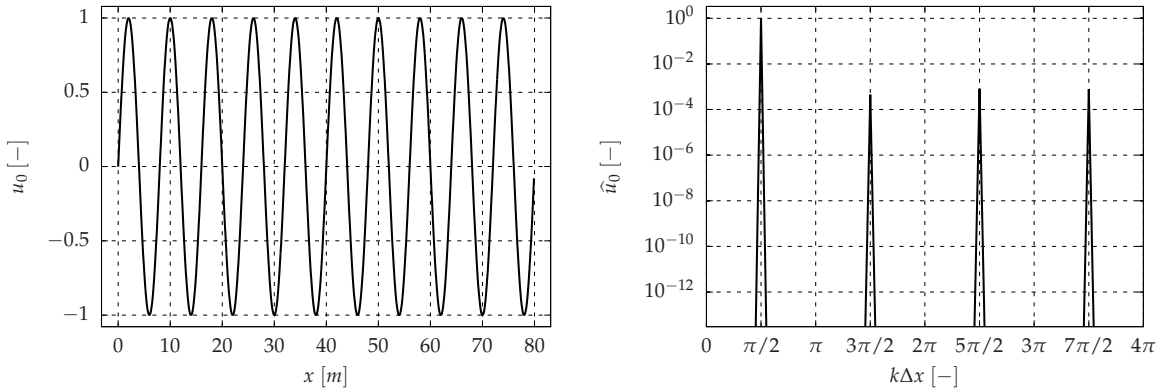
9.5.1 Aliasing and initial solution projection

In this section, we want to analyze the aliasing that occurs with high-order spectral methods. We consider as before an advection equation problem with periodic conditions. For a given Δx, two simulations are performed. For the first one, the wavenumber k is chosen such that $k\Delta x = \pi/2$, while for the second case, the wavenumber k is such that $k\Delta x = 3\pi/2$. For both cases, we select $p = 3$.

For the first computation, the initial, final solutions, and their corresponding Fourier transform, are shown in Fig. 9.11 and Fig. 9.12. The solution is almost conserved and the energy repartition per mode does not change significantly.

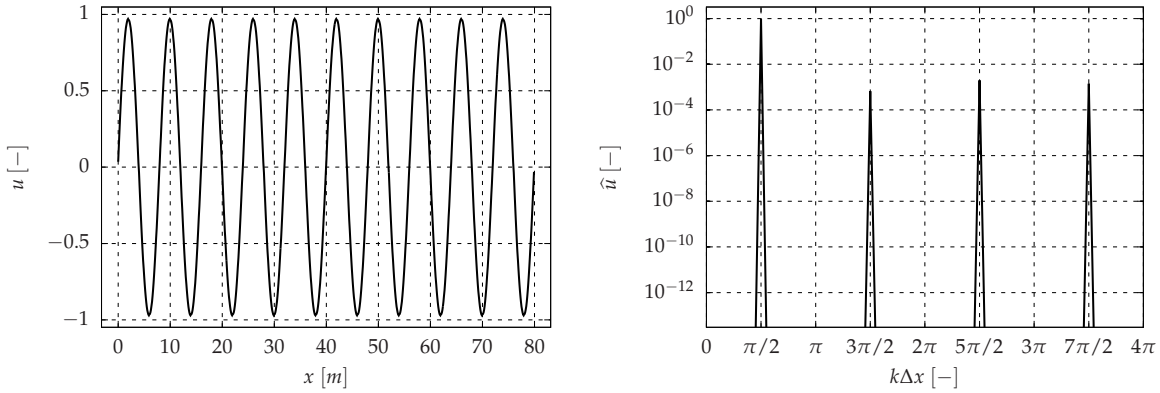
Regarding the second case with $k\Delta x = 3\pi/2$, the initial solution and the associated Fourier spectrum are shown in Fig. 9.13. After many iterations (we do not define the number of iterations here since it is the topic of Sec. 9.5.3 and Sec. 9.5.4), the final solution and the associated spectrum are shown in Fig. 9.14. We remark that the mode $k\Delta x = \pi/2$ now contains the largest part of the energy, while the mode $k\Delta x = 3\pi/2$ initially had the largest energy. All modes larger than π are damped and it remains essentially the mode associated with $k\Delta x = \pi/2$.

This numerical experiment points out the aliasing phenomenon revealed by our analysis. Such aliasing occurs for any value of p, but the time to see an effect varies with p, the number of iterations, the CFL number and the wavenumber one looks for. Moreover, the standard Fourier spectral analysis can only be applied once the initial wavenumber is unchanged: dissipation and dispersion are defined for a given frequency. Here, since the frequency (with the main part of the energy) changes, it is mandatory to introduce a new way to estimate dissipation and dispersion. In the following, dissipation is expressed using relations defined with energy preserving relations, while dispersion is obtained from a scalar product.



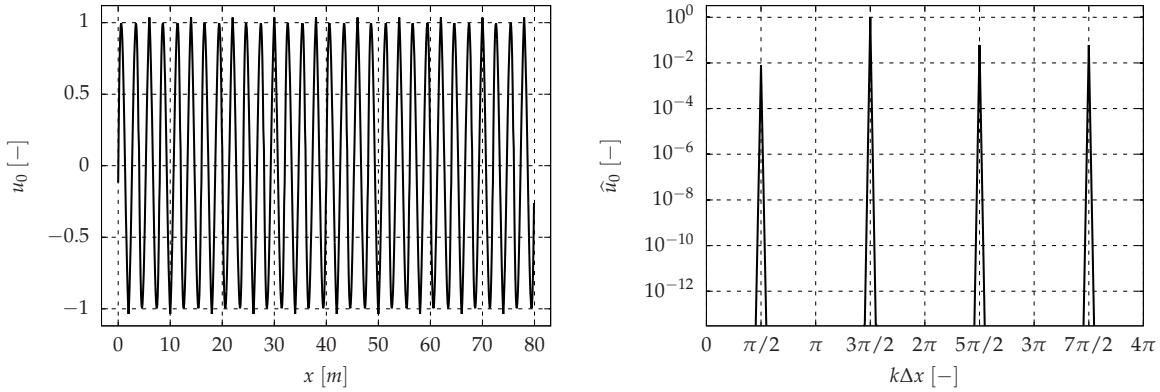
(a) *Initial solution.* **(b)** *Initial solution Fourier Transform.*

Fig. 9.11. *Initial solution for* $k\Delta x = \pi/2$.



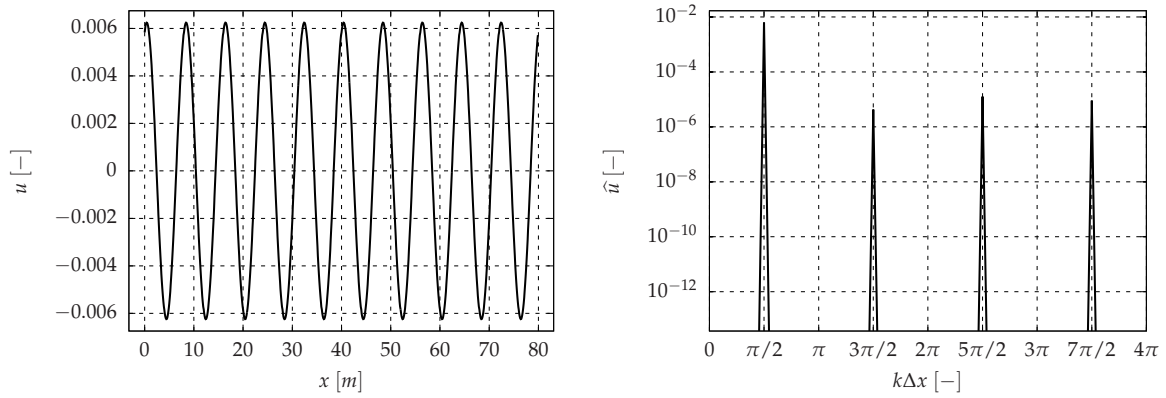
(a) *Final solution.* **(b)** *Final solution Fourier Transform.*

Fig. 9.12. *Final solution for* $k\Delta x = \pi/2$.



(a) *Initial solution.* **(b)** *Initial solution Fourier Transform.*

Fig. 9.13. *Initial solution for* $k\Delta x = 3\pi/2$.



(a) *Final solution.* **(b)** *Final solution Fourier Transform.*

Fig. 9.14. *Final solution for* $k\Delta x = 3\pi/2$.

9.5.2 Mathematical consideration

Let us define the space $L^2([-1,+1])$ of complex-valued functions on the closed interval $[-1,+1]$ and the associated complex scalar product by:

$$\forall (f,g) \in \left(L^2([-1,+1])\right)^2,\ \langle f,g\rangle = \int_{-1}^{+1} f(x) \times \mathrm{conj}(g)(x)\, dx, \tag{9.48}$$

where $\mathrm{conj}(g)$ is the complex conjugate of g. Obviously, the norm associated with this scalar product is defined by:

$$\forall f \in L^2([-1,+1]),\ \|f\| = \sqrt{\int_{-1}^{+1} |f(x)|^2\, dx}. \tag{9.49}$$

Moreover, $\left\|\alpha\, e^{j\beta} f\right\| = \alpha \|f\|$ and $\langle \alpha\, e^{j\beta} f, g\rangle = \alpha\, e^{j\beta}\langle f,g\rangle$ for all $(\alpha,\beta) \in \mathbb{R}^+ \times [0,2\pi[$ by sesquilinearity of the complex scalar product.

Let us use this complex scalar product to compute the dispersion and the dissipation of the Spectral Difference method. For wavenumbers in $[0,\pi]$, the spectral analysis follows the MPM. However, for wavenumbers larger than π, coming back to the physical meaning of dissipation and dispersion, *dissipation can be expressed as the rate of loss in energy of the signal, while dispersion is the phase shift of the signal*. In the case of spectral analysis, the functions f are simply represented by the complex exponential basis $f^n(x) = \exp(-j\omega n\Delta t + jkx) = \alpha\, e^{j\beta} f^0(x)$. The dissipation and the dispersion after n iterations can be expressed as the product of the initial solution by $\alpha\, e^{j\beta}$. Mathematically, the dissipation is defined as the loss of the L^2-norm of the signal f between time 0 and n by:

$$\frac{\|f^n\|}{\|f^0\|} = \frac{\left\|\alpha\, e^{j\beta} f^0\right\|}{\|f^0\|} = \alpha. \tag{9.50}$$

The phase shift of the signal f between time 0 and n can be simply expressed as:

$$\arg\left(\langle f^n, f^0\rangle\right) = \arg\left(\langle \alpha\, e^{j\beta} f^0, f^0\rangle\right) = \arg\left(\alpha\, e^{j\beta}\langle f^0, f^0\rangle\right) = \arg\left(\alpha\, e^{j\beta} \left\|f^0\right\|^2\right) = \beta. \tag{9.51}$$

The complex scalar product is defined on $L^2([-1,+1])$ and this is the right way to take into account the fact that all quantities are defined by polynomials. However, (9.50) and (9.51) are obtained using only the sesquilinearity of the complex scalar product since the definition of the scalar product (9.48) is never explicitly used. Actually, the definition of both dispersion

and dissipation does not depend on the considered scalar product. It is more convenient (and simple!) to replace the L^2 norm for functions by the ℓ^2-norm $\|\cdot\|_2$ of vectors in $\mathbb{C}^{p+1}$, playing directly with the solutions at solution points. Of course, this norm is derived from the following complex scalar product:

$$\forall\,(x,y)\in\left(\mathbb{C}^{p+1}\right)^2,\ \langle x,y\rangle=\sum_{l=1}^{p+1}x_l\times\mathrm{conj}(y_l). \tag{9.52}$$

9.5.3 Energy loss estimation

As explained before, we replace the standard L^2-norm of functions by the ℓ^2-norm of the solution point vectors $\widehat{U}_i^n$. Let us introduce the ratio $\rho_{n,m}$ for $(n,m)\in\mathbb{N}^+\times\mathbb{N}^+$ with $n>m$:

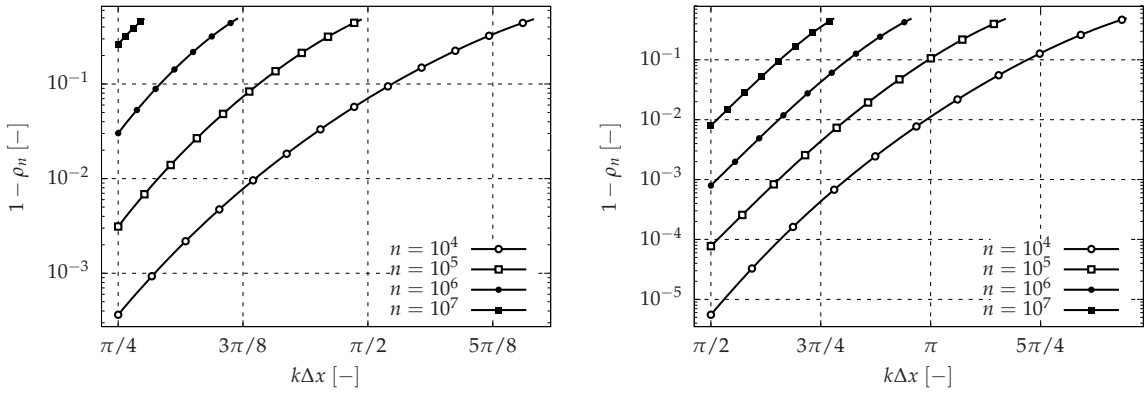
$$\rho_{n,m}=\frac{\left\|\widehat{U}_i^n\right\|_2}{\left\|\widehat{U}_i^{n-m}\right\|_2}. \tag{9.53}$$

By definition, $\rho_{n,m}$ represents the energy loss of the solution between iteration $n-m$ $(n\geq m)$ and iteration n. The MPM gives the behavior of $\rho_{n,m}$ when n is sufficiently large:

$$\rho_{n,m}=\sqrt{\frac{\sum_{l=1}^{p+1}\left|\alpha_l^{(0)}\lambda_l^n\right|^2}{\sum_{l=1}^{p+1}\left|\alpha_l^{(0)}\lambda_l^{n-m}\right|^2}}\underset{n\to+\infty}{\sim}\left|\lambda_{p+1}^m\right|\underset{n\to+\infty}{\sim}\rho^m. \tag{9.54}$$

Using (9.54), it is clear that $\rho_{n,m}$ is a generalization of the standard criterion for scheme dissipation and $\rho_{n,m}$ behaves as the spectral radius raised to a power equal to the number of iterations.

Let $\rho_n=\rho_{n,n}$ represent an estimation of the energy loss by the initial signal after n iterations. In Fig. 9.15, the loss in energy for two values of p is shown as a function of the number of iterations for computations performed at CFL number $\nu=0.1$. As expected, the loss in energy increases with the number of iterations to perform. ρ_n measures the dissipation effect above π but we need to introduce a new quantity to account for dispersion.



(a) *SD3.* **(b)** *SD5.*

Fig. 9.15. *Loss in energy for two values of p as a function of the number of iterations ($\nu=0.1$).*

9.5.4 Phase shift estimation

Let us introduce $\delta\varphi_{n,m}$ for $(n,m)\in\mathbb{N}^+\times\mathbb{N}^+$ with $n>m$:

$$\delta\varphi_{n,m}=\arg\left(\langle\widehat{U}_i^n,\widehat{U}_i^{n-m}\exp\left(-jm\nu\,k\Delta x\right)\rangle\right). \tag{9.55}$$

$\delta\varphi_{n,m}$ represents the phase shift of the solution between iterations $n-m$ and n. Indeed, the factor $\exp(-jm\nu\, k\Delta x)$ takes into account the theoretical advection given by the analytical solution of the advection equation with a constant advective velocity c. We have:

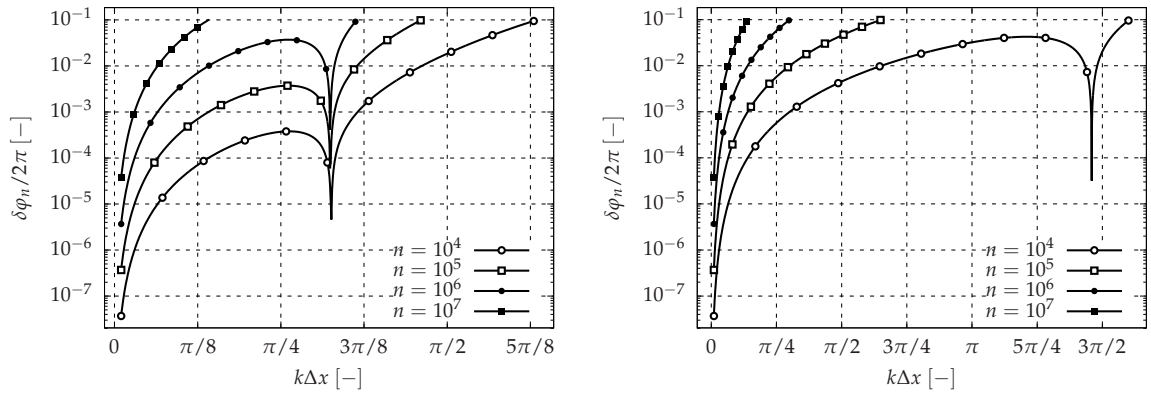
$$\delta\varphi_{n,m} \underset{n\to+\infty}{\sim} \arg\left(\alpha_{p+1}^{(0)}\lambda_{p+1}^{n}\mathrm{conj}(\alpha_{p+1}^{(0)}\lambda_{p+1}^{n-m})\exp(jm\nu\, k\Delta x)\right) \tag{9.56}$$

$$\underset{n\to+\infty}{\sim} \arg\left(\left|\alpha_{p+1}^{(0)}\right|^2\rho^n\exp(-jn\nu\varphi)\,\rho^{n-m}\exp(j(n-m)\nu\varphi)\exp(jm\nu\, k\Delta x)\right) \tag{9.57}$$

$$\underset{n\to+\infty}{\sim} \arg\left(\rho^{2n-m}\left|\alpha_{p+1}^{(0)}\right|^2\exp(jm\nu\,(k\Delta x-\varphi))\right) \tag{9.58}$$

$$\underset{n\to+\infty}{\sim} m\nu\,(k\Delta x-\varphi)\,. \tag{9.59}$$

This criterion computes the phase shift of the signal induced by the numerical scheme between iteration n and iteration $n-m$. Let us note $\delta\varphi_{n,n}=\delta\varphi_n$ which computes an estimation of the phase delay between the signal after n iterations and the initial signal which has been analytically convected. It measures the overall dispersion induced by the time integration loop. This latter is a generalization of the usual criterion but it takes into account the effect of the time integration. Indeed, the spectral behavior evolves during the number of iterations. Furthermore, $\delta\varphi_n/2\pi$ gives the phase shift length per number of wavelength. For example, $\delta\varphi_n/2\pi=0.1$ means that both signals are separated by 0.1λ where λ is the signal wavelength. In Fig. 9.16, the phase shifts for SD3 and SD5 are shown as a function of the number of iterations, in a similar way as for Fig. 9.15.



(a) *SD3.* **(b)** *SD5.*

Fig. 9.16. *Phase shift for two values of p as a function of the number of iterations ($\nu=0.1$).*

9.5.5 Summary

In this section, new criteria for dissipation and dispersion have been introduced. The criteria take into account both space and time schemes. It is shown that for wavenumbers above π, aliasing occurs but the aliasing speed depends on the CFL number, the time integration procedure, the number of time steps and the polynomial degree p. The new criteria based on energy loss and phase shift are extensions of the standard Fourier approach introduced in the previous sections. In particular, the results presented in Sec. 9.4 are recovered by the new approach for wavenumbers in $[0,\pi]$ and a large number of iterations (asymptotic behavior). We note that the new criteria are general in the sense that they can be applied to any kind of numerical scheme.

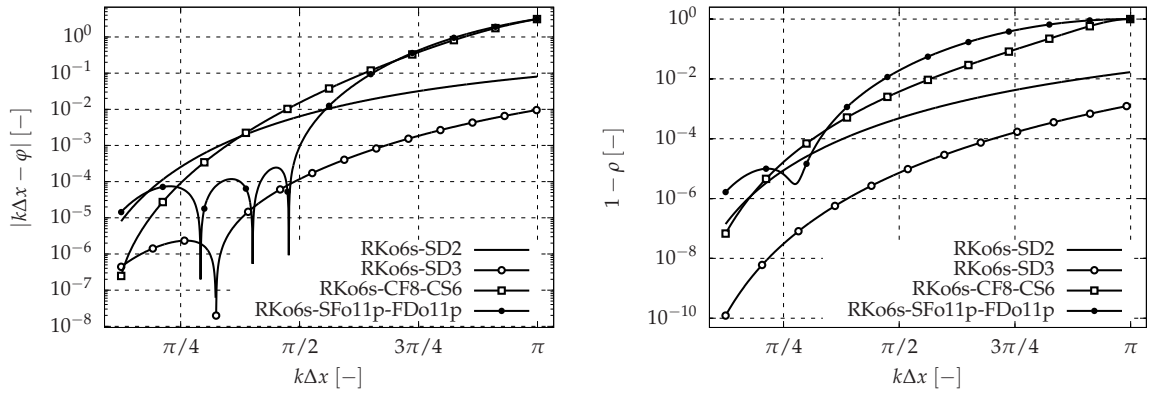
In the next section, these criteria are used to compare the Spectral Difference scheme with the Finite Difference approaches introduced in Sec. 9.3.

9.6 Comparison with standard high-order schemes

This section is devoted to the comparison of the spectral behavior between the standard schemes for aeroacoustics and the Spectral Difference method.

9.6.1 Naive comparison

For standard spectral analysis of FD schemes, $\mathcal{G}(k\Delta x)$ defined in (9.17) can be seen as the transfer function between $\widehat{u}_i^n$ and $\widehat{u}_i^{n+1}$ after applying the spatial numerical scheme, the time integration and the filter. Like all transfer functions, it can be characterized by its modulus, which was called $\rho = |\mathcal{G}|$, and its argument, which was called $\varphi = -\arg(\mathcal{G})/\nu$. Note that the argument of $\mathcal{G}(k\Delta x)$ was (i) multiplied by $\mathbf{(-1)}$ because of the normal mode choice $u_i^n = \exp((\mathbf{-1})jn\,\omega\Delta t + ji\,k\Delta x)$ and (ii) divided by the CFL number ν to obtain a quantity which is always equal to $k\Delta x$ in the exact case since the dimensionless dispersion relation is $\omega\Delta t = \nu k\Delta x$. For the spectral analysis of the Spectral Difference method, the transfer function is λ_{p+1}. Indeed, when the number of iterations is large, $\widehat{U}_i^{n+1}$ is obtained from $\widehat{U}_i^n$ multiplied by λ_{p+1}. Analogously, $\varphi = -\arg(\lambda_{p+1})/\nu$ and $\rho = |\lambda_{p+1}|$ were defined. These quantities are plotted in Fig. 9.17a and Fig. 9.17b respectively.



(a) *Dispersion error.* **(b)** *Dissipation error.*

Fig. 9.17. *Spectral analysis for $\nu = 0.1$: comparison of the schemes. For SD schemes, these results are obviously those given by the MPM.*

In the light of these figures, the RKo6s-SD2 scheme seems to have a similar spectral behavior as both RKo6s-CF8-CS6 and RKo6s-SFo11p-FDo11p. Nevertheless one has to remember that several degrees of freedom are located within any mesh element in the Spectral Difference approach. It is therefore preferable to introduce a new criterion that takes into account the total number of degrees of freedom instead of the number of mesh elements and their size.

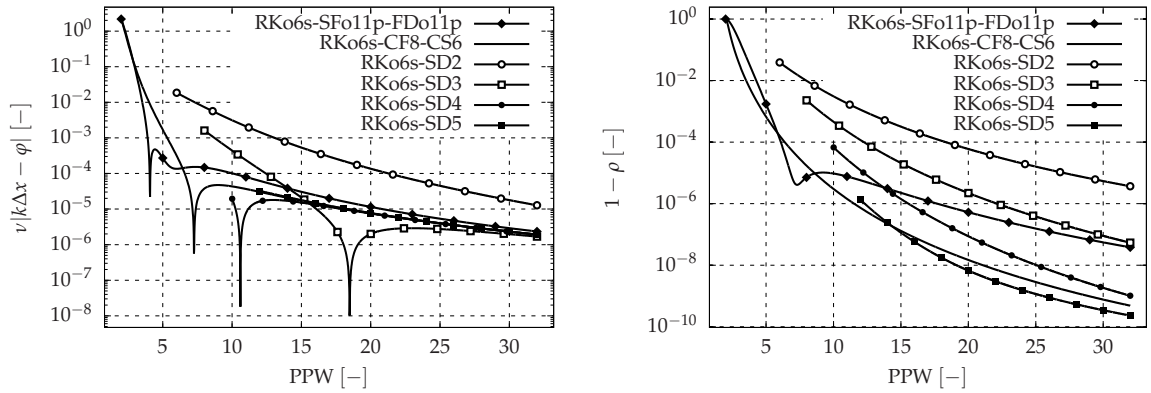
9.6.2 Rescaling by the number of degrees of freedom

For a fair comparison, the analysis may be performed considering the dimensionless wavenumber $k\Delta x'$ built with the mean distance between two degrees of freedom and not with the element size Δx. It means that, for the Spectral Difference method, the dimensionless wavenumber should be $k\Delta x/(p+1)$ whereas, for standard schemes, the dimensionless wavenumber remains $k\Delta x$. Of course, the same rescaling is performed for the CFL number: $\widehat{\nu} = (p+1)\,c\Delta t/\Delta x$ for the SD method while $\nu = c\Delta t/\Delta x$ for FD schemes. Dispersion and dissipation are now defined as a function of the number of points per wavelength (PPW) and two definitions are introduced: $\text{PPW} = (p+1)\,\lambda/\Delta x$ for the Spectral Difference method while $\text{PPW} = \lambda/\Delta x$ for the FD schemes. Moreover, the CFL number is kept constant: $\widehat{\nu} = 0.7$ for SD method and $\nu = 0.7$ for standard schemes. This choice is motivated by the fact that both the SD and the

finite difference methods lead to the same physical time step. Moreover, the CFL value is chosen in agreement with the one for previous aeroacoustic computations using the compact scheme.

Finally, it must be mentioned that the new quantity $\nu\,|k\Delta x - \varphi|$ is introduced to account for the change in CFL number definition for analyzing dispersion.

In terms of dispersion, the RKo6s-SD4 method seems to be equivalent to the RKo6s-CF8-CS6 combination whereas in terms of dissipation, the RKo6s-CF8-CS6 combination is between the RKo6s-SD4 and RKo6s-SD5 method (Fig. 9.18). However, two problems can be observed. First, the dimensionless wavenumber belongs to $[0, \pi/(p+1)]$ for the SD method. So, it cannot be compared on the full range of wavenumbers in $[0, \pi]$. Moreover, dispersion and dissipation in $[0, \pi/(p+1)]$ come from the asymptotic behavior introduced in Sec. 9.4.8 and do not benefit from the matrix form of the SD method for wavenumbers higher than $\pi/(p+1)$. It is thus mandatory to extend the analysis in energy loss and phase shift to standard schemes for a fair comparison.



(a) *Dispersion error.* **(b)** *Dissipation error.*

Fig. 9.18. *Spectral analysis for $\nu = 0.7$: comparison of the considered schemes.*

9.6.3 New criteria extension to standard aeroacoustic schemes

For the considered Finite Difference schemes with time integration and filtering, the solution is updated using a simple multiplication:

$$\widehat{u}_i^{n+1} = \mathcal{G}\widehat{u}_i^n \tag{9.60}$$

Introducing as before the loss in energy and the phase shift, it comes:

$$\rho_n = \frac{\left\|\widehat{u}_i^n\right\|_2}{\left\|\widehat{u}_i^0\right\|_2} = \frac{\left\|\mathcal{G}^n\widehat{u}_i^0\right\|_2}{\left\|\widehat{u}_i^0\right\|_2} = |\mathcal{G}|^n = \rho^n. \tag{9.61}$$

and

$$\delta\varphi_n = \arg\left(\widehat{u}_i^n \mathrm{conj}(\widehat{u}_i^0 \exp\left(-jn\nu k\Delta x\right))\right), \tag{9.62}$$

$$= \arg\left(\mathcal{G}^n\widehat{u}_i^0 \mathrm{conj}(\widehat{u}_i^0 \exp\left(-jn\nu k\Delta x\right))\right), \tag{9.63}$$

$$= \arg\left(\mathcal{G}^n \exp\left(jn\nu\, k\Delta x\right)\right), \tag{9.64}$$

$$= n\nu\, k\Delta x + n\arg\left(\mathcal{G}\right), \tag{9.65}$$

$$= n\nu\left(k\Delta x - \varphi\right). \tag{9.66}$$

Remark: The previous expressions are equalities and not asymptotic limits since $\mathcal{G}$ is a scalar quantity.

9.6.4 Points Per Wavelength (PPW) - Dissipation-based criterion

Introducing the new criteria for Finite Difference schemes, it is now possible to define a cut-off wavenumber $k_c^\varepsilon \Delta x$ that is associated with the conservation of a certain percentage $1-\varepsilon$ of energy after n iterations. To do so, it is mandatory to solve the following problem:

$$\begin{aligned} &\text{For a given set } (n,\nu,p) \in \left(\mathbb{N}^* \times \mathbb{R}^{+*} \times \mathbb{N}^*\right), \text{ look for} \\ &\quad k\Delta x \in [0, k_c^\varepsilon \Delta x] \text{ such that } \rho_n\left(k\Delta x, n, \nu, p\right) \geqslant 1-\varepsilon \end{aligned}$$

This cut-off wavenumber can be obviously interpreted as a number of points per wavelength (PPW). The dissipation-based criteria is finally shown in Fig. 9.19 for the four Finite Difference schemes (RKo6s-UP1, RKo6s-UP2, RKo6s-CF8-CS6 and RKo6s-SFo11p-FDo11p), and for the Spectral Difference approach with p ranging from 2 to 5 and n from 10^4 to 10^7 at CFL number $\nu = 0.7$. The cases with $n = 10^4$ and $n = 10^5$ correspond to the typical numbers of iterations for standard LES. The extreme cases with $n > 10^5$ are only computed to illustrate the asymptotic behavior.

Finally, the PPW criterion is now written as a function of the number of iterations and as a function of the accepted loss of accuracy in Tab. 9.11, Tab. 9.12, Tab. 9.13 and Tab. 9.14.

As expected, when the dissipation criterion is less restrictive, the number of PPW needed to discretize the signal decreases. For Spectral Difference schemes, when the polynomial order p increases, the number of PPW needed to discretize the signal decreases as well. The RKo6s-SFo11p-FDo11p needs more PPW than the RKo6s-CF8-CS6 according to this dissipation criterion. This fact could be explained by the accuracy of the associated filters (sixth-order versus eighth-order). Finally, the RKo6s-SD5 is equivalent to the RKo6s-CF8-CS6 in terms of dissipation. Note the extreme PPW requirements needed for RKo6s-UP1 and RKo6s-UP2 schemes to fulfill the criteria.

Tab. 9.11. *PPW for dissipation criterion* $n = 10^4$.

	$\rho_{10^4} = 99.99\%$	$\rho_{10^4} = 99.9\%$	$\rho_{10^4} = 99\%$
RKo6s-UP1	37170.90	11751.82	3707.86
RKo6s-UP2	406.38	228.49	128.34
RKo6s-SFo11p-FDo11p	40.30	27.03	17.67
RKo6s-CF8-CS6	21.08	15.54	11.52
RKo6s-SD2	85.51	58.38	39.74
RKo6s-SD3	40.87	29.89	22.17
RKo6s-SD4	25.52	19.86	15.59
RKo6s-SD5	19.01	15.18	12.32

Tab. 9.12. *PPW for dissipation criterion* $n = 10^5$.

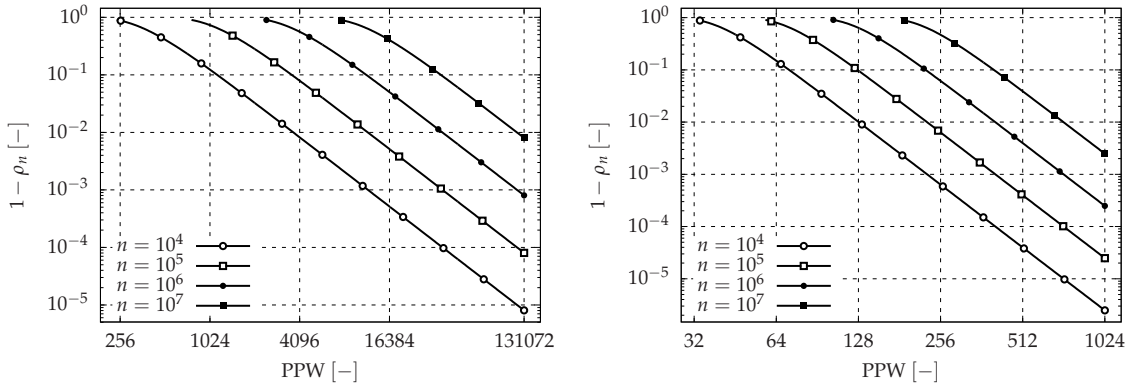
	$\rho_{10^5} = 99.99\%$	$\rho_{10^5} = 99.9\%$	$\rho_{10^5} = 99\%$
RKo6s-UP1	117544.69	37162.53	11725.29
RKo6s-UP2	722.66	406.33	228.23
RKo6s-SFo11p-FDo11p	59.55	40.29	27.00
RKo6s-CF8-CS6	28.89	21.07	15.53
RKo6s-SD2	126.03	85.89	58.45
RKo6s-SD3	53.39	39.68	29.61
RKo6s-SD4	32.22	25.10	19.73
RKo6s-SD5	25.46	19.11	15.18

Tab. 9.13. *PPW for dissipation criterion* $n = 10^6$.

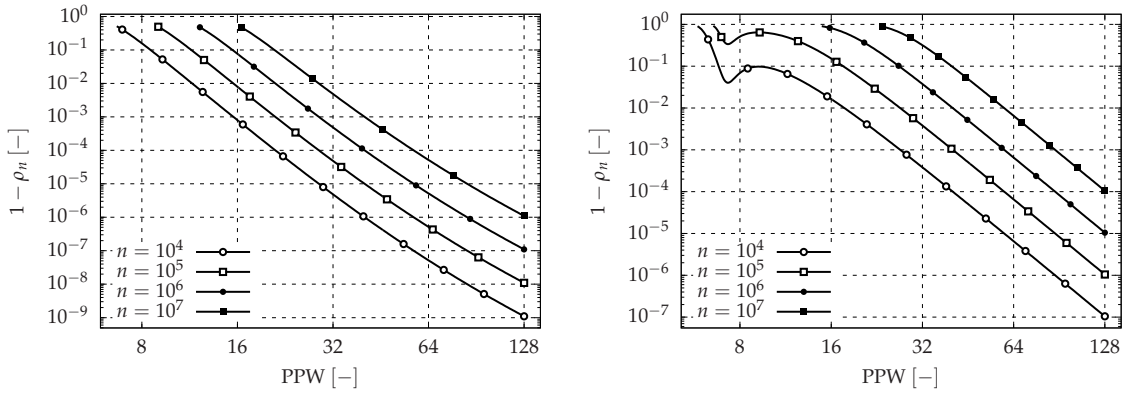
	$\rho_{10^6} = 99.99\%$	$\rho_{10^6} = 99.9\%$	$\rho_{10^6} = 99\%$
RKo6s-UP1	371709.00	117518.24	37078.62
RKo6s-UP2	1285.09	722.58	405.87
RKo6s-SFo11p-FDo11p	87.70	59.55	40.26
RKo6s-CF8-CS6	40.26	28.89	21.06
RKo6s-SD2	185.17	126.14	85.86
RKo6s-SD3	70.83	52.88	39.54
RKo6s-SD4	41.97	32.08	25.04
RKo6s-SD5	36.73	25.52	19.11

Tab. 9.14. *PPW for dissipation criterion* $n = 10^7$.

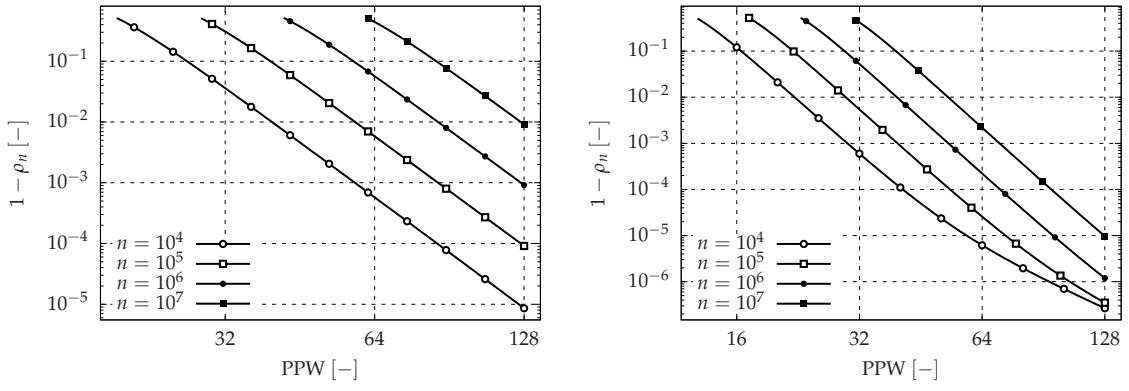
	$\rho_{10^7} = 99.99\%$	$\rho_{10^7} = 99.9\%$	$\rho_{10^7} = 99\%$
RKo6s-UP1	1175450.70	371625.29	117252.90
RKo6s-UP2	2285.25	1284.95	721.76
RKo6s-SFo11p-FDo11p	128.96	87.69	59.50
RKo6s-CF8-CS6	57.46	40.25	28.87
RKo6s-SD2	271.85	185.19	126.06
RKo6s-SD3	94.57	70.61	52.80
RKo6s-SD4	57.36	41.92	32.05
RKo6s-SD5	54.66	36.74	25.51



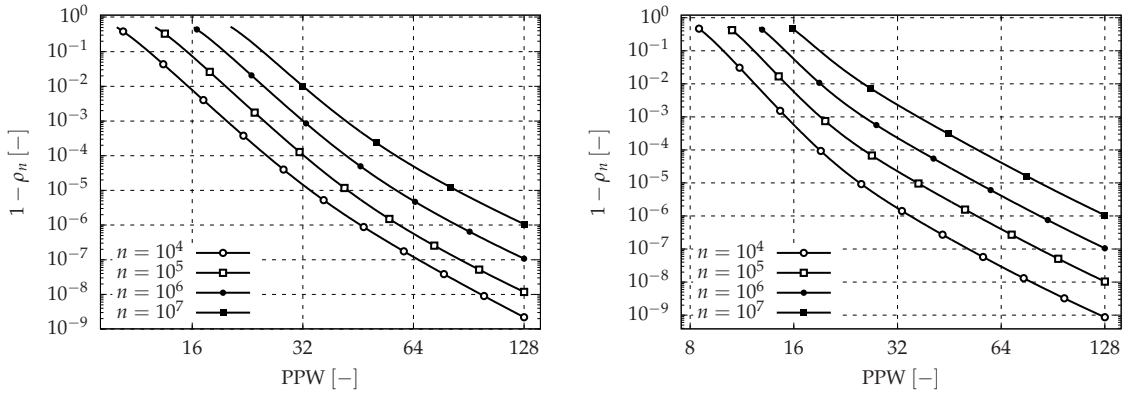
(a) *RKo6s-UP1* **(b)** *RKo6s-UP2*



(c) *RKo6s-CF8-CS6* **(d)** *RKo6s-SFo11p-FDo11p*



(e) *RKo6s-SD2* **(f)** *RKo6s-SD3*



(g) *RKo6s-SD4* **(h)** *RKo6s-SD5*

Fig. 9.19. *Spectral analysis for $\nu = 0.7$: analysis of the dissipation-based criterion.*

9.6.5 Points Per Wavelength (PPW) - Dispersion-based criterion

A cut-off wavenumber $k_c^{\phi}\Delta x$ associated with the phase shift error ϕ after n iterations can be defined in a similar way as for the dissipation-based criterion:

$$\begin{aligned}&\text{For a given set } (n,\nu,p) \in \left(\mathbb{N}^* \times \mathbb{R}^{+*} \times \mathbb{N}^*\right), \text{ look for}\\ &k\Delta x \in \left[0, k_c^{\phi}\Delta x\right] \text{ such that } \delta\varphi_n/2\pi\left(k\Delta x, n, \nu, p\right) \leqslant \phi\end{aligned}$$

This cut-off wavenumber can be obviously interpreted as a number of points per wavelength (PPW). The phase-based criterion is finally shown in Fig. 9.20 for the four Finite Difference schemes (RKo6s-UP1, RKo6s-UP2, RKo6s-CF8-CS6 and RKo6s-SFo11p-FDo11p), and for the Spectral Difference approach with p ranging from 2 to 5 and n from 10^4 to 10^7 at CFL number $\nu = 0.7$.

As before, the PPW criterion is now written as a function of the number of iterations and as a function of the accepted phase shift in Tab. 9.15, Tab. 9.16, Tab. 9.17 and Tab. 9.18.

As expected, when the dispersion criterion is less restrictive, the number of PPW needed to discretize the signal decreases. For Spectral Difference schemes, when the polynomial order p increases, the PPW needed to discretize the signal decreases as well except for pathological cases. This is due to the non-monotonic behavior of dispersion curves, for example, for the RKo6s-SD2 with $n = 10^6$. The RKo6s-SFo11p-FDo11p approximatively needs the same number of PPW as for the RKo6s-CF8-CS6 according to this dispersion criterion. Finally, the RKo6s-SD4 is equivalent to the RKo6s-CF8-CS6 in terms of dispersion. Note the extreme PPW requirements needed for RKo6s-UP1 and RKo6s-UP2 schemes to fulfill the criteria.

Tab. 9.15. *PPW for dispersion criterion* $n = 10^4$.

	$\delta\varphi_{10^4}/2\pi = 0.001$	$\delta\varphi_{10^4}/2\pi = 0.01$	$\delta\varphi_{10^4}/2\pi = 0.1$
RKo6s-UP1	358.19	166.25	77.16
RKo6s-UP2	451.78	209.68	97.29
RKo6s-SFo11p-FDo11p	48.07	23.86	11.95
RKo6s-CF8-CS6	46.51	21.39	6.76
RKo6s-SD2	54.78	36.56	23.38
RKo6s-SD3	46.09	16.65	13.21
RKo6s-SD4	46.51	21.31	9.32
RKo6s-SD5	46.52	21.45	7.38

Tab. 9.16. *PPW for dispersion criterion* $n = 10^5$.

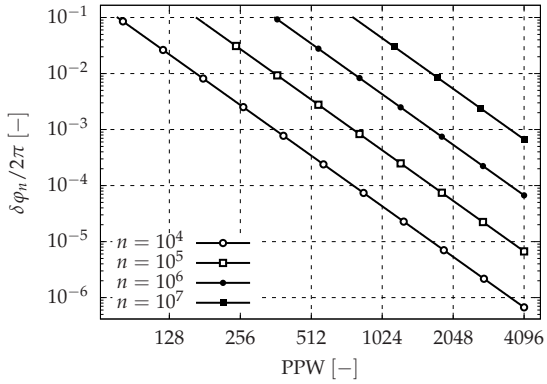
	$\delta\varphi_{10^5}/2\pi = 0.001$	$\delta\varphi_{10^5}/2\pi = 0.01$	$\delta\varphi_{10^5}/2\pi = 0.1$
RKo6s-UP1	771.70	358.19	166.25
RKo6s-UP2	973.35	451.78	209.68
RKo6s-SFo11p-FDo11p	101.14	48.07	23.86
RKo6s-CF8-CS6	100.34	46.51	21.39
RKo6s-SD2	75.16	54.96	36.61
RKo6s-SD3	100.30	46.07	16.66
RKo6s-SD4	100.34	46.51	21.31
RKo6s-SD5	100.34	46.52	21.46

Tab. 9.17. *PPW for dispersion criterion* $n = 10^6$.

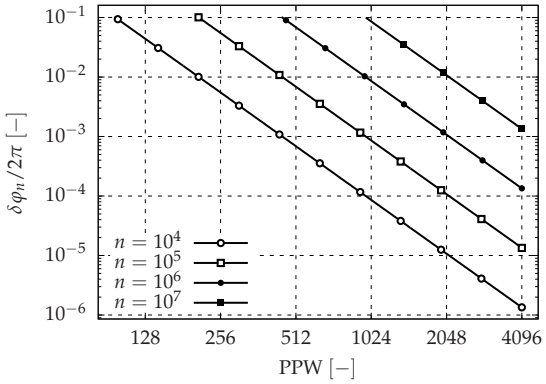
	$\delta\varphi_{10^6}/2\pi = 0.001$	$\delta\varphi_{10^6}/2\pi = 0.01$	$\delta\varphi_{10^6}/2\pi = 0.1$
RKo6s-UP1	1662.58	771.70	358.19
RKo6s-UP2	2097.03	973.35	451.78
RKo6s-SFo11p-FDo11p	216.62	101.14	48.07
RKo6s-CF8-CS6	216.24	100.34	46.51
RKo6s-SD2	200.71	75.22	54.97
RKo6s-SD3	216.24	100.30	46.07
RKo6s-SD4	216.24	100.34	46.51
RKo6s-SD5	216.24	100.34	46.52

Tab. 9.18. *PPW for dispersion criterion* $n = 10^7$.

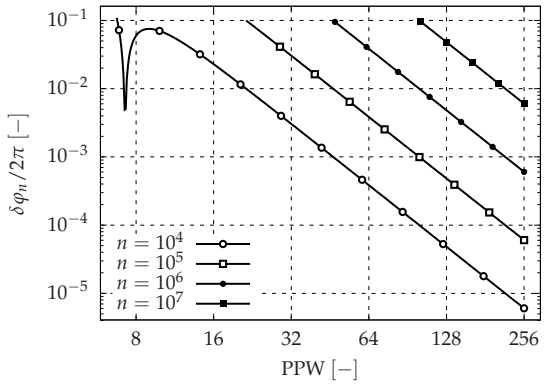
	$\delta\varphi_{10^7}/2\pi = 0.001$	$\delta\varphi_{10^7}/2\pi = 0.01$	$\delta\varphi_{10^7}/2\pi = 0.1$
RKo6s-UP1	3581.93	1662.58	771.70
RKo6s-UP2	4517.91	2097.03	973.35
RKo6s-SFo11p-FDo11p	466.08	216.62	101.14
RKo6s-CF8-CS6	465.91	216.24	100.34
RKo6s-SD2	459.89	200.68	75.22
RKo6s-SD3	465.90	216.24	100.30
RKo6s-SD4	465.91	216.24	100.34
RKo6s-SD5	465.91	216.24	100.34



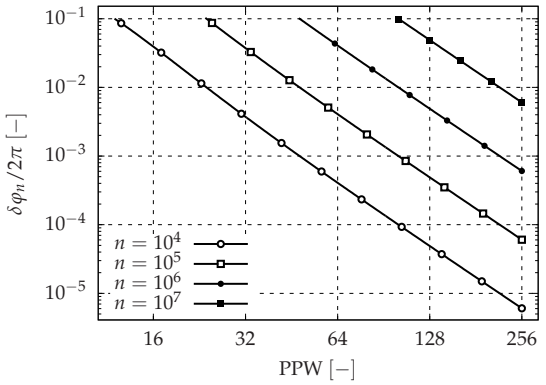
(a) *RKo6s-UP1*



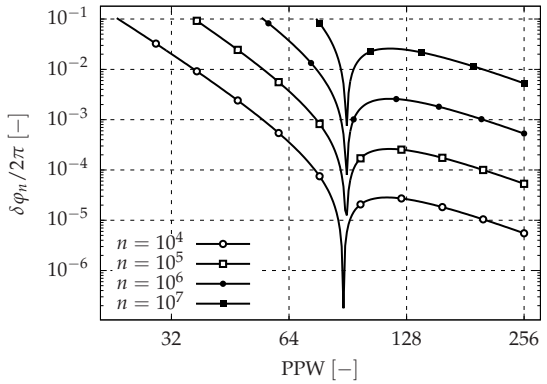
(b) *RKo6s-UP2*



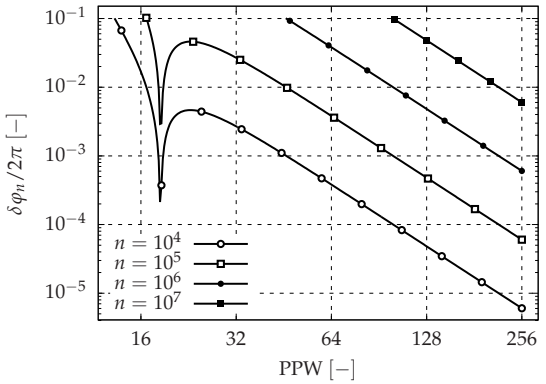
(c) *RKo6s-CF8-CS6*



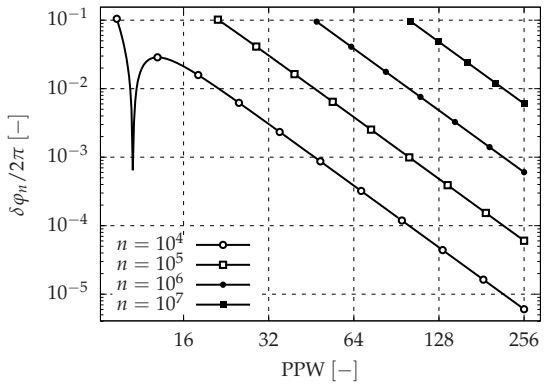
(d) *RKo6s-SFo11p-FDo11p*



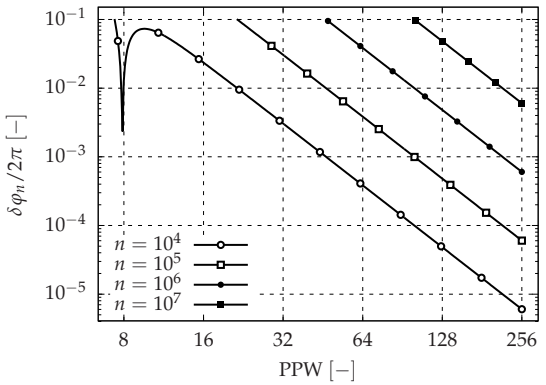
(e) *RKo6s-SD2*



(f) *RKo6s-SD3*



(g) *RKo6s-SD4*



(h) *RKo6s-SD5*

Fig. 9.20. *Spectral analysis for $\nu = 0.7$: analysis of the dispersion-based criterion.*

9.6.6 Computational cost

The spectral analysis shows that the Spectral Difference method is a viable alternative to the standard aeroacoustic schemes for its accuracy. However, the interest is also motivated by the CPU cost associated with the method when solving the Navier-Stokes equations, since providing accurate results in a low restitution time is of utmost importance in an industrial setting.

The numerical methods of this chapter are implemented in three different codes. The RKo6s-CF8-CS6 is implemented in the industrial **elsA** (standing for **e**nsemble **l**ogiciel de **s**imulation en **A**érodynamique) software of ONERA [3, 6]. Even if the implementation is optimized, the performance attained is not the best one could obtain with a dedicated solver. In fact, **elsA** is a generic solver able to compute external aerodynamics and turbomachinery flows with RANS, URANS, LES and DES modeling. Such flexibility is associated with a CPU overhead per degree of freedom. The RKo6s-SFo11p-FDo11p is implemented in the **ALESIA** (standing for **A**ppropriate **L**arge **E**ddy **SI**mulation for **A**eroacoustics) in-house CFD code developed by Dr. Christophe Bogey [175] for research purpose. The code is optimized to run computations on massively parallel platforms. Such an optimized solver is likely to be providing the best performance. Finally, the Spectral Difference method is implemented in the **JAGUAR** (standing for pro**J**ect of an **A**erodynamic solver using **G**eneral **U**nstructured grids **A**nd high orde**R** schemes) code of CERFACS [7]. The structure of **JAGUAR** has been optimized in order to tackle the best CPU performance in serial and parallel computations, including OpenMP and MPI paradigms [176]. In particular, the restitution time [7] is shown to be only weakly sensitive to the polynomial degree p.

Tab. 9.19. *Computational cost τ in μs per iteration n and per Degree Of Freedom (DOF) for the three considered solvers. The computational cost τ (measured on a Intel Bi-Xeon E5-2660v3) is related to the simulation of an academic jet noise configuration when solving the complete Navier-Stokes equations.*

CFD code	**JAGUAR**	**elsA**	**ALESIA**
Schemes	RKo6s-SD	RKo6s-CF8-CS6	RKo6s-SFo11p-FDo11p
Mesh	Unstructured	Structured	Structured
Approach	SD	FV	FD
$\tau\ [\mu s/n/DOF]$	8	10	3

Tab. 9.19 demonstrates that the Spectral Difference method on unstructured grids is a viable alternative to the standard high-order schemes for aeroacoustics applied to structured grids. It opens a new research area for the application of this unstructured paradigm to problems of industrial relevance, including geometrical complexity.

9.7 Conclusion

The spectral analysis based on Fourier modes is one of the standard tools for defining the behavior of a convection scheme. Injecting a solution projected onto the Fourier basis in the derivative, the exact and numerical approximation of the derivative can be compared, leading to a loss in amplitude (dissipation) and a phase-shift (dispersion). In this chapter, we have revisited the spectral analysis focusing our attention on four aspects.

First, we account for the time discretization in the analysis. Such an approach is applied to two sets of standard schemes for aeroacoustics simulations, the compact scheme of Lele and the DRP scheme of Bogey and Bailly (both with filters), while the time integration is performed with the second-order DRP scheme of Bogey and Bailly. The introduction of the time integration is shown to be able to have strong effects on the overall scheme capability, which results in

changes in the standard Point-Per-Wavelength criterion. To the authors' knowledge, this first kind of result had never been highlighted before for the two proposed schemes.

Secondly, the space-time spectral analysis is applied to one of the spectral discontinuous schemes, namely the Spectral Difference scheme. The analysis shows that any solution is a combination of projected components on the set of eigenvectors. In the previous studies, one mode was referred to as a main mode and the others were called spurious. We think that this naming is prone to possible misunderstandings since the solution is always projected onto the whole set of eigenvalues / eigenvectors. Moreover, after simple algebraic simplifications, it is shown that the spectral behavior is well-defined for wavenumbers in $[0, \pi]$ and aliasing always occurs for wavenumbers larger than π. Such a behavior is general and valid for any high-order spectral discontinuous method. However, aliasing was not found in dissipation and dispersion figures published in the literature. Focusing on wavenumbers in $[0, \pi]$, since the Spectral Difference approach behaves as a solution projected onto a set of eigenvalues and eigenvectors, only the component associated with the eigenvector with the largest eigenvalue modulus is kept when the number of iterations to perform is large. For wavenumbers larger than π, aliasing always occurs. A naive conclusion would be to assume that spectral discontinuous methods are not competitive with respect to standard Finite Difference schemes applied in aeroacoustics: the cell size would be the same, but the number of degrees of freedom increases with the polynomial order p.

Thirdly, we were able to extend the space-time analysis for wavenumbers larger than π and we introduced two new criteria to define dissipation and dispersion. Dissipation is measured by an energy loss while dispersion is measured by a phase-shift. These new criteria are general enough to be applied to any kind of scheme, including (of course) Spectral Difference, Spectral Volume, Discontinuous Galerkin or Flux Reconstruction schemes. The criteria recover the standard criteria for any scheme with one degree of freedom per cell / mesh node. Figures related to both dissipation and dispersion were introduced for a range of polynomial order p and number of iterations, for a given CFL number and for a given input wavenumber.

Finally, the accuracy of standard Finite Difference and Spectral Difference schemes is compared. It is shown that the number of Points-Per-Wavelength of the compact scheme of Lele coupled with the time integration procedure and the associated filter is also obtained with the Spectral Difference scheme with $p = 5$ (sixth order of accuracy) in terms of dissipation and $p = 4$ (fifth order of accuracy) in terms of dispersion and the same time integration procedure. In fact, the spectral accuracy of the Lele's scheme coupled with the time integration and the filter is recovered by the Spectral Difference approach coupled with the same time integration procedure. To the best of our knowledge, it is the first time that a method for unstructured grids presents the same spectral accuracy as the compact scheme. As a consequence, it offers the numerical capability required in aeroacoustics computations. It is also shown using dedicated solvers that the CPU times per degree of freedom for the three considered approaches are of the same order of magnitude. Therefore, the SD method presents the same numerical efficiency as the standard Finite Difference schemes and the Spectral Difference method can be considered as an alternative to the Finite Difference schemes. Moreover the Spectral Difference method provides the additional flexibility of unstructured grids necessary to perform computations with complex geometries of industrial interest. Hence we plan to use the Spectral Difference method to perform aeroacoustics simulations for both academic and industrial configurations in a near future.

VI Conclusion & Perspectives

10 Conclusion 165

11 Perspectives 167

10 Conclusion

"Science is built up of facts, as a house is with stones. But a collection of facts is no more a science than a heap of stones is a house."

— Jules Henri Poincaré (1854 - 1912)

The objective of this Ph.D. work was to find an efficient (in terms of HPC) and accurate (to capture the flow physics) CFD method to deal with complex geometries and convection of unsteady flow over a long distance. It relied on the extension of a method commonly used by industries during preliminary design stages: the Finite Volume Method implemented in the ***elsA*** solver co-developed by ONERA and CERFACS. The Reynolds Average Navier-Stokes (RANS) equations are solved with standard second-order schemes on structured multiblock meshes. ***elsA*** was recently extended to deal with unstructured grids in the context of URANS simulations to bring flexibility for meshing. To treat always more complex industrial applications, it was decided to develop a hybrid approach to be able to compute simultaneously on structured and unstructured zones. This being said, the focus can be put on the coupling of unstructured and structured blocks and dedicated schemes inside a single computational domain. In the present work, the coupling between structured and unstructured zones was performed through the use of a nonconforming grid interface (NGI) treatment.

In Chap. 2, the nonconforming grid interface was studied in details with a special emphasis on unsteady flows using structured grids. For this purpose, a new framework was proposed to handle the spectral analysis of numerical schemes applied on a nonconforming grid interface between two structured blocks for the two-dimensional advection equation. To avoid spurious reflection, a dedicated treatment, based on an approximate Riemann solver, should be applied to take into account the potential discontinuity of metrics.

In Chap. 3, the attention beared on the Vichnevetsky's framework which highlights the fully unsteady behavior of this kind of interface. It explains the reflection of spurious waves when a discontinuity occurs in the computation. Among these discontinuities, there are numerical schemes discontinuities, refinement or coarsening discontinuities, element type discontinuities if the mesh is unstructured, nonconforming grid interface or even hybrid nonconforming grid interface.

In Chap. 4, the hybrid approach for the convective flux for first- and second-order schemes was validated. The used test case deals with the convection of an isentropic compressible vortex. The validation was performed on several meshes with different topologies to test the implementation. Expected results for low-order schemes were obtained. In Chap. 5, the hybrid approach for the diffusive flux was also validated. The validation for the diffusive flux was performed with the Taylor-Green vortex and gave the expected results. For applications where second-order schemes are broadly acceptable, typically dealing with RANS or URANS simulations, the hybrid approach was successfully validated and could be used for these kinds of application.

What is more, dealing with the convection of unsteady aerodynamics over a long distance also requires high-order CFD methods or at least low dispersion and low dissipation numerical methods. For the structured side, Fosso [6] developed high-order schemes dedicated to the transport of turbulent structures. This sixth-order scheme has low dissipation and low dispersion properties and it is therefore a good solution to compute wake vortices convection. The key point was to develop high-order unstructured schemes. However, high-order unstructured schemes are much more complicated than the structured ones. Chap. 6 was dedicated to high-order reconstruction on unstructured grids derived from the standard second-order reconstruction schemes. The reconstruction is compacted using the gradients in each cell and therefore, the accuracy is directly driven by the accuracy of this gradient. A Riemann solver is used after a high-order extrapolation step. It was shown that the metrics must be included into this gradient to improve the accuracy of the extrapolation.

In Chap. 7, an Unstructured Interface Gradient (UIG) was developed since the standard gradient formulation suffers from a lack of accuracy for general unstructured grids. This new gradient is second-order accurate on general unstructured grids. The goal was also to improve actual diffusion schemes and to find a high-order unstructured compact scheme for the convection term. However, Chap. 8 showed that this kind of formulation recovers the k-exact formulation. Indeed, to increase the order of accuracy of any numerical scheme, information is needed and this implies the need for an extended stencil even if it can be informatically reduced. This kind of formulation can however lead to a loss of HPC efficiency which is why the focus was put on more promising high-order methods.

An alternative approach, the Spectral Difference Method, is studied in Chap. 9. Starting from the point that the accuracy of a LES depends on the local mesh refinement and the scheme properties, the spectral analysis of Spectral Discontinuous Methods was revisited. Several key points were highlighted. The first point concerned the effects of time integration on the spectral analysis. It was shown, with standard high-order Finite Difference schemes dedicated to aeroacoustics, that the time integration has an effect on the required number of points per wavelength. Secondly, a coupled space-time approach based on the Matrix Power Method is applied to the new class of high-order Spectral Discontinuous approaches, focusing especially on the Spectral Difference method. It was shown that the Spectral Difference method is as accurate as the considered Finite Difference schemes or even more which leads to believe that this kind of method is very promising. Moreover, the computational time appears to be only weakly dependent on the order of the scheme.

11 Perspectives

"Your theory is crazy, but it's not crazy enough to be true."

— Niels Henrik David Bohr (1885 - 1962)

In this Ph.D. work, the hybrid approach was successfully validated with low-order schemes. One of the most obvious perspectives is to compute an application where second-order schemes are broadly acceptable, typically in RANS or URANS simulations. Moreover, it could be interesting to implement the k-exact approach in the unstructured kernel of the ***elsA*** software even if this kind of schemes does not seem to be efficient in terms of computation time. All in all, the unstructured k-exact schemes could be coupled with the sixth-order scheme of Fosso [6] to be able to do a high-order hybrid computation, even if, again, this solution does not seem to be very attractive, according to the results of the spectral analysis and to the efficiency in terms of computation time of the Spectral Difference Method.

In Chap. 9, the spectral analysis was revisited for Spectral Discontinuous Methods. In particular, the time integration effect was taken into account in (9.41). The matrix G depends on the polynomial order, on the Courant-Friedrichs-Lewy (CFL) number and on the coefficients of the Runge-Kutta integration. The basic idea is therefore to perform an optimization on this Runge-Kutta coefficients to increase the CFL stability limit of the time integration. It is even possible to consider this optimization to build low dispersion and low dissipation space/time coupled schemes. The first results are very encouraging since the technique allows to gain up to 60% on the CFL stability limit. Low dissipation and low dispersion schemes are not designed yet. But once they are designed, a paper will be submitted in the Journal of Computational Physics [177] and presented at the AIAA Conference [178]. Another time integration procedure can be also considered such as Implicit-Explicit (IMEX) time integration or exponential integrator.

Another future research field concerns the initialization of the solution when using the Spectral Difference Method. Indeed, this method uses a polynomial basis on which the initial solution has to be projected. This topic is of utmost importance when computing the error. Since the Spectral Difference Method has excellent dissipation and dispersion properties, the projection step onto the polynomial basis can leave some numerical artifacts that can distort the error analysis. A paper is under preparation for the Journal of Computational Physics [179] to properly define how to compute an error with this method and to show what happens in the opposite case.

Other perspectives of this Ph.D. work are related to the treatment of more complex geometries. Indeed, at the moment, the Spectral Difference Method is restricted to hexahedra. For sure, it exists extensions of the Spectral Difference Method on triangles [180, 181] but nothing on tetrahedra. It would be worth spending some time working on an extension of the method on tetrahedral meshes. Then, it could be also interesting to develop the sliding mesh capability for turbomachinery applications thanks to the Mortar element Method. Zhang *et al.* [182] only developed this capability for two-dimensional computations but not for three-dimensional

computations. To treat very complex geometries, the Spectral Difference Method could be also extended to polyhedral meshes to treat body-fitted Cartesian meshes.

Finally, the last future line of research deals with shock-capturing methods adapted to the Spectral Difference Method if the latter is successfully extended to tetrahedra. One of the solutions could be the tetrahedral mesh adaptation. It could be used to refine the meshes only in the close vicinity of the shocks to improve the accuracy of their capture. Moreover, since the order of accuracy is an input of the Spectral Difference Method thanks to the polynomial basis, it could be possible to couple the space adaptation previously mentioned, called the h refinement, with the refinement of the polynomial order, called the p refinement, to implement hp refinement.

Julien Vanharen, March 2017.

VII Appendices

A The elsA workflow 171
A.1 Hybrid mesh generation
A.2 Hybrid splitter
A.3 Face-based connectivity
A.4 Coprocessing

B Analytical order of accuracy 173
B.1 Preliminaries
B.2 Heat equation
B.3 Advection-diffusion equation
B.4 Error computation

C The COVO analytic solution 177

D Order analysis for unsteady equations 183
D.1 Truncation error
D.2 Discretization error

E About the work unit evaluation 187

A The *elsA* workflow

"I decry the current tendency to seek patents on algorithms. There are better ways to earn a living than to prevent other people from making use of one's contributions to computer science."

— Donald Ervin Knuth (1938 -)

In this appendix, the ***elsA*** worflow is detailed with a special emphasis on hybrid mesh generation, hybrid splitter, face-based connectivity computation and coprocessing.

A.1 Hybrid mesh generation

No industrial tools are available to generate hybrid meshes. Structured and unstructured meshes are separately generated. Structured meshes are generated either by Python scripts or by Ansys ICEM CFD whereas unstructured meshes are generated either by Gmsh [183] or by CENTAUR. Then, the concept of mesh algebra was developed to be able to couple structured and unstructured mesh through the nonconforming grid interface. Basically, each mesh is read and then dumped into a CGNS [184] file. A data structure readable by ***elsA*** is added into the CGNS tree to define the nonconforming grid interface. It is indicated either by families or by geometric reconstruction by computing the convex hull of windows which are shared by the nonconforming grid interface.

A.2 Hybrid splitter

A hybrid splitter was developed. It couples the ***elsA*** splitter for structured meshes and an unstructured splitter based on Metis [185] for unstructured meshes. The load balancing is based on the number of faces.

A.3 Face-based connectivity

elsA needs the face-based connectivity for its unstructured kernel. Following the notations of Fig. A.1, the element-based connectivity is given by $(1,2,3,4)$ and should be transformed into the face-based connectivity given by $(1,3,2)$, $(1,2,4)$, $(2,3,4)$ and $(3,1,4)$. This face-based connectivity requires the computation of the dual graph based on the Metis [185] library.

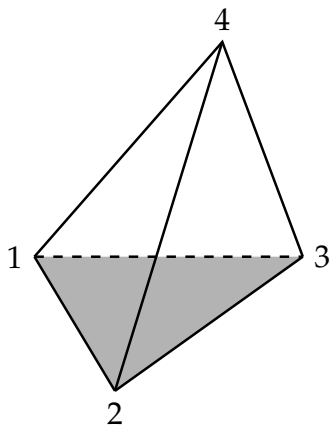


Fig. A.1. *Tetrahedral element connectivity.*

A.4 Coprocessing

Unsteady computations generates a lot of data. Because ***elsA*** benefits from a Python wrapper, it allows the execution of Python code during the computation. A set of coprocessing tools based on Antares [186] was developed to extract on the fly the quantities of interest.

B Analytical order of accuracy

"Science has not yet taught us if madness is or is not the sublimity of the intelligence."

— Edgar Allan Poe (1809 - 1849)

In this appendix, the solution of the advection-diffusion equation is detailed to demonstrate (4.10) and how to obtain (4.4) from (4.11).

B.1 Preliminaries

B.1.1 Fourier transform

Let us define the Fourier Transform $\mathcal{FT}$ and the Inverse Fourier Transform $\mathcal{FT}^{-1}$ for a function $f \in L^2\left(\mathbb{R}^d\right)$ where d is the dimension. This definition is used in the whole appendix.

$$\left\{\mathcal{FT}\left[f\left(x\right)\right]\right\}\left(\xi\right) = \widehat{f}\left(\xi\right) = \frac{1}{\left[2\pi\right]^{d/2}} \int_{\mathbb{R}^d} f\left(x\right) \mathrm{e}^{-jx\xi} \,\mathrm{d}x, \tag{B.1}$$

$$\left\{\mathcal{FT}^{-1}\left[\widehat{f}\left(\xi\right)\right]\right\}\left(x\right) = f\left(x\right) = \frac{1}{\left[2\pi\right]^{d/2}} \int_{\mathbb{R}^d} \widehat{f}\left(\xi\right) \mathrm{e}^{+jx\xi} \,\mathrm{d}\xi. \tag{B.2}$$

Moreover, the Fourier Transform of the derivative is given by:

$$\left\{\mathcal{FT}\left[\frac{\mathrm{d}^n f}{\mathrm{d}x^n}\left(x\right)\right]\right\}\left(\xi\right) = \left(j\xi\right)^n \widehat{f}\left(\xi\right). \tag{B.3}$$

Finally, the Fourier Transform of a gaussian function is given by:

$$\left\{\mathcal{FT}\left[\exp\left(-\left(\frac{x}{R_c}\right)^2\right)\right]\right\}\left(\xi\right) = \frac{R_c}{\sqrt{2}} \exp\left[-\frac{R_c^2\xi^2}{4}\right], \tag{B.4}$$

where R_c is a free parameter.

B.1.2 The Gaussian integral

The Gaussian integral is given by:

$$\int_{-\infty}^{+\infty} \exp\left[-x^2\right] \,\mathrm{d}x = \sqrt{\pi}. \tag{B.5}$$

Then, considering the substitution $y = \dfrac{x}{R_c}$, $\mathrm{d}y = \dfrac{1}{R_c}\,\mathrm{d}x$, one obtains:

$$\int_{-\infty}^{+\infty} \exp\left[-\left(\frac{x}{R_c}\right)^2\right] \,\mathrm{d}x = R_c \int_{-\infty}^{+\infty} \exp\left[-y^2\right] \,\mathrm{d}y, \tag{B.6}$$

$$= R_c\sqrt{\pi}. \tag{B.7}$$

B.2 Heat equation

Let us start by the diffusion equation also called the heat equation for $x \in \mathbb{R}$ and $t > 0$,

$$\frac{\partial f}{\partial t}(x,t) = \nu \frac{\partial^2 f}{\partial x^2}(x,t), \tag{B.8}$$

with $f(x, t=0) = f_0(x) = \exp\left[-\left(\frac{x}{R_c}\right)^2\right]$ and ν the constant diffusion coefficient. Taking the Fourier Transform of the heat equation, one obtains:

$$\frac{\partial \widehat{f}}{\partial t}(\xi,t) = -\nu\xi^2 \widehat{f}(\xi,t). \tag{B.9}$$

Integrating this ordinary differential equation for a given ξ, one obtains:

$$\widehat{f}(\xi,t) = \widehat{f_0}(\xi,t)\exp\left(-\nu\xi^2 t\right). \tag{B.10}$$

By taking the Inverse Fourier Transform, one obtains the general solution:

$$f(x,t) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{+\infty} \widehat{f_0}(\xi,t)\, \mathrm{e}^{-\nu\xi^2 t}\mathrm{e}^{+jx\xi}\,\mathrm{d}\xi. \tag{B.11}$$

Using (B.4), one obtains:

$$f(x,t) = \frac{R_c}{\sqrt{2}}\frac{1}{\sqrt{2\pi}}\int_{-\infty}^{+\infty} \exp\left[-\nu\xi^2 t - \frac{R_c^2\xi^2}{4}\right]\mathrm{e}^{+jx\xi}\,\mathrm{d}\xi, \tag{B.12}$$

$$= \frac{R_c}{\sqrt{2}}\left\{\mathcal{FT}^{-1}\left[\exp\left(-\frac{4\nu t + R_c^2}{4}\xi^2\right)\right]\right\}(x), \tag{B.13}$$

$$= \frac{R_c}{\sqrt{2}}\frac{\sqrt{2}}{\sqrt{4\nu t + R_c^2}}\exp\left[-\frac{x^2}{4\nu t + R_c^2}\right]. \tag{B.14}$$

Finally, the solution of the diffusion equation with the given initial solution is:

$$f(x,t) = \frac{R_c}{\sqrt{4\nu t + R_c^2}}\exp\left[-\frac{x^2}{4\nu t + R_c^2}\right]. \tag{B.15}$$

B.3 Advection-diffusion equation

Let us consider the advection-diffusion equation for $x \in \mathbb{R}$ and $t > 0$,

$$\frac{\partial f}{\partial t}(x,t) + c\frac{\partial f}{\partial x}(x,t) = \nu \frac{\partial^2 f}{\partial x^2}(x,t), \tag{B.16}$$

with $f(x, t=0) = f_0(x) = \exp\left[-\left(\frac{x}{R_c}\right)^2\right]$, c the constant advective velocity and ν the constant diffusion coefficient. Considering the substitution $\tau = t$ and $\sigma = x - ct$ and injecting into (B.16), one obtains:

$$\frac{\partial f}{\partial \sigma}\frac{\partial \sigma}{\partial t} + \frac{\partial f}{\partial \tau}\frac{\partial \tau}{\partial t} + c\left(\frac{\partial f}{\partial \sigma}\frac{\partial \sigma}{\partial x} + \frac{\partial f}{\partial \tau}\frac{\partial \tau}{\partial x}\right) = \nu\left(\frac{\partial}{\partial \sigma}\frac{\partial \sigma}{\partial x} + \frac{\partial}{\partial \tau}\frac{\partial \tau}{\partial x}\right)\left(\frac{\partial f}{\partial \sigma}\frac{\partial \sigma}{\partial x} + \frac{\partial f}{\partial \tau}\frac{\partial \tau}{\partial x}\right). \tag{B.17}$$

Since $\frac{\partial \sigma}{\partial t} = -c$, $\frac{\partial \sigma}{\partial x} = 1$, $\frac{\partial \tau}{\partial t} = 1$ and $\frac{\partial \tau}{\partial x} = 0$, one finds:

$$-c\frac{\partial f}{\partial \sigma} + \frac{\partial f}{\partial \tau} + c\frac{\partial f}{\partial \sigma} = \nu\frac{\partial^2 f}{\partial \sigma^2}, \tag{B.18}$$

$$\frac{\partial f}{\partial \tau} = \nu\frac{\partial^2 f}{\partial \sigma^2}. \tag{B.19}$$

The solution of (B.19) is given in Sec. B.2. One obtains:

$$f(\sigma,\tau) = \frac{R_c}{\sqrt{4\nu\tau + R_c^2}} \exp\left[-\frac{\sigma^2}{4\nu\tau + R_c^2}\right]. \tag{B.20}$$

Finally, by expressing σ and τ, the solution of the advection-diffusion equation with the given initial solution is:

$$f(x,t) = \frac{R_c}{\sqrt{4\nu t + R_c^2}} \exp\left[-\frac{(x-ct)^2}{4\nu t + R_c^2}\right]. \tag{B.21}$$

B.4 Error computation

Let us start from:

$$\varepsilon_f^2 = \int_{-\infty}^{+\infty} \left\{ \frac{R_c}{\sqrt{4\nu t + R_c^2}} \exp\left[-\frac{(x-ct)^2}{4\nu t + R_c^2}\right] - \exp\left[-\left(\frac{x-ct}{R_c}\right)^2\right] \right\}^2 \mathrm{d}x, \tag{B.22}$$

$$= \mathcal{I}_1 - 2\mathcal{I}_2 + \mathcal{I}_3, \tag{B.23}$$

where $\mathcal{I}_1$, $\mathcal{I}_2$ and $\mathcal{I}_3$ are given by:

$$\mathcal{I}_1 = \int_{-\infty}^{+\infty} \frac{R_c^2}{4\nu t + R_c^2} \exp\left[-\frac{2(x-ct)^2}{4\nu t + R_c^2}\right] \mathrm{d}x, \tag{B.24}$$

$$\mathcal{I}_2 = \int_{-\infty}^{+\infty} \frac{R_c}{\sqrt{4\nu t + R_c^2}} \exp\left[-\frac{(x-ct)^2}{4\nu t + R_c^2} - \frac{(x-ct)^2}{R_c^2}\right] \mathrm{d}x, \tag{B.25}$$

$$\mathcal{I}_3 = \int_{-\infty}^{+\infty} \exp\left[-\frac{2(x-ct)^2}{R_c^2}\right] \mathrm{d}x, \tag{B.26}$$

$$\tag{B.27}$$

Then, considering the substitution $y = x - ct$, $\mathrm{d}y = \mathrm{d}x$, one obtains:

$$\mathcal{I}_1 = \int_{-\infty}^{+\infty} \frac{R_c^2}{4\nu t + R_c^2} \exp\left[-\frac{2y^2}{4\nu t + R_c^2}\right] \mathrm{d}y, \tag{B.28}$$

$$\mathcal{I}_2 = \int_{-\infty}^{+\infty} \frac{R_c}{\sqrt{4\nu t + R_c^2}} \exp\left[-\frac{y^2}{4\nu t + R_c^2} - \frac{y^2}{R_c^2}\right] \mathrm{d}y, \tag{B.29}$$

$$\mathcal{I}_3 = \int_{-\infty}^{+\infty} \exp\left[-\frac{2y^2}{R_c^2}\right] \mathrm{d}y, \tag{B.30}$$

$$\tag{B.31}$$

Using the integral (B.7), one finds:

$$\mathcal{I}_1 = \frac{R_c^2}{4\nu t + R_c^2} \int_{-\infty}^{+\infty} \exp\left[-\frac{2y^2}{4\nu t + R_c^2}\right] \mathrm{d}y, \tag{B.32}$$

$$= \frac{R_c^2}{4\nu t + R_c^2} \sqrt{\frac{4\nu t + R_c^2}{2}} \sqrt{\pi}, \tag{B.33}$$

$$= \frac{R_c^2}{\sqrt{4\nu t + R_c^2}} \sqrt{\frac{\pi}{2}}. \tag{B.34}$$

$$\mathcal{I}_2 = \frac{R_c}{\sqrt{4\nu t + R_c^2}} \int_{-\infty}^{+\infty} \exp\left[-\frac{y^2}{4\nu t + R_c^2} - \frac{y^2}{R_c^2}\right] \mathrm{d}y, \tag{B.35}$$

$$= \frac{R_c}{\sqrt{4\nu t + R_c^2}} \int_{-\infty}^{+\infty} \exp\left[-y^2 \frac{2R_c^2 + 4\nu t}{R_c^2\left(R_c^2 + 4\nu t\right)}\right] \mathrm{d}y, \tag{B.36}$$

$$= \frac{R_c}{\sqrt{4\nu t + R_c^2}} \sqrt{\frac{R_c^2\left(R_c^2 + 4\nu t\right)}{2R_c^2 + 4\nu t}} \sqrt{\pi}, \tag{B.37}$$

$$= \frac{R_c^2}{\sqrt{R_c^2 + 2\nu t}} \sqrt{\frac{\pi}{2}}. \tag{B.38}$$

$$\mathcal{I}_3 = \int_{-\infty}^{+\infty} \exp\left[-\frac{2y^2}{R_c^2}\right] \mathrm{d}y, \tag{B.39}$$

$$= R_c \sqrt{\frac{\pi}{2}}. \tag{B.40}$$

Finally, one obtains:

$$\varepsilon_f^2 = \mathcal{I}_1 - 2\mathcal{I}_2 + \mathcal{I}_3, \tag{B.41}$$

$$= R_c \sqrt{\frac{\pi}{2}} \left[\frac{R_c}{\sqrt{4\nu t + R_c^2}} - \frac{2R_c}{\sqrt{R_c^2 + 2\nu t}} + 1\right]. \tag{B.42}$$

Since $\nu = \dfrac{c\Delta x}{2}$ and $t = \dfrac{d}{c}$ where d is the convection distance, one finds $\nu t = \dfrac{d\Delta x}{2}$ and,

$$\varepsilon_f^2 = R_c \sqrt{\frac{\pi}{2}} \left[\frac{R_c}{\sqrt{R_c^2 + 2d\Delta x}} - \frac{2R_c}{\sqrt{R_c^2 + d\Delta x}} + 1\right], \tag{B.43}$$

$$= R_c \sqrt{\frac{\pi}{2}} \left[\frac{1}{\sqrt{1 + 2\dfrac{d\Delta x}{R_c^2}}} - \frac{2}{\sqrt{1 + \dfrac{d\Delta x}{R_c^2}}} + 1\right]. \tag{B.44}$$

Introducing $\eta = \dfrac{d\Delta x}{R_c^2} = \dfrac{d}{\Delta x}\left(\dfrac{R_c}{\Delta x}\right)^{-2}$, one recovers (4.4):

$$\varepsilon_f^2(\eta, R_c) = R_c \sqrt{\frac{\pi}{2}} \left[\frac{1}{\sqrt{1 + 2\eta}} - \frac{2}{\sqrt{1 + \eta}} + 1\right]. \tag{B.45}$$

C The COVO analytic solution

"A scientific truth does not triumph by convincing its opponents and making them see the light, but rather because its opponents eventually die and a new generation grows up that is familiar with it."

— Max Karl Ernst Ludwig Planck (1858 - 1947)

In this appendix, the analytic solution of the convection of a vortex is demonstrated. The demonstration is inspired by Masatsuka [187]. Let us consider the two-dimensional Euler equations, where ρ is the density, $\boldsymbol{V} = (u, v)^\top$ is the velocity, p is the pressure, and e is the specific internal energy. The gravity and all other exterior forces are neglected. The system of equations is:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \boldsymbol{V}) = 0, \tag{C.1}$$

$$\frac{\partial \boldsymbol{V}}{\partial t} + \boldsymbol{V} \cdot \nabla \boldsymbol{V} + \frac{\nabla p}{\rho} = 0, \tag{C.2}$$

$$\frac{\partial e}{\partial t} + \boldsymbol{V} \cdot \nabla e + \frac{p}{\rho} \nabla \cdot \boldsymbol{V} = 0. \tag{C.3}$$

(C.3) may be replaced by several forms of the energy equation, in particular by the entropy equation. The entropy is noted s.

$$\frac{\partial s}{\partial t} + \boldsymbol{V} \cdot \nabla s = 0. \tag{C.4}$$

Let us seek an homentropic solution. A homentropic flow has uniform and constant entropy. It distinguishes itself from an isentropic or particle isentropic flow, where the entropy level of each fluid particle does not change with time, but may vary from particle to particle. This means that a homentropic flow is necessarily isentropic, but an isentropic flow need not be homentropic. So (C.4) and (C.3) are trivially satisfied. Moreover, let us seek a solution for which the velocity field is the superposition of a constant mean flow $(\boldsymbol{V}_0, T_0)$ where $\boldsymbol{V}_0 = (u_0, v_0)^\top$ and a perturbation $(\boldsymbol{V}', T')$ convected by the mean flow where $\boldsymbol{V}' = (u', v')^\top$:

$$\boldsymbol{V}(\boldsymbol{x}, t) = \boldsymbol{V}_0 + \boldsymbol{V}'(\boldsymbol{x} - \boldsymbol{x_0} - \boldsymbol{V}_0 t), \tag{C.5}$$

$$T(\boldsymbol{x}, t) = T_0 + T'(\boldsymbol{x} - \boldsymbol{x_0} - \boldsymbol{V}_0 t), \tag{C.6}$$

where $\boldsymbol{x}$ is the point of coordinate $(x, y)^\top$ and $\boldsymbol{x_0} = (x_0, y_0)^\top$ indicates a reference center location of the perturbation field. The density ρ and the pressure p may be written in the following form:

$$\rho(\boldsymbol{x}, t) = \rho(\boldsymbol{x} - \boldsymbol{x_0} - \boldsymbol{V}_0 t), \tag{C.7}$$

$$p(\boldsymbol{x}, t) = p(\boldsymbol{x} - \boldsymbol{x_0} - \boldsymbol{V}_0 t), \tag{C.8}$$

Let us define:

$$(\boldsymbol{x} - \boldsymbol{x_0} - \boldsymbol{V}_0 t)^\top = (\xi(x,t), \eta(x,t))^\top \tag{C.9}$$
$$= (x - x_0 - u_0 t, y - y_0 - v_0 t)^\top, \tag{C.10}$$

and (C.1) becomes:

$$\frac{\partial \rho}{\partial t}(\xi,\eta) + \nabla \cdot (\rho \boldsymbol{V}) = 0, \tag{C.11}$$
$$\frac{\partial \rho}{\partial \xi}\frac{\partial \xi}{\partial t} + \frac{\partial \rho}{\partial \eta}\frac{\partial \eta}{\partial t} + \boldsymbol{V}_0 \cdot \nabla \rho + \nabla \cdot (\rho \boldsymbol{V}') = 0, \tag{C.12}$$
$$-u_0\frac{\partial \rho}{\partial \xi} - v_0\frac{\partial \rho}{\partial \eta} + (u_0, v_0)^\top \cdot \left(\frac{\partial \rho}{\partial x}, \frac{\partial \rho}{\partial y}\right)^\top + \nabla \cdot (\rho \boldsymbol{V}') = 0, \tag{C.13}$$
$$-u_0\frac{\partial \rho}{\partial \xi} - v_0\frac{\partial \rho}{\partial \eta} + (u_0, v_0)^\top \cdot \left(\frac{\partial \rho}{\partial \xi}\frac{\partial \xi}{\partial x}, \frac{\partial \rho}{\partial \eta}\frac{\partial \eta}{\partial y}\right)^\top + \nabla \cdot (\rho \boldsymbol{V}') = 0, \tag{C.14}$$
$$-u_0\frac{\partial \rho}{\partial \xi} - v_0\frac{\partial \rho}{\partial \eta} + (u_0, v_0)^\top \cdot \left(\frac{\partial \rho}{\partial \xi}, \frac{\partial \rho}{\partial \eta}\right)^\top + \nabla \cdot (\rho \boldsymbol{V}') = 0, \tag{C.15}$$
$$-u_0\frac{\partial \rho}{\partial \xi} - v_0\frac{\partial \rho}{\partial \eta} + u_0\frac{\partial \rho}{\partial \xi} + v_0\frac{\partial \rho}{\partial \eta} + \nabla \cdot (\rho \boldsymbol{V}') = 0, \tag{C.16}$$
$$\nabla \cdot (\rho \boldsymbol{V}') = 0. \tag{C.17}$$

Applying the same transformation to (C.2) projected on x-axis:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + \frac{1}{\rho}\frac{\partial p}{\partial x} = 0, \tag{C.18}$$
$$\frac{\partial u'}{\partial t} + (u_0 + u')\frac{\partial u'}{\partial x} + (v_0 + v')\frac{\partial u'}{\partial y} + \frac{1}{\rho}\frac{\partial p}{\partial x} = 0, \tag{C.19}$$
$$\frac{\partial u'}{\partial \xi}\frac{\partial \xi}{\partial t} + \frac{\partial u'}{\partial \eta}\frac{\partial \eta}{\partial t} + (u_0 + u')\frac{\partial u'}{\partial x} + (v_0 + v')\frac{\partial u'}{\partial y} + \frac{1}{\rho}\frac{\partial p}{\partial x} = 0, \tag{C.20}$$
$$-u_0\frac{\partial u'}{\partial \xi} - v_0\frac{\partial u'}{\partial \eta} + (u_0 + u')\frac{\partial u'}{\partial x} + (v_0 + v')\frac{\partial u'}{\partial y} + \frac{1}{\rho}\frac{\partial p}{\partial x} = 0, \tag{C.21}$$
$$-u_0\frac{\partial u'}{\partial \xi} - v_0\frac{\partial u'}{\partial \eta} + (u_0 + u')\frac{\partial u'}{\partial \xi}\frac{\partial \xi}{\partial x} + (v_0 + v')\frac{\partial u'}{\partial \eta}\frac{\partial \eta}{\partial y} + \frac{1}{\rho}\frac{\partial p}{\partial x} = 0, \tag{C.22}$$
$$-u_0\frac{\partial u'}{\partial \xi} - v_0\frac{\partial u'}{\partial \eta} + (u_0 + u')\frac{\partial u'}{\partial \xi} + (v_0 + v')\frac{\partial u'}{\partial \eta} + \frac{1}{\rho}\frac{\partial p}{\partial x} = 0, \tag{C.23}$$
$$u'\frac{\partial u'}{\partial \xi} + v'\frac{\partial u'}{\partial \eta} + \frac{1}{\rho}\frac{\partial p}{\partial x} = 0, \tag{C.24}$$
$$u'\frac{\partial u'}{\partial x} + v'\frac{\partial u'}{\partial y} + \frac{1}{\rho}\frac{\partial p}{\partial x} = 0. \tag{C.25}$$

Applying the same transformation to (C.2) projected on y-axis, a similar demonstration gives:

$$u'\frac{\partial v'}{\partial x} + v'\frac{\partial v'}{\partial y} + \frac{1}{\rho}\frac{\partial p}{\partial y} = 0. \tag{C.26}$$

All in all, (C.25) and (C.26) can be written in a vector form:

$$\boldsymbol{V}' \cdot \nabla \boldsymbol{V}' + \frac{\nabla p}{\rho} = 0. \tag{C.27}$$

Since the flow is homentropic, one obtains:

$$p = \frac{p_0}{T_0^{\gamma/\gamma-1}} T^{\gamma/\gamma-1}, \tag{C.28}$$

$$\frac{\partial p}{\partial x} = \frac{p_0}{T_0^{\gamma/\gamma-1}} \frac{\partial T^{\gamma/\gamma-1}}{\partial x}, \tag{C.29}$$

$$\frac{\partial p}{\partial x} = \frac{p_0}{T_0^{\gamma/\gamma-1}} \frac{\gamma}{\gamma-1} T^{\gamma/\gamma-1-1} \frac{\partial T}{\partial x}, \tag{C.30}$$

$$\frac{\partial p}{\partial x} = \frac{p}{T^{\gamma/\gamma-1}} \frac{\gamma}{\gamma-1} T^{\gamma/\gamma-1-1} \frac{\partial T}{\partial x}, \tag{C.31}$$

$$\frac{\partial p}{\partial x} = \frac{p}{T} \frac{\gamma}{\gamma-1} \frac{\partial T}{\partial x}, \tag{C.32}$$

Using the perfect gas law $p = \rho R_{gas} T$, one obtains:

$$\frac{1}{\rho} \frac{\partial p}{\partial x} = \frac{\gamma R_{gas}}{\gamma-1} \frac{\partial T}{\partial x}. \tag{C.33}$$

Analogously, one obtains:

$$\frac{1}{\rho} \frac{\partial p}{\partial y} = \frac{\gamma R_{gas}}{\gamma-1} \frac{\partial T}{\partial y}. \tag{C.34}$$

It is now possible to rewrite the pressure gradient divided by the density:

$$\frac{\nabla p}{\rho} = \frac{\gamma R_{gas}}{\gamma-1} \nabla T'. \tag{C.35}$$

Finally, the Euler equations are reduced to:

$$\nabla \cdot \left(\rho \boldsymbol{V}'\right) = 0, \tag{C.36}$$

$$\boldsymbol{V}' \cdot \nabla \boldsymbol{V}' + \frac{\gamma R_{gas}}{\gamma-1} \nabla T' = 0. \tag{C.37}$$

Expressing in polar coordinates (r, θ), one obtains:

$$\boldsymbol{V}' = v_r' \boldsymbol{e_r} + v_\theta' \boldsymbol{e_\theta}, \tag{C.38}$$

$$T' = T'\left(r, \theta\right), \tag{C.39}$$

$$\rho = \rho\left(r, \theta\right), \tag{C.40}$$

$$p = p\left(r, \theta\right). \tag{C.41}$$

By choosing all variables independent of θ and the perturbation velocity such as

$$v_r' = 0, \tag{C.42}$$

$$v_\theta' = f\left(r\right), \tag{C.43}$$

it trivially satisfies

$$\nabla \cdot \left(\rho \boldsymbol{V}'\right) = \frac{1}{r} \frac{\partial}{\partial r} \left[r \rho\left(r\right) v_r'\right] + \frac{1}{r} \frac{\partial}{\partial \theta} \left[\rho\left(r\right) v_\theta'\right], \tag{C.44}$$

$$= \frac{1}{r} \frac{\partial}{\partial \theta} \left[\rho\left(r\right) f\left(r\right)\right], \tag{C.45}$$

$$= 0. \tag{C.46}$$

Indeed, $v'_r = 0$ and $v'_\theta = f(r)$ is independent from θ. Let us compute $\boldsymbol{V}' \cdot \nabla \boldsymbol{V}'$ in polar coordinates:

$$\boldsymbol{V}' \cdot \nabla \boldsymbol{V}' = \left[v'_r \frac{\partial v'_r}{\partial r} + \frac{v'_\theta}{r} \frac{\partial v'_r}{\partial \theta} - \frac{[v'_\theta]^2}{r} \right] \boldsymbol{e_r} + \left[v'_r \frac{\partial v'_\theta}{\partial r} + \frac{v'_\theta}{r} \frac{\partial v'_\theta}{\partial \theta} - \frac{v'_\theta v'_r}{r} \right] \boldsymbol{e_\theta}, \tag{C.47}$$

$$= -\frac{[v'_\theta]^2}{r} \boldsymbol{e_r}, \tag{C.48}$$

$$= -\frac{[f(r)]^2}{r} \boldsymbol{e_r}. \tag{C.49}$$

Finally, (C.37) simply becomes:

$$\frac{[f(r)]^2}{r} = \frac{\gamma R_{gas}}{\gamma - 1} \frac{\mathrm{d}T'}{\mathrm{d}r}. \tag{C.50}$$

Integrating this equation from $+\infty$ to r, one obtains:

$$T' = \frac{\gamma - 1}{\gamma R_{gas}} \int_{+\infty}^{r} \frac{[f(r)]^2}{r} \mathrm{d}r. \tag{C.51}$$

Now, the form of $f(r)$ needs to be chosen so that this can be fully integrated. In particular, when r tends to infinity, the perturbation is assumed to vanish. One chooses:

$$f(r) = \frac{\beta V_0 r}{R_c} \exp\left[-\frac{r^2}{2R_c^2} \right], \tag{C.52}$$

with $V_0^2 = u_0^2 + v_0^2$ and β and R_c are two free parameters. Substituting (C.52) in (C.51), one obtains:

$$T' = \beta^2 V_0^2 \frac{\gamma - 1}{\gamma R_{gas}} \int_{+\infty}^{r} \frac{r}{R_c^2} \exp\left[-\frac{r^2}{R_c^2} \right] \mathrm{d}r, \tag{C.53}$$

$$= \frac{\beta^2 V_0^2}{2} \frac{\gamma - 1}{\gamma R_{gas}} \left[-\exp\left[-\frac{r^2}{R_c^2} \right] \right]_{+\infty}^{r}, \tag{C.54}$$

$$= -\frac{\beta^2 V_0^2}{2} \frac{\gamma - 1}{\gamma R_{gas}} \exp\left[-\frac{r^2}{R_c^2} \right]. \tag{C.55}$$

Finally, one obtains:

$$\boldsymbol{V} = \boldsymbol{V}_0 + v'_\theta \boldsymbol{e_\theta}, \tag{C.56}$$

$$= \boldsymbol{V}_0 + \frac{\beta V_0 r}{R_c} \exp\left[-\frac{r^2}{2R_c^2} \right] \boldsymbol{e_\theta}, \tag{C.57}$$

$$T = T_0 + T', \tag{C.58}$$

$$= T_0 - \frac{\beta^2 V_0^2}{2} \frac{\gamma - 1}{\gamma R_{gas}} \exp\left[-\frac{r^2}{R_c^2} \right]. \tag{C.59}$$

Using the following expressions:

$$x - x_0 - u_0 t = r \cos\theta, \tag{C.60}$$

$$y - y_0 - v_0 t = r \sin\theta, \tag{C.61}$$

$$\boldsymbol{e_\theta} \cdot \boldsymbol{e_x} = -\sin\theta, \tag{C.62}$$

$$\boldsymbol{e_\theta} \cdot \boldsymbol{e_y} = \cos\theta, \tag{C.63}$$

the analytical solution is expressed in cartesian coordinates:

$$u = u_0 - \frac{\beta V_0}{R_c}(y - y_0 - v_0 t)\exp\left[-\frac{(x - x_0 - u_0 t)^2 + (y - y_0 - v_0 t)^2}{2R_c^2}\right], \tag{C.64}$$

$$v = v_0 + \frac{\beta V_0}{R_c}(x - x_0 - u_0 t)\exp\left[-\frac{(x - x_0 - u_0 t)^2 + (y - y_0 - v_0 t)^2}{2R_c^2}\right], \tag{C.65}$$

$$T = T_0 - \frac{\beta^2 V_0^2}{2}\frac{\gamma - 1}{\gamma R_{gas}}\exp\left[-\frac{(x - x_0 - u_0 t)^2 + (y - y_0 - v_0 t)^2}{R_c^2}\right]. \tag{C.66}$$

In the specific case where $\boldsymbol{V}_0 = (U_0, 0)^\top$, t is equal to 0 to express the initial solution and the reference center $\boldsymbol{x}_0$ is chosen to be equal to $\boldsymbol{x}_c = (x_c, y_c)^\top$, one recovers the expressions of (4.19), (4.20) and (4.21):

$$u = U_0 - \frac{\beta U_0}{R_c}(y - y_c)\exp\left[-\frac{(x - x_c)^2 + (y - y_c)^2}{2R_c^2}\right], \tag{C.67}$$

$$v = \frac{\beta U_0}{R_c}(x - x_c)\exp\left[-\frac{(x - x_c)^2 + (y - y_c)^2}{2R_c^2}\right], \tag{C.68}$$

$$T = T_0 - \frac{\beta^2 U_0^2}{2}\frac{\gamma - 1}{\gamma R_{gas}}\exp\left[-\frac{(x - x_c)^2 + (y - y_c)^2}{R_c^2}\right]. \tag{C.69}$$

D

Order analysis for unsteady equations

"It is not knowledge, but the act of learning, not possession but the act of getting there, which grants the greatest enjoyment."

— Carl Friedriech Gauss (1777 - 1855)

D.1 Truncation error

Let us consider the one-dimensional conservation equation (D.1). W is the unknown. $\boldsymbol{F}$ is the flux vector. The following proof is general and the complete definition of the flux is not necessary. The general form of the equation reads:

$$\frac{\partial W}{\partial t} + \nabla \cdot \boldsymbol{F}\left(W, \nabla \boldsymbol{W}\right) = 0. \tag{D.1}$$

Let Ω_i be an open set, and let $\partial\Omega_i$ be its boundary. Let $\boldsymbol{n}$ be the unit outward normal vector of $\partial\Omega_i$. Integrating (D.1) on Ω_i, it comes:

$$\int_{\Omega_i} \frac{\partial W}{\partial t}\,\mathrm{d}\mathcal{V} + \int_{\partial\Omega_i} \boldsymbol{F}\cdot\boldsymbol{n}\,\mathrm{d}\mathcal{S} = 0. \tag{D.2}$$

We assume that the cell volume $\mathcal{V}$ and the interface area $\mathcal{S}$ are constant. Δx is the length of any cell in the x-direction. Discretizing (D.2) over a one-dimensional grid of uniform cells, and introducing the mean value $\overline{W}_i$ of W over Ω_i, one obtains:

$$\frac{\mathrm{d}}{\mathrm{d}t}\overline{W}_i + \frac{F_{i+1/2} - F_{i-1/2}}{\Delta x} = 0, \tag{D.3}$$

where interface fluxes $F_{i+1/2}$ and $F_{i-1/2}$ are unknown. Since the mesh is kept unchanged during the time process, the equation (D.3) is the one that is solved using the Finite Volume Method. Now, let us suppose that one is able to find an extrapolation formula for the interface fluxes of order p:

$$F_{i+1/2} = F^{\mathrm{num}}_{i+1/2} + \mathcal{O}\left(\Delta x^p\right), \tag{D.4}$$

$$F_{i-1/2} = F^{\mathrm{num}}_{i-1/2} + \mathcal{O}\left(\Delta x^p\right). \tag{D.5}$$

Integrating in time by the Euler explicit algorithm, one obtains:

$$\frac{\overline{W}_i^{n+1} - \overline{W}_i^{n}}{\Delta t} + \frac{F^{\mathrm{num}}_{i+1/2} - F^{\mathrm{num}}_{i-1/2}}{\Delta x} = 0, \tag{D.6}$$

where Δt is the time step. The interface fluxes are computed with the states at the time step n. Computing the truncation error $\mathcal{E}\left(\Delta t, \Delta x\right)$ given by the difference between (D.3) and (D.6):

$$\mathcal{E}\left(\Delta t, \Delta x\right) = \frac{\mathrm{d}}{\mathrm{d}t}\overline{W}_i - \frac{\overline{W}_i^{n+1} - \overline{W}_i^{n}}{\Delta t} + \frac{F_{i+1/2} - F_{i-1/2}}{\Delta x} - \frac{F^{\mathrm{num}}_{i+1/2} - F^{\mathrm{num}}_{i-1/2}}{\Delta x}. \tag{D.7}$$

Simplifying, one finds:

$$\mathcal{E}\left(\Delta t, \Delta x\right) = \mathcal{O}\left(\Delta t\right) + \mathcal{O}\left(\Delta x^{p-1}\right). \tag{D.8}$$

As a consequence, the truncation error of the spatial discretization is $(p-1)$-th order accurate. It must also be highlighted that for some configurations, the $p-$th order of accuracy can be recovered on uniform grids. This is due to the fact that one can obtain a $p+1$-th accurate flux balance $F_{i+1/2} - F_{i-1/2}$. Indeed, the error term in Δx^p can be identical for $F_{i+1/2}$ and $F_{i-1/2}$ and the contribution vanishes. Examples of such a situation are given in Sec. 6.2, Sec. 6.3 and Sec. 6.4 by equations (6.4), (6.13) and (6.22). This feature then gives a truncation error of the spatial discretization which is p-th order accurate with an extrapolation formula for the interface fluxes of order p. Obviously, this feature is lost on general unstructured grids.

D.2 Discretization error

However, in practice, the measure of accuracy is generally performed by computing the discrete ℓ^2-norm of the discretization error $\mathcal{D}\left(\Delta t, \Delta x\right)$. The discretization for the cell i at time step n is the difference between the exact solution $\overline{V}_i^n$ and its numerical approximation $\overline{W}_i^n$:

$$\mathcal{D}_i^n\left(\Delta t, \Delta x\right) = \overline{V}_i^n - \overline{W}_i^n. \tag{D.9}$$

Let us define the progression for $k \in [1, n]$:

$$\mathcal{A}_i^k = \mathcal{D}_i^k - \mathcal{D}_i^{k-1}, \tag{D.10}$$

$$= \left(\overline{V}_i^k - \overline{W}_i^k\right) - \left(\overline{V}_i^{k-1} - \overline{W}_i^{k-1}\right), \tag{D.11}$$

$$= \left(\overline{V}_i^k - \overline{V}_i^{k-1}\right) - \left(\overline{W}_i^k - \overline{W}_i^{k-1}\right). \tag{D.12}$$

Besides,

$$\frac{\mathrm{d}}{\mathrm{d}t}\overline{V}_i = \frac{\overline{V}_i^n - \overline{V}_i^{n-1}}{\Delta t} + \mathcal{O}\left(\Delta t\right), \tag{D.13}$$

$$\overline{W}_i^n - \overline{W}_i^{n-1} = -\frac{\Delta t}{\Delta x}\left(F_{i+1/2}^{\mathrm{num}} - F_{i-1/2}^{\mathrm{num}}\right), \tag{D.14}$$

where the interface fluxes are computed with the states at the time step $n-1$. Substituting (D.13) and (D.14) in (D.12), one obtains:

$$\mathcal{A}_i^k = \Delta t\left(\frac{\mathrm{d}}{\mathrm{d}t}\overline{V}_i + \mathcal{O}\left(\Delta t\right)\right) + \frac{\Delta t}{\Delta x}\left(F_{i+1/2}^{\mathrm{num}} - F_{i-1/2}^{\mathrm{num}}\right), \tag{D.15}$$

$$= \frac{\Delta t}{\Delta x}\left(-\left(F_{i+1/2} - F_{i-1/2}\right) + F_{i+1/2}^{\mathrm{num}} - F_{i-1/2}^{\mathrm{num}} + \mathcal{O}\left(\Delta t\right)\right), \tag{D.16}$$

$$= \frac{\Delta t}{\Delta x}\left(\mathcal{O}\left(\Delta x^p\right) + \mathcal{O}\left(\Delta t\right)\right), \tag{D.17}$$

$$= \Delta t\left(\mathcal{O}\left(\Delta x^{p-1}\right) + \mathcal{O}\left(\Delta t\right)\right). \tag{D.18}$$

The computation of the sum of all $\mathcal{A}_i^k$ gives:

$$\sum_{k=1}^{n} \mathcal{A}_i^k = \mathcal{D}_i^n - \mathcal{D}_i^0, \tag{D.19}$$

and assuming that the initial numerical approximation $\overline{W}_i^0$ is the initial exact solution $\overline{V}_i^0$, it comes easily:

$$\sum_{k=1}^{n} \mathcal{A}_i^k = \mathcal{D}_i^n. \tag{D.20}$$

Using (D.18) and (D.20), one obtains:

$$\mathcal{D}_i^n = \sum_{k=1}^{n} \mathcal{A}_i^k \tag{D.21}$$

$$= \sum_{k=1}^{n} \Delta t \left(\mathcal{O}\left(\Delta x^{p-1}\right) + \mathcal{O}\left(\Delta t\right)\right), \tag{D.22}$$

$$= n\Delta t \left(\mathcal{O}\left(\Delta x^{p-1}\right) + \mathcal{O}\left(\Delta t\right)\right), \tag{D.23}$$

$$= t^n \left(\mathcal{O}\left(\Delta x^{p-1}\right) + \mathcal{O}\left(\Delta t\right)\right), \tag{D.24}$$

$$= \mathcal{O}\left(\Delta x^{p-1}\right) + \mathcal{O}\left(\Delta t\right), \tag{D.25}$$

where t^n is the physical time of the whole computation. Summing the discretization error $\mathcal{D}_i^n$ for the cell i for all N cell of the considered mesh, one finally finds the discretization error $\mathcal{D}^n\left(\Delta t, \Delta x\right)$:

$$\mathcal{D}^n\left(\Delta t, \Delta x\right) = \mathcal{O}\left(\Delta x^{p-1}\right) + \mathcal{O}\left(\Delta t\right). \tag{D.26}$$

As a consequence, the discretization error of the spatial discretization is $(p-1)$-th order accurate. Again, it must also be highlighted that for some configurations, the $p-$th order of accuracy can be recovered on uniform grids. This is due to the fact that one can obtain a $p+1$-th accurate flux balance $F_{i+1/2} - F_{i-1/2}$. Indeed, the error term in Δx^p can be identical for $F_{i+1/2}$ and $F_{i-1/2}$ and the contribution vanishes. Examples of such a situation are given in Sec. 6.2, Sec. 6.3 and Sec. 6.4 by equations (6.4), (6.13) and (6.22). This feature then gives a discretization error of the spatial discretization which is p-th order accurate with an extrapolation formula for the interface fluxes of order p. Obviously, this feature is lost on general unstructured grids. However, supra-convergence phenomenon can occur in very specific cases. Indeed, as stated in [188], "a qualitative explanation of supra-convergence observed for [Finite Volume discretization] schemes on irregular grids was suggested by Giles [189]. Giles' arguments can be illustrated as follows. First consider a node-centered [Finite Volume discretization] scheme on a sequence of regular grids. On such grids, truncation and discretization errors would converge with the same design order. [This is confirmed by (D.8) and (D.26).] Now consider effects of a local irregularity introduced, for example, by shifting one node. The node shift leads to changes in faces (and corresponding fluxes) of the control volume containing this node and its neighboring control volumes. Fluxes outside of this immediate neighborhood remain unchanged. Each (changed) flux at an internal face contributes equal and opposite amounts to the undivided truncation errors on both sides of the face. Thus, the change introduced into the local truncation error can be large (the change in a flux divided by the control volume), but the mean value (and, often, higher moments) of the changes introduced into the undivided truncation errors is zero. If the convolution kernel is sufficiently smooth, e.g., for all nodes outside of the immediate neighborhood of the perturbed node, the weighting factors do not vary too much over the control volumes with perturbed truncation errors, then the discretization error changes are close to zero outside of this immediate neighborhood. Even within this neighborhood, the discretization error changes are small comparing with the changes in the truncation errors because of the averaging effect."

E

About the work unit evaluation

"Science is a differential equation. Religion is a boundary condition."

— Alan Mathison Turing (1912 - 1954)

In this appendix, the evaluation of the work unit time τ_w is detailed. The cost of a computation should be expressed in work units. The work unit time is used to assess the computation time regardless of the machine. But, obviously, it should be measure on each machine. Here, it was measured on nodes composed of 12 bi-socket Intel Haswell (E5-2680v3) cores at 2.5 Ghz with 64 GB RAM. To estimate the work unit time, 1000 TauBench runs were launched and the computation time for each run is reported in Fig. E.1. TauBench was launched with the command

mpirun -np 1 ./TauBench -n 250000 -s 10

where the option *-n* specifies the number of grid points and the option *-s* specifies the number of pseudo-steps. The TauBench is an unstructured grid benchmark. The respective kernels are derived from Tau [190, 191, 192], a Navier Stokes solver of DLR. Finally, the mean work unit time is 7633 $[ms]$ with a standard deviation of 18 $[ms]$.

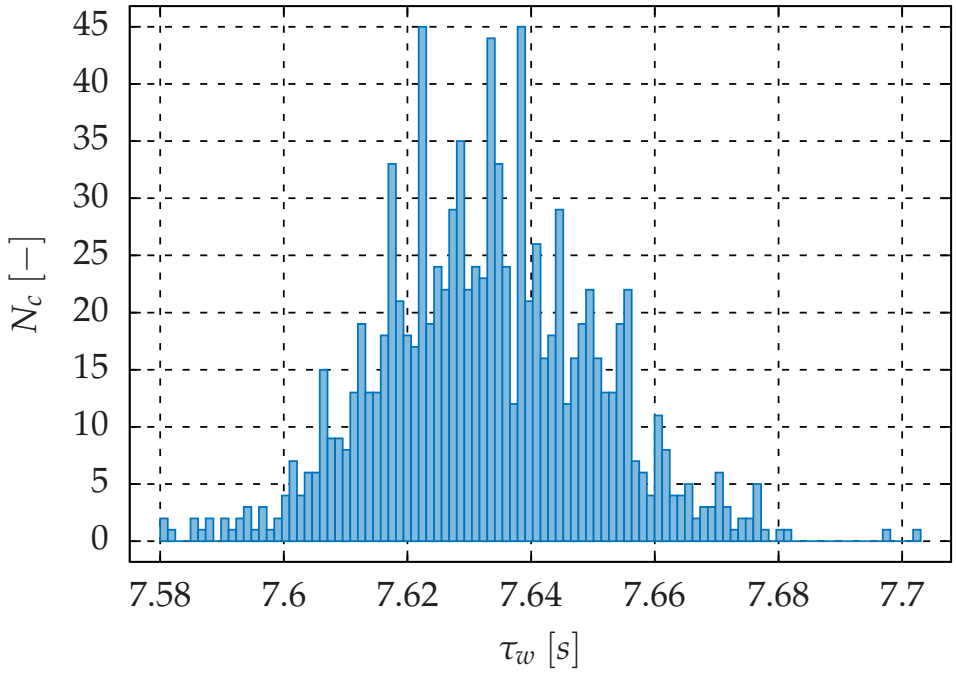


Fig. E.1. *Evaluation of the work unit time τ_w with several computations. The mean work unit time is* 7633 $[ms]$ *with a standard deviation of* 18 $[ms]$.

VIII Bibliography

[1] M. Darecki, C. Edelstenne, T. Enders, E. Fernandez, P. Hartman, J.-P. Herteman, M. Kerkloh, I. King, P. Ky, M. Mathieu, G. Orsi, G. Schotman, C. Smith, and J.-D. Wörner. Flightpath 2050 Europe's vision for aviation. *Tech. report*. European Commission. (2011). (Cited on page 3). http://ec.europa.eu/transport/sites/transport/files/modes/air/doc/flightpath2050.pdf.

[2] Z.J. Wang, K. Fidkowski, R. Abgrall, F. Bassi, D. Caraeni, A. Cary, H. Deconinck, R. Hartmann, K. Hillewaert, H.T. Huynh, N. Kroll, G. May, P.-O. Persson, B. Van Leer, and M. Visbal. High-order CFD Methods: current status and perspective. *Int. J. Numer. Meth. Fluids* 72 (8) (2013), 811–845. (Cited on pages 7, 22, 60, 85, 87, 88). http://dx.doi.org/10.1002/fld.3767.

[3] L. Cambier, S. Heib, and S. Plot. The Onera elsA CFD software: input from research and feedback from industry. *Mech. Ind.* 14 (3) (2013), 159–174. (Cited on pages 7, 22, 161). http://dx.doi.org/10.1051/meca/2013056.

[4] G. Puigt, M. Gazaix, M. Montagnac, M.-C. Le Pape, M. de la Llave Plata, C. Marmignon, J.-F. Boussuge, and V. Couaillier. Development of a new hybrid compressible solver inside the CFD elsA software. 20th AIAA Computational Fluid Dynamics Conference. Honolulu, Hawaii, USA. AIAA 2011-3379 (2011). (Cited on pages 7, 14, 22) http://dx.doi.org/10.2514/6.2011-3379.

[5] R. Vichnevetsky and J.B. Bowles. Fourier analysis of numerical approximations of hyperbolic equations. Edited by J.A. Nohel. SIAM Studies in Applied Mathematics (1982). (Cited on pages 7, 34, 47, 48, 130). http://dx.doi.org/10.1137/1.9781611970876.

[6] A. Fosso, H. Deniau, F. Sicot, and P. Sagaut. Curvilinear Finite Volume schemes using high-order compact interpolation. *J. Comput. Phys.* 229 (13) (2010), 5090–5122. (Cited on pages 7, 15, 89, 161, 166, 167). http://dx.doi.org/10.1016/j.jcp.2010.03.027.

[7] A. Cassagne, J.-F. Boussuge, N. Villedieu, G. Puigt, I. D'Ast, and A. Genot. JAGUAR: a new CFD code dedicated to massively parallel high-order LES computations on complex geometry. 50th 3AF International Conference on Applied Aerodynamics. Toulouse, France. (2015). (Cited on pages 8, 161). https://www.researchgate.net/profile/Guillaume_Puigt/publication/271014084_JAGUAR_a_new_CFD_code_dedicated_to_massively_parallel_high--order_LES_computations_on_complex_geometry/links/54e755af0cf2cd2e02934f93.pdf?origin=publication_list.

[8] S. Chapman and T.G. Cowling. The Mathematical theory of non-uniform gases: an account of the kinetic theory of viscosity, thermal conduction and diffusion in gases. Cambridge University Press (1970). (Cited on page 8). https://books.google.fr/books?id=Cbp5JP2OTrwC.

[9] L. Boltzmann. Weitere Studien über das Wärmegleichgewicht unter Gasmolekülen. Vieweg+Teubner Verlag (1970), 115–225. (Cited on page 8). http://dx.doi.org/10.1007/978-3-322-84986-1_3.

[10] L. Boltzmann. Further studies on the thermal equilibrium of gas molecules. *The Kinetic Theory Of Gases* 1 (2003), 262–349. (Cited on page 8). http://dx.doi.org/10.1142/9781848161337_0015.

[11] G.R. McNamara and G. Zanetti. Use of the Boltzmann equation to simulate lattice-gas automata. *Phys. Rev. Lett.* 61 (20) (1988), 2332–2335. (Cited on page 8). http://dx.doi.org/10.1103/PhysRevLett.61.2332.

[12] X. He and L. Luo. A priori derivation of the Lattice Boltzmann equation. *Phys. Rev. E* 55 (6) (1997), R6333–R6336. (Cited on page 8). http://dx.doi.org/10.1103/PhysRevE.55.R6333.

[13] X. He and L. Luo. Theory of the Lattice Boltzmann method: from the Boltzmann equation to the Lattice Boltzmann equation. *Phys. Rev. E* 56 (6) (1997), 6811–6817. (Cited on page 8). http://dx.doi.org/10.1103/PhysRevE.56.6811.

[14] T. Krüger, H. Kusumaatmaja, A. Kuzmin, O. Shardt, G. Silva, and E.M. Viggen. The Lattice Boltzmann Method: principles and practice. Springer International Publishing (2017). (Cited on page 8). http://dx.doi.org/10.1007/978-3-319-44649-3.

[15] S. Marié, D. Ricot, and P. Sagaut. Comparison between lattice Boltzmann method and Navier-Stokes high order schemes for computational aeroacoustics. *J. Comput. Phys.* 228 (4) (2009), 1056–1070. (Cited on page 8). http://dx.doi.org/10.1016/j.jcp.2008.10.021.

[16] D. Lagrava, O. Malaspinas, J. Latt, and B. Chopard. Advances in multi-domain Lattice Boltzmann grid refinement. *J. Comput. Phys.* 231 (14) (2012), 4808–4822. (Cited on page 8). http://dx.doi.org/10.1016/j.jcp.2012.03.015.

[17] Y.H. Qian and S.A. Orszag. Lattice BGK Models for the Navier-Stokes equation: nonlinear deviation in compressible regimes. *EPL* 21 (3) (1993), 255. (Cited on page 8). http://dx.doi.org/10.1209/0295-5075/21/3/001.

[18] J. Vanharen, G. Puigt, and M. Montagnac. Theoretical and numerical analysis of nonconforming grid interface for unsteady flows. *J. Comput. Phys.* 285 (2015), 111–132. (Cited on page 13). http://dx.doi.org/10.1016/j.jcp.2015.01.013.

[19] J. Vanharen, G. Puigt, and M. Montagnac. Two-dimensional spectral analysis of nonconforming grid interface. Emphasis on unsteady flows. 50th 3AF International Conference on Applied Aerodynamics. (FP36-2015-vanharen). Toulouse, France. (2015). (Cited on page 13). http://www.cerfacs.fr/~cfdbib/repository/TR_CFD_15_23.pdf.

[20] D.J. Mavriplis. Unstructured grid techniques. *Ann. Rev. Fluid Mech.* 29 (1) (1997), 473–514. (Cited on page 14). http://dx.doi.org/10.1146/annurev.fluid.29.1.473.

[21] J.L. Steger, F.C. Dougherty, and J.A. Benek. A Chimera Grid Scheme. Advances in grid generation. Proceedings of the Applied Mathematics, Bioengineering, and Fluids Enginneering Conference. Edited by K.N. Ghia and U. Ghia. Houston (TX), USA. The American Society of Mechanical Engineers volume 5 (1983), 59–69. (Cited on page 14). https://ntrs.nasa.gov/search.jsp?R=19840028795.

[22] B. Landmann and M. Montagnac. A highly automated parallel Chimera method for overset grids based on the implicit hole cutting technique. *Int. J. Numer. Meth. Fluids* 66 (6) (2011), 778–804. (Cited on page 14). http://dx.doi.org/10.1002/fld.2292.

[23] X. He, X. He, L. He, Z. Zhao, and L. Zhang. HyperFLOW: A structured/unstructured hybrid integrated computational environment for multi-purpose fluid simulation. *Procedia Eng.* 126 (2015), 645–649. (Cited on page 14). http://dx.doi.org/10.1016/j.proeng.2015.11.254.

[24] K.-H. Kao and M.-S. Liou. Advance in overset grid schemes - From Chimera to DRAGON grids. *AIAA J.* 33 (10) (1995), 1809–1815. (Cited on page 14). http://dx.doi.org/10.2514/3.12921.

[25] M.M. Rai. A conservative treatment of zonal boundaries for Euler equation calculations. *J. Comput. Phys.* 62 (2) (1986), 472–503. (Cited on page 14). http://dx.doi.org/10.1016/0021-9991(86)90141-5.

[26] S. Le Bras, H. Deniau, and C. Bogey. A flux reconstruction technique for non-conforming grid interfaces in aeroacoustic simulations. 22nd AIAA/CEAS Aeroacoustics Conference. (AIAA-2016-2972). Lyon, France. (2016). (Cited on page 14) http://dx.doi.org/10.2514/6.2016-2972.

[27] S. Deparis, D. Forti, P. Gervasio, and A. Quarteroni. INTERNODES: an accurate interpolation-based method for coupling the Galerkin solutions of PDEs on subdomains featuring non-conforming interfaces. *Comput. Fluids* 141 (2016), 22–41. (Cited on page 14). http://dx.doi.org/10.1016/j.compfluid.2016.03.033.

[28] L. Ramírez, C. Foulquié, X. Nogueira, S. Khelladi, J.-C. Chassaing, and I. Colominas. New high-resolution-preserving sliding mesh techniques for higher-order Finite Volume schemes. *Comput. Fluids* 118 (2015), 114–130. (Cited on page 14). http://dx.doi.org/10.1016/j.compfluid.2015.06.008.

[29] M.M. Rai. A relaxation approach to patched-grid calculations with the Euler equations. *J. Comput. Phys.* 66 (1) (1986), 99–131. (Cited on page 14). http://dx.doi.org/10.1016/0021-9991(86)90056-2.

[30] M.M. Rai. An implicit, conservative, zonal-boundary scheme for Euler equation calculations. *Comput. Fluids* 14 (3) (1986), 295–319. (Cited on page 14). http://dx.doi.org/10.1016/0045-7930(86)90027-7.

[31] M.M. Rai. Three-dimensional Navier-Stokes simulations of turbine rotor-stator interaction; Part II - Results. *J. Propul. Power* 5 (3) (1989), 312–319. (Cited on page 14). http://dx.doi.org/10.2514/3.51296.

[32] M.M. Rai. Three-dimensional Navier-Stokes simulations of turbine rotor-stator interaction. Part I - Methodology. *J. Propul. Power* 5 (3) (1989), 305–311. (Cited on page 14). http://dx.doi.org/10.2514/3.23154.

[33] R.T. Biedron and J.L. Thomas. A generalized patched-grid algorithm with application to the F-18 forebody with actuated control strake. *Computing Systems in Engineering* 1 (2–4) (1990), 563–576. (Cited on page 14). http://dx.doi.org/10.1016/0956-0521(90)90036-K.

[34] B. Epstein and S. Peigin. Treatment of nonmatched grids for high-accuracy Navier-Stokes solutions. *AIAA J.* 48 (7) (2010), 1542–1553. (Cited on page 14). http://dx.doi.org/10.2514/1.J050300.

[35] C.L. Rumsey. Computation of acoustic waves through sliding-zone interfaces using an Euler/Navier-Stokes code. *AIAA J.* 35 (1996), 263–268. (Cited on page 14). http://dx.doi.org/10.2514/6.1996-1752.

[36] G. Fillola, M.-C. Le Pape, and M. Montagnac. Numerical simulations around wing control surfaces. Proceedings of 24th International Congress of the Aeronautical Sciences (ICAS). Yokohama, Japan. (2004). (Cited on page 14). http://www.icas.org/ICAS_ARCHIVE/ICAS2006/PAPERS/364.PDF.

[37] A. Lerat and Z.N. Wu. Stable conservative multidomain treatments for implicit Euler solvers. *J. Comput. Phys.* 123 (1) (1996), 45–64. (Cited on page 14). http://dx.doi.org/10.1006/jcph.1996.0004.

[38] J. Berland, C. Bogey, O. Marsden, and C. Bailly. High-order, low dispersive and low dissipative explicit schemes for multiple-scale and boundary problems. *J. Comput. Phys.* 224 (2) (2007), 637–662. (Cited on pages 14, 20). http://dx.doi.org/10.1016/j.jcp.2006.10.017.

[39] S.K. Lele. Compact finite difference schemes with spectral-like resolution. *J. Comput. Phys.* 103 (1) (1992), 16–42. (Cited on pages 15, 129, 131, 132). http://dx.doi.org/10.1016/0021-9991(92)90324-R.

[40] J.W. Kim and D.J. Lee. Optimized compact finite difference schemes with maximum resolution. *AIAA J.* 34 (5) (1996), 887–893. (Cited on page 15). http://dx.doi.org/10.2514/3.13164.

[41] G. Ashcroft and X. Zhang. Optimized prefactored compact schemes. *J. Comput. Phys.* 190 (2) (2003), 459–477. (Cited on page 15). http://dx.doi.org/10.1016/S0021-9991(03)00293-6.

[42] D.V. Nance, K.Viswanathan, and L.N. Sankar. Low-dispersion Finite Volume scheme for aeroacoustic applications. *AIAA J.* 35 (2) (1997), 255–262. (Cited on page 15). http://dx.doi.org/10.2514/2.115.

[43] T.K. Sengupta, G. Ganeriwal, and S. De. Analysis of central and upwind compact schemes. *J. Comput. Phys.* 192 (2) (2003), 677–694. (Cited on page 15). http://dx.doi.org/10.1016/j.jcp.2003.07.015.

[44] C. Bogey and C. Bailly. A family of low dispersive and low dissipative explicit schemes for flow and noise computations. *J. Comput. Phys.* 194 (1) (2004), 194–214. (Cited on pages 15, 55, 66, 87, 131, 132, 136, 137, 143). http://dx.doi.org/10.1016/j.jcp.2003.09.003.

[45] C.K.W. Tam and J.C. Webb. Dispersion-Relation-Preserving Finite Difference schemes for computational acoustics. *J. Comput. Phys.* 107 (2) (1993), 262–281. (Cited on pages 15, 130). http://dx.doi.org/10.1006/jcph.1993.1142.

[46] M. Popescu, W. Shyy, and M. Garbey. Finite Volume treatment of Dispersion-Relation-Preserving and optimized prefactored compact schemes for wave propagation. *J. Comput. Phys.* 210 (2) (2005), 705–729. (Cited on page 15). http://dx.doi.org/10.1016/j.jcp.2005.05.011.

[47] H.-O. Kreiss and G. Scherer. Finite Element and Finite Difference methods for hyperbolic partial differential equations. Mathematical Aspects of Finite Elements in Partial Differential Equations. Edited by C. de Boor. Academic Press (1974), 195–212. (Cited on page 15). http://dx.doi.org/10.1016/B978-0-12-208350-1.50012-1.

[48] B. Strand. Summation by parts for Finite Difference approximations for d/dx. *J. Comput. Phys.* 110 (1) (1994), 47–67. (Cited on page 15). http://dx.doi.org/10.1006/jcph.1994.1005.

[49] P. Olsson. Summation by parts, projections, and stability. I. *Math. Comp.* 64 (211) (1995), 1035–1065. (Cited on page 15). http://dx.doi.org/10.2307/2153482.

[50] D.J. Bodony, G. Zagaris, A. Reichert, and Q. Zhang. Provably stable overset grid methods for computational aeroacoustics. *J. Sound Vib.* 330 (17) (2011), 4161–4179. (Cited on page 15). http://dx.doi.org/10.1016/j.jsv.2011.02.010.

[51] M.H. Carpenter, J. Nordström, and D. Gottlieb. Revisiting and extending interface penalties for multi-domain Summation-By-Parts operators. *J. Sci. Comput.* 45 (1) (2010), 118–150. (Cited on page 15). http://dx.doi.org/10.1007/s10915-009-9301-5.

[52] K.W. Thompson. Time dependent boundary conditions for hyperbolic systems. *J. Comput. Phys.* 68 (1) (1987), 1–24. (Cited on page 15). http://dx.doi.org/10.1016/0021-9991(87)90041-6.

[53] T.J. Poinsot and S.K. Lele. Boundary conditions for direct simulations of compressible viscous flows. *J. Comput. Phys.* 101 (1) (1992), 104–129. (Cited on page 15). http://dx.doi.org/10.1016/0021-9991(92)90046-2.

[54] J.W. Kim and D.J. Lee. Generalized characteristic boundary conditions for computational aeroacoustics. *AIAA J.* 38 (11) (2000), 2040–2049. (Cited on page 15). http://dx.doi.org/10.2514/2.891.

[55] J.W. Kim and D.J. Lee. Characteristic interface conditions for multiblock high-order computation on singular structured grid. *AIAA J.* 41 (12) (2003), 2341–2348. (Cited on page 15). http://dx.doi.org/10.2514/2.6858.

[56] S. Bocquet, P. Sagaut, and J.-C. Jouhaud. A compressible wall model for Large-Eddy Simulation with application to prediction of aerothermal quantities. *Phys. Fluids* 24 (6) (2012), 065103. (Cited on page 15). http://dx.doi.org/10.1063/1.4729614.

[57] P.L. Roe. Approximate Riemann solvers, parameter vectors, and difference schemes. *J. Comput. Phys.* 43 (2) (1981), 357–372. (Cited on pages 15, 35, 66, 87). http://dx.doi.org/10.1016/0021-9991(81)90128-5.

[58] P.L. Roe and J. Pike. Efficient construction and utilisation of approximate Riemann solutions. Symposium on Computing Methods in Applied Sciences and Engineering, VI. Versailles, France. (1985), 499–518. (Cited on pages 15, 35, 66, 87). http://www.jackpike.co.uk/38.Efficient%20Construction%20and%20Utilisation%20of%20Approximate%20Riemann%20Solutions.pdf.

[59] R. Vichnevetsky. Energy and group velocity in semi-discretization of hyperbolic equations. *Math. Comput. Simul.* 23 (4) (1981), 333–343. (Cited on pages 34, 47). http://dx.doi.org/10.1016/0378-4754(81)90020-3.

[60] R. Vichnevetsky. Propagation through numerical mesh refinement for hyperbolic equations. *Math. Comput. Simul.* 23 (4) (1981), 344–353. (Cited on pages 34, 47). http://dx.doi.org/10.1016/0378-4754(81)90021-5.

[61] G. Yu, B. Yu, S. Sun, and W.-Q. Tao. Comparative study on triangular and quadrilateral meshes by a Finite Volume Method with a central difference scheme. *Numer. Heat Transfer, Part B: Fundamentals* 62 (4) (2012), 243–263. (Cited on page 75). http://dx.doi.org/10.1080/10407790.2012.709416.

[62] W. Sutherland. LII. The viscosity of gases and molecular force. *Philos. Mag. Series 5* 36 (223) (1893). (Cited on page 86). http://dx.doi.org/10.1080/14786449308620508.

[63] J.H. Williamson. Low-storage Runge-Kutta schemes. *J. Comput. Phys.* 35 (1) (1980), 48–56. (Cited on page 87). http://dx.doi.org/http://dx.doi.org/10.1016/0021-9991(80)90033-9.

[64] W.M. van Rees, A. Leonard, D.I. Pullin, and P. Koumoutsakos. A comparison of vortex and pseudo-spectral methods for the simulation of periodic vortical flows at high Reynolds numbers. *J. Comput. Phys.* 230 (8) (2011), 2794–2805. (Cited on page 87). http://dx.doi.org/10.1016/j.jcp.2010.11.031.

[65] B. van Leer. Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov's method. *J. Comput. Phys.* 32 (1) (1979), 101–136. (Cited on pages 94, 99, 102). http://dx.doi.org/10.1016/0021-9991(79)90145-1.

[66] G. Puigt, P. Cayot, J. Vanharen, and P. Sagaut. Unstructured interface gradient for the diffusion operator. *Under preparation for J. Sci. Comput.* (2017). (Cited on pages 99, 114).

[67] J. Droniou and C. Le Potier. Construction and convergence study of schemes preserving the elliptic local maximum principle. *SIAM J. Numer. Anal.* 49 (2) (2011), 459–490. (Cited on page 100). http://dx.doi.org/10.1137/090770849.

[68] C. Cancès, M. Cathala, and C. Le Potier. Monotone corrections for generic cell-centered finite volume approximations of anisotropic diffusion equations. *Numer. Math.* 125 (3) (2013), 387–417. (Cited on page 100). http://dx.doi.org/10.1007/s00211-013-0545-5.

[69] C. Le Potier. A nonlinear finite volume scheme satisfying maximum and minimum principles for diffusion operator. *Int. J. Finite Vol.* 6 (2) (2009), 1–20. (Cited on page 100). https://hal.archives-ouvertes.fr/hal-01116968/document.

[70] C. Le Potier. Correction non linéaire et principe du maximum pour la discrétisation d'opérateurs de diffusion avec des schémas volumes finis. *C. R. Math. Acad. Sci. Paris* 348 (11-12) (2010), 691–695. (Cited on page 100). http://dx.doi.org/10.1016/j.crma.2010.04.017.

[71] C. Le Potier. Correction non linéaire d'ordre 2 et principe du maximum pour la discrétisation d'opérateurs de diffusion. *C. R. Math. Acad. Sci. Paris* 352 (11) (2014), 947–952. (Cited on page 100). http://dx.doi.org/10.1016/j.crma.2014.08.010.

[72] E. Sozer, C. Brehm, and C.C. Kiris. Gradient calculation methods on arbitrary polyhedral unstructured meshes for cell-centered CFD Solvers. 52nd AIAA Aerospace Sciences Meeting. (AIAA-2014-1440). National Harbor, Maryland, USA. (2014). (Cited on pages 101, 102, 106) http://dx.doi.org/10.2514/6.2014-1440.

[73] M. Galle. Unstructured viscous flow solution using adaptive hybrid grids. *Tech. report*. NASA Langley Research Center, ICASE/LaRC Workshop on Adaptive Grid Methods. (1995), 171–179. (Cited on page 101). https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/19960011646.pdf.

[74] P.I. Crumpton, P. Moinier, and M.B. Giles. An unstructured algorithm for high Reynolds number flows on highly stretched grids. *Numerical methods in laminar and turbulent flow* (1997), 561–572. (Cited on page 101). https://pdfs.semanticscholar.org/2ab6/34e33a469dd2455359b02eada7cfa3b2debb.pdf.

[75] A. Cary, A. Dorgan, and M. Mani. Towards accurate flow predictions using unstructured meshes. 19th AIAA Computational Fluid Dynamics. (AIAA-2009-3650). San Antonio, Texas, USA. (2009). (Cited on page 102) http://dx.doi.org/10.2514/6.2009-3650.

[76] W.K. Anderson and D.L. Bonhaus. An implicit upwind algorithm for computing turbulent flows on unstructured grids. *Comput. Fluids* 23 (1) (1994), 1–21. (Cited on page 102). http://dx.doi.org/10.1016/0045-7930(94)90023-x.

[77] C. Ollivier-Gooch and M. Van Altena. A high-order-accurate unstructured mesh Finite Volume scheme for the advection-diffusion equation. *J. Comput. Phys.* 181 (2) (2002), 729–752. (Cited on page 102). http://dx.doi.org/10.1006/jcph.2002.7159.

[78] D.J. Mavriplis. Revisiting the Least-Squares procedure for gradient reconstruction on unstructured meshes. 16th AIAA Computational Fluid Dynamics Conference. (AIAA-2003-3986). Orlando, Florida, USA. (2003). (Cited on page 102) http://dx.doi.org/10.2514/6.2003-3986.

[79] A. Jalali, M. Sharbatdar, and C. Ollivier-Gooch. Accuracy analysis of unstructured Finite Volume discretization schemes for diffusive fluxes. *Comput. Fluids* 101 (2014), 220–232. (Cited on page 102). http://dx.doi.org/10.1016/j.compfluid.2014.06.008.

[80] C.B. Sejekan and C.F. Ollivier-Gooch. Improving Finite Volume diffusive fluxes through better reconstruction. *Comput. Fluids* 139 (2016), 216–232. (Cited on pages 102, 103). http://dx.doi.org/10.1016/j.compfluid.2016.08.002.

[81] E. Shima, K. Kitamura, and T. Haga. Green-Gauss/Weighted-Least-Squares hybrid gradient reconstruction for arbitrary polyhedra unstructured grids. *AIAA J.* 51 (11) (2013), 2740–2747. (Cited on page 102). http://dx.doi.org/10.2514/1.j052095.

[82] E.H. Moore. On the reciprocal of the general algebraic matrix. *Bull. Am. Math. Soc.* 26 (1920), 394–395. (Cited on pages 103, 113).

[83] R. Penrose and J.A. Todd. A generalized inverse for matrices. *Math. Proc. Cambridge Philos. Soc.* 51 (3) (1955), 406–413. (Cited on pages 103, 113). http://dx.doi.org/10.1017/s0305004100030401.

[84] G. Pont. Self adaptive turbulence models for unsteady compressible flows. PhD thesis. ENSAM, (2015). (Cited on pages 103, 125). https://pastel.archives-ouvertes.fr/tel-01310365/document.

[85] B. Maugars. Méthodes de volumes finis d'ordre élevé en maillages non coïncident pour les écoulements dans les turbomachines. PhD thesis. ENSAM, (2016). (Cited on pages 103, 116). http://www.theses.fr/s147499.

[86] F. Haider, J.-P. Croisille, and B. Courbet. Stability analysis of the cell centered finite-volume MUSCL method on unstructured grids. *Numer. Math.* 113 (2009), 555–600. (Cited on page 103). http://dx.doi.org/10.1007/s00211-009-0242-6.

[87] F. Haider. Discrétisation en maillage non structuré général et applications LES. PhD thesis. Paris VI University, (2009). (Cited on pages 103, 115). https://www.theses.fr/2009PA066171.

[88] W.J. Coirier. An adaptively-refined, cartesian, cell-based scheme for the Euler and Navier-Stokes equations. PhD thesis. University of Michigan, (1994). (Cited on pages 103, 104). http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.35.4703&rep=rep1&type=pdf.

[89] Y. Coudière, J.-P. Vila, and P. Villedieu. Convergence rate of a Finite Volume scheme for a two dimensional convection-diffusion problem. *ESAIM Math. Model. Num.* 33 (3) (1999), 493–516. (Cited on page 104). http://dx.doi.org/10.1051/m2an:1999149.

[90] C. Le Potier. A Finite Volume method for the approximation of highly anisotropic diffusion operators on unstructured meshes. Marrakech, Marocco. (2005). (Cited on page 104).

[91] F. Hermeline. A Finite Volume Method for the Approximation of Diffusion Operators on Distorted Meshes. *J. Comput. Phys.* 160 (2) (2000), 481–499. (Cited on page 104). http://dx.doi.org/10.1006/jcph.2000.6466.

[92] K. Domelevo and P. Omnes. A finite volume method for the Laplace equation on almost arbitrary two-dimensional grids. *M2AN Math. Model. Numer. Anal.* 39 (6) (2005), 1203–1249. (Cited on page 104). http://dx.doi.org/10.1051/m2an:2005047.

[93] J. Droniou. Finite volume schemes for diffusion equations: Introduction to and review of modern methods. *Math. Models Methods in Appl. Sci.* 24 (08) (2014), 1575–1619. (Cited on page 105). http://dx.doi.org/10.1142/s0218202514400041.

[94] I. Aavatsmark. An introduction to multipoint flux approximations for quadrilateral grids". *Comput. Geosci.* 6 (3) (2002), 405–432. (Cited on page 105). http://dx.doi.org/10.1023/a:1021291114475.

[95] I. Aavatsmark, T. Barkve, O. Bøe, and T. Mannseth. Discretization on unstructured grids for inhomogeneous, anisotropic media. Part I: derivation of the methods. *SIAM J. Sci. Comput.* 19 (5) (1998), 1700–1716. (Cited on page 105). http://dx.doi.org/10.1137/s1064827595293582.

[96] I. Aavatsmark, T. Barkve, O. Bøe, and T. Mannseth. Discretization on unstructured grids for inhomogeneous, anisotropic media. Part II: Discussion and numerical results. *SIAM J. Sci. Comput.* 19 (5) (1998), 1717–1736. (Cited on page 105). http://dx.doi.org/10.1137/s1064827595293594.

[97] J. Breil and P.-H. Maire. A cell-centered diffusion scheme on two-dimensional unstructured meshes. *J. Comput. Phys.* 224 (2007), 785–823. (Cited on page 105). http://dx.doi.org/10.1016/j.jcp.2006.10.025.

[98] K. Lipnikov, M. Shashkova, D. Svyatskiya, and Yu. Vassilevski. Monotone finite volume schemes for diffusion equations on unstructured triangular and shape-rmeshes. *J. Comput. Phys.* 227 (1) (2007), 492–512. (Cited on page 105). http://dx.doi.org/10.1016/j.jcp.2007.08.008.

[99] Z. Sheng, J. Yue, and G. Yuan. Monotone Finite Volume schemes of nonequilibrium radiation diffusion equations on distorted meshes. *SIAM J. Sci. Comput.* 31 (4) (2009), 2915–2934. (Cited on page 105). http://dx.doi.org/10.1137/080721558.

[100] J. Droniou and C. Le Potier. Construction and convergence study of schemes preserving the elliptic local maximum principle. *SIAM J. Sci. Comput.* 49 (2) (2011), 459–490. (Cited on page 105). http://dx.doi.org/10.1137/090770849.

[101] Z. Sheng and G. Yuan. An improved monotone Finite Volume scheme for diffusion equation on polygonal meshes. *J. Comput. Phys.* 231 (9) (2012), 3739–3754. (Cited on page 105). http://dx.doi.org/10.1016/j.jcp.2012.01.015.

[102] P. Jacq, P.-H. Maire, and R. Abgrall. A nominally second-order cell-centered Finite Volume scheme for simulating three-dimensional anisotropic diffusion equations on unstructured grids. *Commun. Comput. Phys.* 16 (04) (2014), 841–891. (Cited on page 105). http://dx.doi.org/10.4208/cicp.310513.170314a.

[103] F. Blachère and R. Turpault. An admissibility and asymptotic-preserving scheme for systems of conservation laws with source term on 2D unstructured meshes. *J. Comput. Phys.* 315 (2016), 98–123. (Cited on page 105). http://dx.doi.org/10.1016/j.jcp.2016.03.045.

[104] F. Blachère and R. Turpault. An admissibility and asymptotic preserving scheme for systems of conservation laws with source term on 2D unstructured meshes with high-order MOOD reconstruction. *Comput. Methods in Appl. Mech. Eng.* 317 (2017), 836–867. (Cited on page 105). http://dx.doi.org/10.1016/j.cma.2017.01.012.

[105] D. Pérez-Grande, O. Gonzalez-Martinez, P. Fajardo, and E. Ahedo. Analysis of the numerical diffusion in anisotropic mediums: benchmarks for magnetic field aligned meshes in space propulsion simulations. *Appl. Sci.* 6 (11) (2016), 354. (Cited on page 107). http://dx.doi.org/10.3390/app6110354.

[106] R. Eymard, T. Gallouët, and R. Herbin. A cell-centred Finite Volume approximation for anisotropic diffusion operators on unstructured meshes in any space dimension. *IMA J. Num. Anal.* 26 (2) (2006), 326–353. (Cited on page 107). http://dx.doi.org/10.1093/imanum/dri036.

[107] G. Puigt, V. Auffray, and J.-D. Müeller. Discretisation of diffusive fluxes on hybrid grids. *J. Comput. Phys.* 229 (5) (2010), 1425–1447. (Cited on page 107). http://dx.doi.org/10.1016/j.jcp.2009.10.037.

[108] L. Flandrin. Cell-centered methods for Euler and Navier-Stokes computations on unstructured meshes. PhD thesis. Université de Bordeaux I, (1995). (Cited on page 107).

[109] L. Flandrin, P. Charrier, and B. Dubroca. A robust Finite Volume method for computations of two-dimensional unstructured hybrid meshes. Computational Fluid Dynamics'94. Edited by S. Wagner, E.H. Hirschel, J. Périaux, and R. Piva. (1994), 301–308. (Cited on page 107).

[110] F. Dubois and O. Michaux. Numerical solution of the Navier-Stokes equations for hypersonic ramps with the NS3GR solver. Hypersonic Flows for Reentry Problems. Edited by R. Abgrall, J.-A. Désidéri, R. Glowinski, M. Mallet, and J. Périaux. Antibes, France. Springer-Verlag (1991), 258–269. (Cited on page 108). https://www.math.u-psud.fr/~fdubois/travaux/hyperbolique/antibes91.pdf.

[111] F. Dubois. Lemmes Finis pour la dynamique des gaz. (2012). (Cited on page 108). https://hal.archives-ouvertes.fr/hal-00733937/document.

[112] A. Dervieux. Steady Euler simulations using unstructured meshes. 16th VKI Lecture Series on Computational Fluid Dynamics 1985-04. Von Kármán Institute Rhode-St.-Genèse, Belgium. (1985). (Cited on page 110) http://dx.doi.org/10.1142/9789814415590_0002.

[113] P. Rostand and B. Stoufflet. Finite Volume Galerkin methods for viscous gas dynamics. *Tech. report*. (RR-863). INRIA. (1988). (Cited on page 110). https://hal.inria.fr/inria-00075691.

[114] T. Barth and P. Frederickson. Higher order solution of the Euler equations on unstructured grids using quadratic reconstruction. 28th Aerospace Sciences Meeting. (AIAA 1990-0013). (1990). (Cited on page 115) http://dx.doi.org/10.2514/6.1990-13.

[115] A. Nejat and C. Ollivier-Gooch. A high-order accurate unstructured Finite Volume Newton-Krylov algorithm for inviscid compressible flows. *J. Comput. Phys.* 227 (4) (2008), 2582–2609. (Cited on page 115). http://dx.doi.org/10.1016/j.jcp.2007.11.011.

[116] C. Ollivier-Gooch and M. Van Altena. A high-order-accurate unstructured mesh Finite Volume scheme for the advection-diffusion equation. *J. Comput. Phys.* 181 (2) (2002), 729–752. (Cited on page 115). http://dx.doi.org/10.1006/jcph.2002.7159.

[117] M. Delanaye and Y. Liu. Quadratic reconstruction finite volume schemes on 3D arbitrary unstructured polyhedral grids. 14th Computational Fluid Dynamics Conference. (AIAA-99-3259). Norfolk, Virginia, USA. (1999). (Cited on page 115) http://dx.doi.org/10.2514/6.1999-3259.

[118] A. Jalali and C. Ollivier-Gooch. Higher-order unstructured finite volume RANS solution of turbulent compressible flows. *Comput. Fluids* 143 (2017), 32–47. (Cited on page 115). http://dx.doi.org/10.1016/j.compfluid.2016.11.004.

[119] F. Haider, P. Brenner, B. Courbet, and J.-P. Croisille. Efficient implementation of high order reconstruction in Finite Volume methods. Finite Volumes for Complex Applications VI Problems & Perspectives. Prague, Czech Republic. Springer Berlin Heidelberg (2011), 553–560. (Cited on page 115) http://dx.doi.org/10.1007/978-3-642-20671-9_58.

[120] F. Haider, P. Brenner, B. Courbet, and J.-P. Croisille. Parallel implementation of k-exact Finite Volume reconstruction on unstructured grids. High order nonlinear numerical schemes for evolutionary PDEs. Bordeaux, France. Springer International Publishing (2014), 59–75. (Cited on page 115) http://dx.doi.org/10.1007/978-3-319-05455-1_4.

[121] D. Caraeni and D. Hill. Efficient 3rd-order Finite Volume discretization using iterative quadratic data reconstruction on unstructured grids. 47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition. (AIAA-2009-1332). Orlando, Florida, USA. (2009). (Cited on page 115) http://dx.doi.org/10.2514/6.2009-1332.

[122] D. Caraeni and D.C. Hill. Unstructured-grid third-order Finite Volume discretization using a multistep quadratic data-reconstruction method. *AIAA J.* 48 (4) (2010), 808–817. (Cited on page 115). http://dx.doi.org/10.2514/1.45051.

[123] H.Q. Yang, Z.J. Chen, A. Przekwas, and J. Dudley. A high-order CFD method using successive differentiation. *J. Comput. Phys.* 281 (2015), 690–707. (Cited on page 115). http://dx.doi.org/10.1016/j.jcp.2014.10.046.

[124] A. Harten, B. Engquist, S. Osher, and S.R. Chakravarthy. Uniformly high order accurate essentially non-oscillatory schemes, III. *J. Comput. Phys.* 71 (2) (1987), 231–303. (Cited on page 116). http://dx.doi.org/10.1016/0021-9991(87)90031-3.

[125] R. Abgrall. On Essentially Non-Oscillatory schemes on unstructured meshes: analysis and implementation. *J. Comput. Phys.* 114 (1) (1994), 45–58. (Cited on page 116). http://dx.doi.org/10.1006/jcph.1994.1148.

[126] C.F. Ollivier-Gooch. Quasi-ENO schemes for unstructured meshes based on unlimited data-dependent Least-Squares reconstruction. *J. Comput. Phys.* 133 (1) (1997), 6–17. (Cited on page 116). http://dx.doi.org/10.1006/jcph.1996.5584.

[127] T.J. Barth and H. Deconinck, editors. High-order methods for computational physics. Springer Berlin Heidelberg (1999). (Cited on page 116). http://dx.doi.org/10.1007/978-3-662-03882-6.

[128] X.-D. Liu, S. Osher, and T. Chan. Weighted Essentially Non-Oscillatory schemes. *J. Comput. Phys.* 115 (1) (1994), 200–212. (Cited on page 116). http://dx.doi.org/10.1006/jcph.1994.1187.

[129] O. Friedrich. Weighted Essentially Non-Oscillatory schemes for the interpolation of mean values on unstructured grids. *J. Comput. Phys.* 144 (1) (1998), 194–212. (Cited on page 116). http://dx.doi.org/10.1006/jcph.1998.5988.

[130] C. Hu and C.-W. Shu. Weighted Essentially Non-Oscillatory schemes on triangular meshes. *J. Comput. Phys.* 150 (1) (1999), 97–127. (Cited on page 116). http://dx.doi.org/10.1006/jcph.1998.6165.

[131] S. Clain, S. Diot, and R. Loubère. A high-order Finite Volume method for systems of conservation laws-Multi-dimensional Optimal Order Detection (MOOD). *J. Comput. Phys.* 230 (10) (2011), 4028–4050. (Cited on page 116). http://dx.doi.org/10.1016/j.jcp.2011.02.026.

[132] S. Diot, S. Clain, and R. Loubère. Improved detection criteria for the Multi-dimensional Optimal Order Detection (MOOD) on unstructured meshes with very high-order polynomials. *Comput. Fluids* 64 (2012), 43–63. (Cited on page 116). http://dx.doi.org/10.1016/j.compfluid.2012.05.004.

[133] S. Diot, R. Loubère, and S. Clain. The Multidimensional Optimal Order Detection method in the three-dimensional case: very high-order finite volume method for hyperbolic systems. *Int. J. Numer. Meth. Fluids* 73 (4) (2013), 362–392. (Cited on page 116). http://dx.doi.org/10.1002/fld.3804.

[134] J. Vanharen, G. Puigt, X. Vasseur, J.-F. Boussuge, and P. Sagaut. Revisiting the spectral analysis for high-order spectral discontinuous methods. *J. Comput. Phys.* 337 (2017), 379–402. (Cited on page 129). http://dx.doi.org/10.1016/j.jcp.2017.02.043.

[135] J. Vanharen, G. Puigt, X. Vasseur, J.-F. Boussuge, and P. Sagaut. Revisiting the spectral analysis for high-order spectral discontinuous methods. TILDA - Symposium & Workshop on Industrial LES & DNS. Toulouse, France. (2016). (Cited on page 129). https://www.researchgate.net/publication/310606899_Revisiting_the_spectral_analysis_for_high-order_spectral_discontinuous_methods.

[136] J. Vanharen, G. Puigt, X. Vasseur, J.-F. Boussuge, and P. Sagaut. Comparing the resolution power of standard finite difference and spectral difference schemes. TILDA - Symposium & Workshop on Industrial LES & DNS. Toulouse, France. (2016). (Cited on page 129). https://www.researchgate.net/publication/310606902_Comparing_the_resolution_power_of_standard_finite_difference_and_spectral_difference_schemes.

[137] M.M. Rai and P. Moin. Direct simulations of turbulent flow using Finite Difference schemes. *J. Comput. Phys.* 96 (1) (1991), 15–53. (Cited on page 129). http://dx.doi.org/10.1016/0021-9991(91)90264-L.

[138] W.H. Reed and T.R. Hill. Triangular mesh methods for the neutron transport equation. *Tech. report*. (LU-UR-73-279). Los Alamos National Laboratory. New Mexico, USA. (1973). (Cited on page 130). https://www.osti.gov/scitech/servlets/purl/4491151.

[139] B. Cockburn and C.W. Shu. TVB Runge-Kutta local projection Discontinuous Galerkin finite method for conservation laws II: general framework. *Math. Comp.* 52 (186) (1989), 411–435. (Cited on page 130). http://dx.doi.org/10.1090/S0025-5718-1989-0983311-4.

[140] B. Cockburn, S.-Y. Lin, and C.-W. Shu. TVB Runge-Kutta local projection Discontinuous Galerkin Finite Element method for conservation laws III: one-dimensional systems. *J. Comput. Phys.* 84 (1) (1989), 90–113. (Cited on page 130). http://dx.doi.org/10.1016/0021-9991(89)90183-6.

[141] B. Cockburn, S. Hou, and C.-W. Shu. The Runge-Kutta local projection Discontinuous Galerkin Finite Element method for conservation laws IV: the multidimensional case. *Math. Comp.* 54 (1990), 545–581. (Cited on page 130). http://dx.doi.org/10.1090/S0025-5718-1990-1010597-0.

[142] J.S. Hesthaven and T. Warburton. Nodal Discontinuous Galerkin Methods: algorithms, analysis and applications. Edited by J.E. Marsden, L. Sirovich, and S.S. Antman. Springer Publishing Company, Incorporated (2008). (Cited on page 130). http://dx.doi.org/10.1007/978-0-387-72067-8.

[143] D.A. Kopriva and J.H. Kolias. A conservative staggered-grid Chebyshev multidomain method for compressible flows. *J. Comput. Phys.* 125 (1) (1996), 244–261. (Cited on pages 130, 140). http://dx.doi.org/10.1006/jcph.1996.0091.

[144] Y. Liu, M. Vinokur, and Z.J. Wang. Spectral Difference method for unstructured grids I: basic formulation. *J. Comput. Phys.* 216 (2) (2006), 780–801. (Cited on page 130). http://dx.doi.org/10.1016/j.jcp.2006.01.024.

[145] Z.J. Wang, Y. Liu, G. May, and A. Jameson. Spectral Difference Method for Unstructured grids II: extension to the Euler equations. *J. Sci. Comput.* 32 (1) (2007), 45–71. (Cited on page 130). http://dx.doi.org/10.1007/s10915-006-9113-9.

[146] C. Liang, A. Jameson, and Z.J. Wang. Spectral difference method for compressible flow on unstructured grids with mixed elements. *J. Comput. Phys.* 228 (8) (2009), 2847–2858. (Cited on page 130). http://dx.doi.org/10.1016/j.jcp.2008.12.038.

[147] Z.J. Wang. Spectral (Finite) Volume method for conservation laws on unstructured grids: basic formulation. *J. Comput. Phys.* 178 (1) (2002), 210–251. (Cited on page 130). http://dx.doi.org/10.1006/jcph.2002.7041.

[148] Z.J. Wang and Y. Liu. Spectral (Finite) Volume method for conservation laws on unstructured grids II: extension to two-dimensional scalar equation. *J. Comput. Phys.* 179 (2) (2002), 665–697. (Cited on page 130). http://dx.doi.org/10.1006/jcph.2002.7082.

[149] Z.J. Wang and Y. Liu. Spectral (Finite) Volume method for conservation laws on unstructured grids III: one dimensional system and partition optimization. *J. Sci. Comput.* 20 (2004), 137–157. (Cited on page 130). http://dx.doi.org/10.1023/A:1025896119548.

[150] Z.J. Wang, L. Zang, and Y. Liu. Spectral (Finite) Volume method for conservation laws on unstructured grids IV: Extension to two-dimensional systems. 194 (2) (2004), 716–741. (Cited on page 130). http://dx.doi.org/10.1016/j.jcp.2003.09.012.

[151] Y. Liu, M. Vinokur, and Z.J. Wang. Spectral (Finite) Volume method for conservation laws on unstructured grids V: extension to three-dimensional systems. *J. Comput. Phys.* 212 (2) (2006), 454–472. (Cited on page 130). http://dx.doi.org/10.1016/j.jcp.2005.06.024.

[152] H.T. Huynh. A Flux Reconstruction approach to high-order schemes including Discontinuous Galerkin Methods. 18th AIAA Computational Fluid Dynamics Conference. (AIAA-2007-4079). Miami, Florida, USA. (2007). (Cited on pages 130, 140) http://dx.doi.org/10.2514/6.2007-4079.

[153] Z.J. Wang and H. Gao. A unifying lifting collocation penalty formulation including the Discontinuous Galerkin, spectral volume/difference methods for conservation laws on mixed grids. *J. Comput. Phys.* 228 (21) (2009), 8161–8186. (Cited on page 130). http://dx.doi.org/10.1016/j.jcp.2009.07.036.

[154] P.E. Vincent, P. Castonguay, and A. Jameson. A new class of high-order energy stable Flux Reconstruction schemes. *J. Sci. Comput.* 47 (1) (2011), 50–72. (Cited on pages 130, 131, 143, 148). http://dx.doi.org/10.1007/s10915-010-9420-z.

[155] P. Castonguay, P.E. Vincent, and A. Jameson. A new class of high-order energy stable Flux Reconstruction schemes for triangular elements. *J. Sci. Comput.* 51 (2012), 224–256. (Cited on page 130). http://dx.doi.org/10.1007/s10915-011-9505-3.

[156] H.T. Huynh, Z.J. Wang, and P.E. Vincent. High-order methods for computational fluid dynamics: a brief review of compact differential formulations on unstructured grids. *Comput. Fluids* 98 (2014), 209–220. (Cited on page 130). http://dx.doi.org/10.1016/j.compfluid.2013.12.007.

[157] Y. Allaneau and A. Jameson. Connections between the filtered Discontinuous Galerkin method and the Flux Reconstruction approach to high-order discretizations. *Comput. Methods in Appl. Mech. Eng.* 200 (2011), 3626–3636. (Cited on page 130). http://dx.doi.org/10.1016/j.cma.2011.08.019.

[158] F.Q. Hu, M.Y. Hussaini, and P. Rasetarinera. An analysis of the Discontinuous Galerkin Method for wave propagation problems. *J. Comput. Phys.* 151 (1999), 921–946. (Cited on pages 131, 143, 148). http://dx.doi.org/10.1006/jcph.1999.6227.

[159] M. Zhang and C. W. Shu. An analysis of three different formulations of the Discontinuous Galerkin method diffusion equations. *Math. Methods Appl. Sci.* 13 (3) (2003), 395–414. (Cited on pages 131, 143, 148). http://dx.doi.org/10.1142/S0218202503002568.

[160] R. Kannan and Z.J. Wang. LGD2: A variant of the LDG flux formulation for the Spectral Volume Method. *J. Sci. Comput.* 46 (2011), 314–328. (Cited on pages 131, 143, 148). http://dx.doi.org/10.1007/s10915-010-9391-0.

[161] K. Van den Abeele, C. Lacor, and Z.J. Wang. On the stability and accuracy of the spectral difference method. *J. Sci. Comput.* 37 (2008), 162–188. (Cited on pages 131, 140, 143, 148). http://dx.doi.org/10.1007/s10915-008-9201-0.

[162] P.E. Vincent, P. Castonguay, and A. Jameson. Insights from Von Neumann analysis of high-order Flux Reconstruction schemes. *J. Comput. Phys.* 230 (22) (2011), 8134–8154. (Cited on pages 131, 143, 148). http://dx.doi.org/10.1016/j.jcp.2011.07.013.

[163] K. Asthana and A. Jameson. High-order Flux Reconstruction schemes with minimal dispersion and dissipation. *J. Sci. Comput.* 62 (3) (2015), 913–944. (Cited on pages 131, 143, 148). http://dx.doi.org/10.1007/s10915-014-9882-5.

[164] D.V. Gaitonde and M.R. Visbal. Further development of a Navier-Stokes solution procedure based on higher-order formulas. 37th Aerospace Sciences Meeting and Exhibit. (AIAA-1999-557). Reno, Nevada, USA. (1999). (Cited on page 132) http://dx.doi.org/10.2514/6.1999-557.

[165] K.M. Aikens, N.S. Dhamankar, C.S. Martha, Y. Situ, and G.A. Blaisdell. Equilibrium wall model for Large Eddy Simulations of jets for aeroacoustics. AIAA SciTech, 52nd Aerospace Sciences Meeting. (AIAA 2014-0180). National Harbor, Maryland, USA. (2014). (Cited on page 132) http://dx.doi.org/10.2514/6.2014-0180.

[166] S. Le Bras, H. Deniau, C. Bogey, and G. Daviller. Development of compressible Large-Eddy Simulations combining high-order schemes and wall modeling. 21st AIAA/CEAS Aeroacoustics Conference, AIAA Aviation Forum. (AIAA 2015-3135). Dallas, Texas, USA. (2015). (Cited on page 132) http://dx.doi.org/10.2514/6.2015-3135.

[167] C. Bogey, N. de Cacqueray, and C. Bailly. A shock-capturing methodology based on adaptative spatial filtering for high-order non-linear computations. *J. Comput. Phys.* 228 (5) (2009), 1447–1465. (Cited on page 132). http://dx.doi.org/10.1016/j.jcp.2008.10.042.

[168] C.E. Shannon. Communication in the presence of noise. Proceedings of the Institute of Radio Engineers. (1). Volume 37 (1949), 10–21. (Cited on page 133) http://dx.doi.org/10.1109/JRPROC.1949.232969.

[169] J. Ramboer, T. Broeckhoven, S. Smirnov, and C. Lacor. Optimization of time integration schemes coupled to spatial discretization for use in CAA applications. *J. Comput. Phys.* 213 (2) (2006), 777–802. (Cited on page 137). http://dx.doi.org/10.1016/j.jcp.2005.08.033.

[170] T.K. Sengupta, A. Dipankar, and P. Sagaut. Error dynamics: beyond von Neumann analysis. *J. Comput. Phys.* 226 (2) (2007), 1211–1218. (Cited on page 137). http://dx.doi.org/10.1016/j.jcp.2007.06.001.

[171] T.K. Sengupta. High accuracy computing methods: fluid flows and wave phenomena. Cambridge University Press (2013). (Cited on page 137). http://www.cambridge.org/fr/academic/subjects/mathematics/computational-science/high-accuracy-computing-methods-fluid-flows-and-wave-phenomena?format=HB&isbn=9781107023635.

[172] A. Jameson. A proof of the stability of the Spectral Difference Method for all orders of accuracy. *J. Sci. Comput.* 45 (1) (2010), 348–358. (Cited on page 140). http://dx.doi.org/10.1007/s10915-009-9339-4.

[173] O. Axelsson. Iterative solution methods. Cambridge University Press (1994). (Cited on page 145). http://dx.doi.org/10.1017/CBO9780511624100.

[174] G.H. Golub and C.F. Van Loan. Matrix computations (3rd Ed.) Johns Hopkins University Press (1996). (Cited on page 145). https://jhupbooks.press.jhu.edu/content/matrix-computations.

[175] C. Bogey, C. Bailly, and D. Juvé. Computation of flow noise using source terms in linearized Euler's equations. *AIAA J.* 40 (2) (2002), 235–243. (Cited on page 161). http://dx.doi.org/10.2514/2.1665.

[176] A. Cassagne, G. Puigt, and J.-F. Boussuge. High-order method for a new generation of Large Eddy Simulation solver. *Tech. report*. Partnership for Advanced Computing in Europe (PRACE). (2015). (Cited on page 161). http://www.cerfacs.fr/~cfdbib/repository/TR_CFD_15_9.pdf.

[177] J. Vanharen and G. Puigt. Optimized Runge-Kutta time integration for Spectral Difference Method. *Under preparation for Journal of Computational Physics* (2017). (Cited on page 167).

[178] J. Vanharen, G. Puigt, J.-F. Boussuge, and A. Balan. Optimized Runge-Kutta time integration for Spectral Difference Method. 23rd AIAA Computational Fluid Dynamics Conference, 2017 AIAA Aviation and Aeronautics Forum and Exposition. Denver, Colorado, USA. (2017). (Cited on page 167).

[179] V. Joncquières, G. Puigt, and J. Vanharen. About the solution initialization for the Spectral Difference Method. *Under preparation for Journal of Computational Physics* (2017). (Cited on page 167).

[180] A. Balan, G. May, and J. Schöberl. A stable high-order Spectral Difference method for hyperbolic conservation laws on triangular elements. *J. Comput. Phys.* 231 (5) (2012), 2359–2375. (Cited on page 167). http://dx.doi.org/http://dx.doi.org/10.1016/j.jcp.2011.11.041.

[181] M. Li, Z. Qiu, C. Liang, and M.A. Sprague. A new high-order Spectral Difference Method for simulating viscous flows on unstructured grids with mixed elements. 55th AIAA Aerospace Sciences Meeting. (AIAA-2017-0520). Grapevine, Texas, USA. (2017). (Cited on page 167) http://dx.doi.org/10.2514/6.2017-0520.

[182] B. Zhang and C. Liang. A simple, efficient, and high-order accurate curved sliding-mesh interface approach to Spectral Difference method on coupled rotating and stationary domains. *J. Comput. Phys.* 295 (2015), 147–160. (Cited on page 167). http://dx.doi.org/10.1016/j.jcp.2015.04.006.

[183] C. Geuzaine and J.-F. Remacle. Gmsh: A 3-D Finite Element mesh generator with built-in pre- and post-processing facilities. *Int. J. Numer. Meth. Eng.* 79 (11) (2009), 1309–1331. (Cited on page 171). http://dx.doi.org/10.1002/nme.2579.

[184] C. Rumsey, B. Wedan, T. Hauser, and M. Poinot. Recent Updates to the CFD General Notation System (CGNS). 50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition. (AIAA-2012-1264). Nashville, Tennessee, USA. (2012). (Cited on page 171) http://dx.doi.org/10.2514/6.2012-1264.

[185] G. Karypis and V. Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM J. Sci. Comput.* 20 (1) (1998), 359–392. (Cited on page 171). http://dx.doi.org/10.1137/s1064827595287997.

[186] Antares Development Team. Antares Documentation Release 1.9.0. (2012 - 2017). (Cited on page 172). http://cerfacs.fr/antares/.

[187] K. Masatsuka. I do like CFD, Vol. 1. (2013). (Cited on page 177). http://www.cfdbooks.com/download_idolikecfd_free_2nd_edition.html.

[188] B. Diskin and J.L. James. Notes on accuracy of Finite-Volume discretization schemes on irregular grids. *Tech. report*. (NF1676L-8756). NASA Langley Research Center. Hampton, Virgina, USA. (2011). (Cited on page 185). https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20110011294.pdf.

[189] M.B. Giles. Accuracy of node-based solutions on irregular meshes. Edited by D.L. Dwoyer, M.Y. Hussaini, and R.G. Voigt. Springer Berlin Heidelberg (1989), 273–277. (Cited on page 185). http://dx.doi.org/10.1007/3-540-51048-6_40.

[190] D. Schwamborn, T. Gerhold, and V. Hannemann. On the validation of the DLR-TAU code. Notes on Numerical Fluid Mechanics (NNFM). Springer Nature (1999), 426–433. (Cited on page 187) http://dx.doi.org/10.1007/978-3-663-10901-3_55.

[191] T. Gerhold. Overview of the hybrid RANS code TAU. MEGAFLOW - Numerical flow simulation for aircraft design. Springer Nature (2005), 81–92. (Cited on page 187) http://dx.doi.org/10.1007/3-540-32382-1_5.

[192] C.-C. Rossow, N. Kroll, and D. Schwamborn. The MEGAFLOW project - Numerical flow simulation for aircraft. Progress in Industrial Mathematics at ECMI 2004. Springer Nature (2006), 3–33. (Cited on page 187) http://dx.doi.org/10.1007/3-540-28073-1_1.



Abstract: This work deals with high-order numerical methods for unsteady flows around complex geometries. In order to cope with the low-order industrial Finite Volume Method, the proposed technique consists in computing on structured and unstructured zones with their associated schemes: this is called a hybrid approach. Structured and unstructured meshes are then coupled by a nonconforming grid interface. The latter is analyzed in details with special focus on unsteady flows. It is shown that a dedicated treatment at the interface avoids the reflection of spurious waves. Moreover, this hybrid approach is validated on several academic test cases for both convective and diffusive fluxes. The extension of this hybrid approach to high-order schemes is limited by the efficiency of unstructured high-order schemes in terms of computational time. This is why a new approach is explored: The Spectral Difference Method. A new framework is especially developed to perform the spectral analysis of Spectral Discontinuous Methods. The Spectral Difference Method seems to be a viable alternative in terms of computational time and number of points per wavelength needed for a given application to capture the flow physics.

Keywords: Finite Volume, Nonconforming Grid Interface, Hybrid Approach, High-Order Unstructured Schemes, Spurious Waves, Spectral Difference, Spectral Analysis.

Résumé : Dans ce travail, on s'intéresse aux méthodes numériques d'ordre élevé pour des écoulements instationnaires autour de géométries complexes. On commence par analyser l'approche hybride pour la méthode industrielle des Volumes Finis à l'ordre faible. Cela consiste à calculer en même temps sur des maillages structurés et non structurés avec des schémas numériques dédiés. Les maillages structurés et non structurés sont ensuite couplés par un raccord non conforme. Ce dernier est analysé en détails avec une attention particulière pour des écoulements instationnaires. On montre qu'un traitement dédié à l'interface empêche la réflexion d'ondes parasites. De plus, l'approche hybride est validée sur plusieurs cas académiques à la fois pour les flux convectifs et pour les flux diffusifs. L'extension de cette approche hybride à l'ordre élevé est limitée par l'efficacité des schémas non structurés d'ordre élevé en terme de temps de calcul. C'est pourquoi une nouvelle approche est explorée : la méthode des différences spectrales. Un nouveau cadre est spécialement développé pour réaliser l'analyse spectrale des méthodes spectrales discontinues. La méthode des différences spectrales semble être une alternative viable en terme de temps de calcul et de nombre de points par longueur d'onde nécessaires à une application donnée pour capturer la physique de l'écoulement.

Mots-clefs : Volume Finis, Raccords Non Conformes, Approche Hybride, Schémas Non Structurés d'Ordre Elevé, Ondes Parasites, Différences Spectrales, Analyse Spectrale.