Shock stabilisation techniques for large eddy simulations using the high order discontinuous Galerkin method

Auteur : Boualem, Zakaria
Promoteur(s) : Hillewaert, Koen
Faculté : Faculté des Sciences appliquées
Diplôme : Master en ingénieur civil physicien, à finalité approfondie
Année académique : 2023-2024
URI/URL : http://hdl.handle.net/2268.2/19571

Shock stabilization techniques for large eddy simulations using the high order discontinuous Galerkin method

Master's thesis submitted in fulfillment of the requirements for the degree of Master in Physics Engineering

Thesis written by ZAKARIA BOUALEM

Advisor
PR. K HILLEWAERT (ULIÈGE)
Co-advisor
A. BILOCQ (ULIÈGE)

Jury
PR. V. TERRAPON (ULIÈGE)
DR. P. SCHROOYEN (VKI)

University of Liège - Faculty of Applied Sciences

Academic year 2023 - 2024

Remerciements

Je souhaite avant tout exprimer ma sincère reconnaissance à mon directeur de thèse, Prof. Koen H., ainsi qu'à son équipe composée d'Amaury B., Nayan L. et Maxime B., pour leur soutien précieux, leurs conseils judicieux et leur patience durant ce parcours de recherche. Leur sérieux scientifique et leur engagement envers l'excellence ont été une inspiration constante tant sur le plan professionnel que personnel.

Mes sincères remerciements vont également aux commitées de lecture pour le temps qu'ils consacreront à la lecture de ce mémoire.

Mes remerciements vont aussi à ma famille, mon père Kamel B., ma mère Fabienne D., mon frère Ismaïl B., ma petite soeur Leïla B. et ma grand-mère Suzy P. pour leur soutien inébranlable. Un merci particulier à mon père Kamel, qui, bien qu'il ne comprenne pas totalement le sujet, a continué à me conseiller avec persévérance. Je tiens également à remercier tout spécialement mon petit frère, qui, bien qu'il n'ait pas contribué directement, est devenu au fil des années une immense source d'inspiration pour moi, en surmontant d'innombrables obstacles pour arriver là où il est aujourd'hui.

Je tiens également à remercier mes vieux amis, Dorian B., Simon B., Timothy B., Maxime B., Cyril D., Guillaume P., Laura M., Aurélien R., Delio S. qui m'ont toujours soutenu et encouragé tout au long de mon parcours, apportant à la fois amitié et précieux conseils. Une mention spéciale pour Guillaume P., pour sa grande patience et sa compréhension face à toutes les situations délicates que je lui ai imposées.

Je voudrais aussi exprimer ma gratitude envers les ingénieurs civil physicien, Julia B., Rayan E., Sarah H., Philippe T. et Jules Z. ainsi que mon groupe de projet intégré Tom S. et William B. qui m'ont accueilli et soutenu dans cette nouvelle phase de ma vie. Leur fraîche perspective, leur enthousiasme et leur amitié ont été une source inestimable de réconfort et d'inspiration, enrichissant mon parcours de manière significative.

Une attention toute particulière est adressée à Lynda B. et Alyzée A.

Enfin, j'adresse mes remerciements les plus sincères à ma meilleure amie, Coralie Meurisse, et à mon frère de projet, Damien Sansen. Leur soutien, leur dévouement et leur collaboration ont été essentiels à la réalisation de cette thèse. Sans leur présence constante, leur encouragement et leur expertise, ce travail n'aurait pas atteint son plein potentiel. Leur contribution a été non seulement professionnelle, mais aussi personnelle, marquant profondément tant ma carrière que ma vie.

Abstract

Numerical simulation of fluid flows plays a crucial role in many areas of science and engineering, including aerodynamics and meteorology. One of the greatest challenges in these simulations is accurately modeling shocks or shock waves, which are sudden discontinuities in fluid properties such as pressure, temperature, and velocity. Shocks typically occur in high-speed flows, such as those involving supersonic or hypersonic speeds, and incorrect handling can lead to non-physical results and numerical instability in the simulation.

Conventional numerical methods, like those based on finite differences, finite volumes, or finite elements, may struggle to capture the abrupt nature of shocks due to their limited resolution and tendency for numerical diffusion. This often leads to poorly resolved physical gradients and inaccurate flow representation, resulting in loss of precision and errors in the final results. To overcome these challenges, various techniques have been developed to better capture shocks in numerical fluid flows.

Among these techniques is one known as artificial viscosity. This method addresses the weaknesses of traditional numerical models by artificially increasing the thickness of shocks, facilitating more accurate and stable resolution in finite volume or finite element methods. While effective for stabilizing simulations, it must be used judiciously to avoid excessive dissipation of energy, which could otherwise lead to the loss of critical details in the simulation.

A new approach to artificial viscosity, aimed at better aligning with real physical principles, has been integrated into ForDGe, the Cartesian solver based on a discontinuous Galerkin formulation developed by the DoT group.

The study aims to compare the performance of this artificial viscosity strategy with a Laplacian method already in place in the software, through the analysis of two specific case studies: Kelvin-Helmholtz instabilities and the inviscid interaction between a strong vortex and a shock wave.

Résumé

La simulation numérique des écoulements fluides joue un rôle crucial dans de nombreux domaines de la science et de l'ingénierie, notamment en aérodynamique et en météorologie. Une des plus grandes difficultés rencontrées dans ces simulations est la modélisation précise des chocs, ou ondes de choc, qui sont des discontinuités soudaines dans les propriétés du fluide telles que la pression, la température et la vitesse. Les chocs surviennent généralement dans des écoulements à haute vitesse, comme ceux impliquant des vitesses supersoniques ou hypersoniques, et leur traitement incorrect peut conduire à des résultats non physiques et à l'instabilité numérique de la simulation.

Les méthodes numériques classiques, telles que celles basées sur les différences finies, les volumes finis ou les éléments finis, peuvent avoir du mal à capturer la nature abrupte des chocs en raison de leur résolution limitée et de la tendance à la diffusion numérique. Cela mène souvent à des gradients physiques mal résolus et à une représentation inexacte du flux, entraînant une perte de précision et des erreurs dans les résultats finaux. Pour surmonter ces défis, diverses techniques ont été développées pour mieux capturer les chocs dans les écoulements de fluides numériques.

Parmi ces techniques, l'une est connue sous le nom de viscosité artificielle. Cette méthode compense les faiblesses des modèles numériques traditionnels en augmentant artificiellement l'épaisseur des chocs, ce qui facilite une résolution plus exacte et stable dans les méthodes de volumes finis ou d'éléments finis. Bien que cette approche soit efficace pour stabiliser les simulations, elle doit être utilisée judicieusement afin d'éviter une dissipation excessive de l'énergie, qui pourrait autrement entraîner la perte de détails essentiels dans la simulation.

Une nouvelle approche de viscosité artificielle avec pour objectif de mieux correspondre aux principes physiques réels, a été intégrée dans ForDGe, le solveur cartésien basé sur une formulation Galerkin discontinue conçu par le groupe DoT.

L'étude propose de comparer les performances de cette stratégie de viscosité artificielle avec une méthode laplacienne déjà en place dans le logiciel, et ce, à travers l'analyse de deux cas d'étude spécifiques : les instabilités de Kelvin-Helmholtz et l'interaction non visqueuse entre un vortex fort et une onde de choc.

Contents

List of Figures

Nomenclature

Acronyms

CFD   Computational Fluid Dynamics

RANS   Reynnolds Averaged Navier-Stokes

LES   Large Eddy Simulation

DNS   Direct Numerical Simulation

DG   Discontinuous Galerkin

ENO   Essentially Non-Oscillatory

WENO   Weighted Essentially Non-Oscillatory

DA   Direct Approach

CBA   Continuous Bernstein Approach

NCBA   Non Continuous Bernstein Approach

Mathematical conventions

$\cdot$ Scalar product

$\times$ Cross product

$\vec{n}$ Normal unit vector

$\vec{q}$ Vector quantity q

$\vec{\vec{q}}$ Tensor quantity q

Physical quantities

ρ	Density	$[\mathrm{kg/m^3}]$
$\vec{v}$	Fluid velocity	$[\mathrm{m/s}]$

E	Internal energy	$[\mathrm{J}]$
p	Pressure	$[\mathrm{Pa}]$
T	Temperature	$[\mathrm{K}]$
$\vec{f}$	Heat flux	$[\mathrm{W/m^2}]$
μ	Shear viscosity	$[\mathrm{Pa\ s}]$
β	Bulk viscosity	$[\mathrm{Pa\ s}]$
κ	Thermal conductivity	$[\mathrm{W/m/K}]$
c^*	Critical speed of sound	$[\mathrm{m/s}]$
γ	Specific heat ratio	$[-]$
R	Ideal gas constant	$[\mathrm{J/kg/K}]$
δ	Kronecker delta	$[-]$
$\vec{\vec{\tau}}$	Viscous stress tensor	$[\mathrm{Pa}]$
σ_f	Penalty factor	$[\mathrm{m/s^2}]$
C_v	Heat capacity at constant volume	$[\mathrm{J/kgK}]$
K	Kinetic energy	$[\mathrm{J}]$
Sch	Numerical schlieren	$[-]$
$\vec{\omega}$	Vorticity	$[\mathrm{1/s}]$
ξ	Enstrophy	$[\mathrm{1/s^2}]$
v_m	Background velocity	$[\mathrm{m/s}]$

Chapter 1

Introduction

1   Context

Computational Fluid Dynamics (CFD) is an essential research field in various engineering and applied science domains. Its objective is to develop and use numerical methods to solve fluid dynamics equations, enabling accurate and efficient simulation and study of fluid flows, combustion, heat transfer, and other fluid-related phenomena.

The equations governing fluid behavior exhibit nonlinearities due to forces, flows, and interactions among different parts of the system. These nonlinear characteristics make the analytical solution of these equations complex, if not impossible without simplifying assumptions, necessitating the use of numerical methods. In 1965, Harlow published one of the earliest papers in the field of CFD [1], describing a numerical method to solve the incompressible Navier-Stokes equations, which are fundamental for numerical simulation. Thanks to the work of numerous researchers like Harlow and advancements in digital computers, CFD has experienced rapid evolution, opening new perspectives in the design and optimization of complex systems.

The study of fluid flows demands appropriate approaches to capture and understand dynamic phenomena. In practice, three approaches are employed, each offering unique characteristics and different perspectives for fluid flow analysis.

The Reynolds Averaged Navier-Stokes (RANS) approach relies on the scale separation assumption, where turbulent effects are modeled using turbulence models. The Navier-Stokes equations are time-averaged and solved to obtain the overall behavior of the flow, resulting in a loss of information about small-scale fluid behavior.

The second approach is Direct Numerical Simulation (DNS), involving the direct solution of the Navier-Stokes equations without any turbulence modeling for all spatial scales of the fluid. This method provides rigorously accurate and detailed information about flow structures and fluctuations. However, this approach is extremely computationally expensive and is generally limited to low Reynolds number problems.

These two approaches are extremes in their resolutions, with one providing information only about mean flow at an acceptable computational cost, while the other yields extremely detailed flow information at a high cost. This is where the third approach, known as Large Eddy Simulations (LES), comes into play. It strikes a balance between the previous two approaches, directly resolving the large turbulent structures of the flow, while smaller scales are modeled using subgrid-scale models [2]. This approach provides more information about turbulent structures while limiting

computational cost, making it particularly suitable for studying high Reynolds number flows. LES is employed in various fields such as aerospace engineering [3–5].

Nowadays, most CFD codes are built upon the foundation of the Reynolds-Averaged Navier-Stokes (RANS) equations, employing numerical methods of orders 1 or 2, such as Finite Volume (FV), Finite Difference (FD), or Finite Element (FE) approaches [6]. While these methods produce satisfactory results, they encounter limitations when dealing with challenges that extend beyond their designated domains. Examples include unsteady propagating vortices or transonic and supersonic flows containing shocks. Despite displaying promising outcomes, these techniques exhibit excessive dissipation and fall short of achieving the level of precision required for viable solutions. Consequently, turning to higher-order methods (beyond order 2) is deemed more advantageous, as they demonstrate promising potential in handling such flows [7].

2   High order methods and Discontinuous Galerkin Method

The main advantage of using higher-order methods is enhanced accuracy. Indeed, for a fixed mesh size h and order p, the numerical error e of the method is given by

$$e \propto h^{p+1}.$$

Within the framework of a fourth-order scheme, if the mesh size is halved, the numerical error e should significantly decrease, by a factor of 16 on a coarser mesh. This illustration highlights the impact of increasing the order as opposed to simple mesh refinement. At first glance, it might be easy to assume that higher-order methods require more computation time than first-order methods. However, this observation holds true only in the context of steady-state solution calculation on an identical mesh. For the same predefined error, higher-order methods will attain this accuracy more rapidly. Furthermore, lower-order methods might require a finer mesh to achieve the same level of precision. It is important to note that if the numerical error is not low enough, higher-order methods might require more computation time [8].

Despite the significant advantages offered by higher-order methods, they are still not commonly used in industrial design processes due to several reasons. Firstly, higher-order methods are more intricate and less robust than their lower-order counterparts [9]. Secondly, their convergence towards a steady state is often slower due to reduced numerical dissipation, which can render their use less practical in contexts where rapid results are required [8]. Moreover, higher-order methods frequently demand the use of implicit time-stepping schemes, which implies increased complexity and greater memory requirements. As a result, their implementation can be more resource-intensive in terms of computational resources.

Higher-order methods hold tremendous promise for the future and offer essential advantages for pushing the current boundaries of industries. Over the past two decades, a multitude of higher-order methods have been developed and applied in various applications. In this work, the Discontinuous Galerkin Finite Element Method (DGFEM) will be addressed and elaborated upon in different sections. This method has remarkable features in terms of accuracy, adaptability to complex geometries and low dissipation to capture physical phenomena more effectively. Its promising potential places it at the forefront of ongoing research in the field of higher-order methods applied to numerical fluid simulation [10–14].

3 Discontinuities and shock capturing methods

One aspect of Computational Fluid Dynamics (CFD) involves studying and managing discontinuities (abrupt changes in fluid properties), presenting unique challenges for simulation and analysis.

Discontinuities

Discontinuities manifest in several ways, each reflecting distinct physical phenomena. Common types include contact discontinuities, separating two different fluids or regions of the same fluid with differing properties, crucial in fluid mixing studies. Shear layer discontinuities, where the fluid velocity gradient is significant, often appear as abrupt variations in velocity profile or fluid properties like viscosity or density, due to changes in fluid nature or flow regime transitions. Finally, shock discontinuities, where fluid density, pressure, temperature, and speed change abruptly, are often encountered in supersonic flows.

Shear and contact discontinuities do not affect propagation speed, allowing treatment without additional complications associated with wave propagation in the fluid. However, shock discontinuities, primarily in supersonic flows, do. They occur when fluid moves faster than sound speed and encounters an obstacle or sudden geometry change. This disturbance causes fluid accumulation and compression before the obstacle, creating high-pressure, high-density zones. A shock wave forms, a quasi-discontinuous surface where fluid properties change abruptly and significantly, propagating through the fluid, significantly affecting flow around the obstacle.

Shock capturing methods

Shock discontinuities, the most complex due to their impact on propagation speed, pose a major challenge in high-speed flow simulations. A key issue is "aliasing," where the numerical representation of the solution is truncated or simplified, leading to difficulties in high-gradient areas like shocks. This simplification can cause unphysical numerical oscillations, distorting the actual solution, and may even yield completely unrealistic results, like negative densities or pressures [15].

Developing effective shock capturing methods is crucial. This requires sufficiently fine meshes to accurately capture flow nuances and stabilization techniques to mitigate solution irregularities while maintaining precision.
Two primary numerical methods are used: finite volumes and finite elements, each offering distinct shock capturing strategies.

Finite Volume Method

The finite volume method applies conservation equations meticulously within each computational domain cell. Managing exact inflow and outflow fluxes per cell ensures adequate representation and maintenance of physical properties within each cell. In other words, this method works with cell-internal average values. From these averages, a more detailed approximation is reconstructed, used to evaluate and refine the overall solution.

Finite volume method shock capturing techniques include limiters, the ENO-WENO approach, and artificial viscosity, each addressing shock discontinuity challenges.

Limiters modulate discretization schemes, dynamically adapting cell-internal gradient reconstructions, thus avoiding excessive approximations that might generate numerical instabilities near discontinuities. This approach maintains precision while ensuring simulation stability[16].

The ENO (Essentially Non-Oscillatory) approach and its improved variant, WENO (Weighted Essentially Non-Oscillatory), represent significant advancements in data reconstruction. ENO adaptively chooses the best stencils to avoid discontinuities [17, 18], while WENO refines this technique by variably weighting different stencils, optimizing reconstruction in high-gradient areas. These methods are particularly effective for managing discontinuities with high precision without introducing oscillations [17, 19, 20].

Artificial viscosity increases numerical dissipation in high-gradient areas, like shocks. By introducing controlled additional viscosity, this method smooths discontinuities and prevents undesirable oscillations, contributing to overall simulation stability.

Finite Element Method

In contrast to the finite volume method, the finite element method does not reconstruct the solution from cell-average values. Instead, it directly computes the solution using a linear combination of specific interpolation functions for each element. These interpolation functions are crucial for defining physical parameter variations within each element. They are expressed in terms of parameter values at nodes.
Various shock capturing methods are available for this category, including flux limiter techniques, entropy-stable methods, and the use of artificial viscosity.

Flux limiters are designed to minimize undesirable numerical oscillations near strong discontinuities like shocks. These limiters dynamically adjust reconstruction or interpolation schemes of fluxes across finite elements, based on fluid local gradients. The goal is to preserve fine flow details while avoiding numerical artifacts [21].

Entropy-stable methods are based on entropy conservation principles. They ensure numerical solutions respect physical entropy laws, particularly in shock discontinuity areas. By preserving this thermodynamic property, entropy-stable methods provide more physically realistic solutions.

Artificial viscosity in finite elements differs from its finite volumes application, where it is integrated into cell interface flux calculations. In finite elements, this approach is adapted to the specific nature of elements and their interconnection. Artificial viscosity is added to the equations governing fluid behavior within each element. It acts directly on deformation or stress terms, thereby smoothing solutions where high gradients, typical of shocks, are detected.
The advantage of this method in finite elements lies in its ability to maintain simulation continuity and consistency across the entire mesh. By locally adjusting viscosity for each element, it allows for greater flexibility and precision, especially in complex problem geometries or when different physical phenomena interactions are significant. However, using artificial viscosity in finite elements requires careful calibration to avoid over-dissipating essential physical flow characteristics.

Excess viscosity can mask important flow details, while insufficient amounts may not stabilize the simulation in shock regions.

4   Research plan

This work was carried out using ForDGe, a Cartesian solver based on a discontinuous Galerkin formulation, currently under development for solving coupled problems in physics. The shock-capturing method employed involves artificial viscosity combined with the discontinuous Galerkin finite element formulation. ForDGe was developed at the University of Liège by the DoT group.

The primary objective is to develop an artificial viscosity method based more on physical principles, while assessing its ability to accurately replicate the solution. Although this viscosity method has yielded satisfactory results in some cases, it remains in a preliminary stage of development. Numerous issues have been identified, and efforts have been made to address them. Currently, this method is not yet fully developed in a general sense, and it is challenging to guarantee its reliability concerning the stabilization of any simulations.

However, despite the challenges encountered, this method has yielded some results, highlighting its potential. Furthermore, innovative outcomes have been achieved by incorporating the use of Bernstein polynomials in an attempt to improve the stability of this method. To provide a critical assessment of this approach, a comparison will be conducted between the results obtained using this method and those obtained with a second artificial viscosity method based on the Laplacian, which has already demonstrated its effectiveness in other research works.

The first test case employed is that of Kelvin-Helmholtz instabilities in a non-viscous context, which is of crucial importance for assessing the effectiveness of artificial viscosity methods. In the absence of viscosity, this test case can quickly lead to simulation failures, making it a suitable ground for determining whether artificial viscosity methods can ensure stability.
This test case is specifically designed to study fluid instabilities characterized by velocity variations. These instabilities highlight a method's sensitivity in detecting and modeling such variations, which is a crucial skill for evaluating its ability to handle complex flows.
A significant advantage of this test case is that it does not involve shock phenomena. This allows for a specific focus on the ability of artificial viscosity methods to accurately handle instabilities without the interference of shocks. Thus, one can isolate the aspects related to artificial viscosity and numerical stability.

The second test case is the Inviscid strong vortex shock wave interaction. It holds particular relevance for evaluating artificial viscosity methods due to its tendency to generate spurious oscillations. These parasitic oscillations can occur when numerical methods are used to simulate complex flows, such as those involving shockwaves and vortices.
The interaction between the vortex and the shockwave in this test case can lead to abrupt pressure and velocity variations, making it prone to the formation of these spurious oscillations. Therefore, it highlights the ability of an artificial viscosity method to effectively attenuate or prevent these undesirable oscillations while maintaining the accuracy of simulation results. This makes this test case even more relevant for evaluating the reliability and stability of the artificial viscosity method

in complex and unstable flow conditions. In summary, it presents a significant challenge for numerical methods and allows for an assessment of their capability to handle spurious oscillations, a crucial aspect of accurately modeling flows.

Finally, the chosen interpolation orders for the results are 3 and 4, aiming to explore high-precision approaches. In practice, these orders are used. However, due to the increased instability of the artificial viscosity method, using even higher interpolation orders has often led to simulation failures and unusable results.

Chapter 2

Description of the methods

The non-linear nature of fluid dynamics becomes evident through the unsteady response to a steady excitation. The Navier-Stokes equations, which describe fluid behavior, do not allow for analytical solutions under real conditions due to the non-uniqueness of these solutions. These equations can quickly become uncontrollable, leading to chaotic and unpredictable flow, known as turbulence. Turbulence can be generated by several factors, including shock generation when the fluid reaches a transonic regime, exceeding the speed of sound (Mach 1). To study such flows, it is essential to use numerical resolutions capable of approximating solutions for this type of flow. This section primarily focuses on three crucial points. Firstly, it provides a brief review of the equations of compressible fluid mechanics. Secondly, it discusses the spatial discretization using the Discontinuous Galerkin (DG) method and temporal discretization using the fourth-order Runge-Kutta (RK4) scheme. Finally, it describes the methodology used for shock capture in the various simulations that will be presented later.

1 Compressible Flow Equations

Assuming that $t_f > 0$ denotes the final time, $\Omega \subseteq \mathbb{R}^d$ represents the domain with $1 \leq d \leq 3$, the governing equations are the unsteady and compressible Navier-Stokes equations formulated in their conservative form,

$$\frac{\partial \vec{u}}{\partial t} + \vec{\nabla} \cdot \vec{\vec{f}}(\vec{u}) + \vec{\nabla} \cdot \vec{\vec{d}}(\vec{u}, \vec{\nabla}\vec{u}) = 0, \qquad \text{in } \Omega \times (0, t_f), \tag{2.1a}$$

$$\vec{B}(\vec{u}, \vec{\nabla}\vec{u}) = 0, \qquad \text{on } \partial\Omega \times (0, t_f), \tag{2.1b}$$

$$\vec{u} - \vec{u}_0 = 0, \qquad \text{on } \partial\Omega \times (0). \tag{2.1c}$$

With the state variables defined as $\vec{u} = (\rho, \rho\vec{v}, \rho E)$ and $\vec{u}_0$ representing the initial condition. These equations include the boundary operator $\vec{B}(\vec{u}, \vec{\nabla}\vec{u})$, as well as the inviscid and viscous fluxes, $\vec{\vec{f}}(\vec{u})$ and $\vec{\vec{d}}(\vec{u}, \vec{\nabla}\vec{u})$, respectively.

The expressions for the fluxes are subject to certain simplifying assumptions, considering a Newtonian fluid behavior akin to a perfect gas in thermodynamic equilibrium, and following the thermal conduction described by Fourier's law (Eq. 2.3c).

$$\vec{\vec{f}}(\vec{u}) = \begin{bmatrix} \rho\vec{v} \\ \rho\vec{v}\vec{v} + \delta\ p \\ \vec{v}(\rho E + p) \end{bmatrix}, \qquad \vec{\vec{d}}(\vec{u}, \vec{\nabla}\vec{u}) = -\begin{bmatrix} 0 \\ \vec{\vec{\tau}} \\ \vec{v}\cdot\vec{\vec{\tau}} - \vec{f} \end{bmatrix}. \tag{2.2}$$

Here, δ represents the Kronecker delta, $\vec{\vec{\tau}}$ denotes the viscous stress tensor, $\vec{f}$ signifies the heat flux, and p corresponds to the thermodynamic pressure. Under the aforementioned assumptions, the pressure, viscous stress tensor, and heat flux are calculated as follows :

$$p = (\gamma - 1)(\rho E - \rho|\vec{v}|^2/2) \tag{2.3a}$$

$$\tau_{ij} = \mu\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i} - \frac{2}{3}\frac{\partial v_k}{\partial x_k}\delta_{ij}\right) + \beta\frac{\partial v_k}{\partial x_k}\delta_{ij} \tag{2.3b}$$

$$f_j = -\kappa\frac{\partial T}{\partial x_j} \tag{2.3c}$$

where $i, j = 1, ..., d$, T denotes the temperature, μ represents the dynamic (shear) viscosity, β signifies the bulk viscosity, κ denotes the thermal conductivity, and $\gamma = c_p/c_v > 1$ stands for the specific heat ratio. Additionally, it is important to note that the Stokes hypothesis implies $\beta = 0$. Remarks: It is worth mentioning that the Euler equations are obtained by nullifying the diffusive flux, $\vec{\vec{d}} = 0$.

2 Spatial discretization - Discontinuous Galerkin Method

Unlike traditional finite element methods, discontinuous Galerkin (DG) methods offer greater flexibility by using basis functions that are not constrained by continuity conditions at interfaces. This characteristic allows for "p-adaptive" refinement [22] for each element, where the interpolation order can be locally adjusted according to the problem's requirements. DG methods have shown particular effectiveness in handling unstructured meshes [23], where the geometry of elements can vary significantly, and they are well-suited for parallelizing computations [24] using high-performance computing techniques.

The central core of this method lies in managing the fluxes at interfaces. Since elements are locally isolated from each other, the exchange of information must occur through interface fluxes. The approach to handling these fluxes strongly depends on the nature of the equations under consideration. Hyperbolic equations involving a convective flux $\vec{\vec{f}}$ and elliptic equations involving a diffusive flux $\vec{\vec{d}}$ are treated using different methods. This section of the study begins with a general introduction to the method, followed by an explanation of the approaches for the specific fluxes corresponding to each type of equation.

2.1 General formulation

The spatial discretization of the governing equations Eq. 2.1 can be done in a completely general way by combining the two fluxes to form a generic flux $\vec{\vec{g}} = \vec{\vec{f}} + \vec{\vec{d}}$. Thus, the set of equations can be written in the form,

$$\frac{\partial\vec{u}}{\partial t} + \vec{\nabla}\cdot\vec{\vec{g}} = 0. \tag{2.4}$$

With DG method, the domain Ω can be decomposed into a sum of non-overlapping elemental domain,

$$\Omega = \sum_e \Omega_e. \tag{2.5}$$

To obtain a weak formulation of the problem, Equation Eq. 2.4 is multiplied by an arbitrary function v belonging to the space of test functions $\mathcal{V}$, which consists of polynomial functions of order p over these elements. The entire expression is integrated over the domain, which, as implied by the previously stated property, yields:

$$\sum_e \int_{\Omega_e} v \left(\frac{\partial \vec{u}}{\partial t} \, dV + \vec{\nabla} \cdot \vec{g} \right) \, dV = 0, \qquad \forall v \in \mathcal{V}. \tag{2.6}$$

By decomposing the second term, $v\vec{\nabla} \cdot \vec{g}$, and applying the divergence theorem, it is quite straight-forward to obtain the following form,

$$\sum_e \int_{\Omega_e} v \frac{\partial \vec{u}}{\partial t} \, dV - \sum_e \int_{\Omega_e} \vec{\nabla} v \cdot \vec{g} \, dV + \sum_e \oint_{\partial \Omega_e} v \vec{g} \cdot \vec{n} \, dS = 0, \qquad \forall v \in \mathcal{V}. \tag{2.7}$$

All element interfaces are counted twice, once for each element they belong to and once with the neighboring element. By fixing a normal vector $\vec{n}$ for a face f, as illustrated in Figure Fig. 2.1, a nomenclature can be established such that e^+ refers to the element where the normal vector points in the same direction as the face normal $\vec{n}^+ = \vec{n}$. Consequently, e^- designates the neighboring element where $\vec{n}^- = -\vec{n}$. The values in the corresponding elements are denoted with this same nomenclature: u^+ and u^- for the solutions, v^+ and v^- for the test functions, and so on. The equation then becomes,

$$\sum_e \int_{\Omega_e} v \frac{\partial \vec{u}}{\partial t} \, dV - \sum_e \int_{\Omega_e} \vec{\nabla} v \cdot \vec{g} \, dV + \sum_f \oint_f \left(v^+ \vec{g}(u^+) - v^- \vec{g}(u^-) \right) \cdot \vec{n} \, dS = 0, \qquad \forall v \in \mathcal{V}. \tag{2.8}$$

From this point on, it is essential to introduce some operators to enhance the readability and comprehension of the mathematical equations concerning DG methods. The jump operator $[[\cdot]]$ and the average operator $\langle \cdot \rangle$ can be defined for scalar and vector quantities, respectively,

$$\begin{aligned}
[[\, a \,]] &= \left(a^+ - a^- \right) \vec{n}, & \langle \, a \, \rangle &= \frac{a^+ + a^-}{2}, \\
[[\, \vec{a} \,]] &= \left(\vec{a}^+ - \vec{a}^- \right) \cdot \vec{n}, & \langle \, \vec{a} \, \rangle &= \frac{\vec{a}^+ + \vec{a}^-}{2}.
\end{aligned} \tag{2.9}$$

Incorporating these operators and replacing the integrand with a more general notation, the interface flux $\gamma = \left[\left[v\vec{g} \right]\right]$. The equation becomes,

$$\sum_e \int_{\Omega_e} v \frac{\partial \vec{u}}{\partial t} \, dV - \sum_e \int_{\Omega_e} \vec{\nabla} v \cdot \vec{g} \, dV + \sum_f \oint_f \gamma \left(u^+, u^-; v^+, v^-; \vec{n} \right) \, dS = 0, \qquad \forall v \in \mathcal{V}. \tag{2.10}$$

This last term γ is an addition compared to classical FEM methods. Indeed, it is expected that for a continuous solution, $u^+ = u^- = \tilde{u}$, and $\vec{g}(u^+) = \vec{g}(u^-)$, the interface flux γ vanishes. Lastly, it is essential to ensure that this latter equation is consistent, meaning that the exact solution must be satisfied.

Having the general formulation and using Galerkin approach,

$$a(\vec{x}, t) = \sum_i a_i(t)\phi_i(\vec{x}) \quad \text{and} \quad \vec{\nabla}a(\vec{x}, t) = \sum_i a_i(t)\vec{\nabla}\phi_i(\vec{x}), \tag{2.11}$$

the equation Eq. 2.4 transforms into a semi-discretized equation that can be solved separately for each element,

$$\frac{du_i}{dt}(t) = r_i(u_j, t). \tag{2.12}$$

Here, the index i represents the nodes of element e. This formulation allows for the temporal evolution of the solution at each node i to be determined independently based on the local contributions from the neighboring nodes j within the same element.
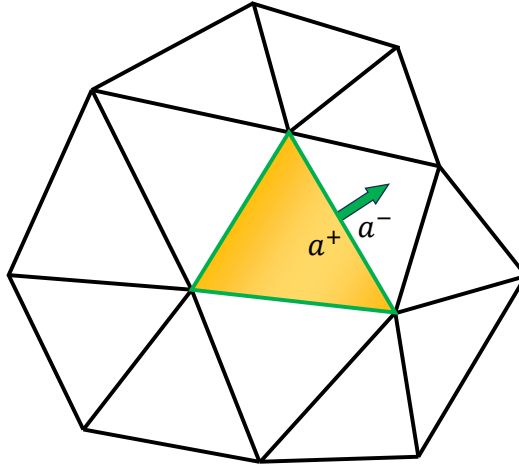
Figure 2.1: Definition of the normal per face and neighbouring values of a.

2.2   Convective flux

By omitting the viscous part of the Navier-Stokes equations and considering plane waves aligned with the face such that their propagation directions are equivalent to the normal of the face defined previously, $\vec{n}^+$, the governing system of equations becomes a linear advection system :

$$\frac{\partial \vec{u}}{\partial t} + \mathcal{A}^n \cdot \vec{\nabla}\vec{u} = 0, \tag{2.13a}$$

$$\mathcal{A}^n = \mathcal{A} \cdot \vec{n}, \ \mathcal{A}^n \in \mathcal{R}^{N \times N}. \tag{2.13b}$$

As a hyperbolic system, $\mathcal{A}$ has N real eigenvalues λ_i such that the matrix can be decomposed as

$$\mathcal{A} = R\Lambda L, \tag{2.14}$$

with $\Lambda = \text{diag}(\lambda_1, ..., \lambda_N)$ being the diagonal matrix containing the eigenvalues, and R and L being the sets of right and left eigenvectors, respectively.
The interface flux can then be defined as

$$\gamma^{convectif}(\vec{u}^+, \vec{u}^-; \vec{n}) = \mathbf{A}^n \frac{q^+ + q^-}{2} + |\mathbf{A}^n| \frac{q^+ - q^-}{2}, \tag{2.15}$$

with

$$|\mathbf{A}^n| = R|\Lambda|L. \tag{2.16}$$

The first term is the numerical average flux, representing an estimation of the flux crossing the interface between two neighboring cells. The second term, on the other hand, is a numerical dissipation term added to the flux discretization to ensure the stability of the scheme and prevent numerical oscillations around discontinuities.
The Lax-Friedrichs flux scheme combines these two terms to obtain an approximation of the flux crossing the interface. While this scheme is easy to implement, it may lead to excessive diffusion and reduced numerical accuracy for complex hyperbolic equations. Therefore, more advanced flux schemes can be employed to overcome these limitations [25].

2.3 Diffusive flux

The spatial discretization of the diffusive flux is a crucial step in solving the Navier-Stokes equations. It is through this step that shear and bulk viscosities, as well as thermal conductivity, are taken into account, as shown in Eq. 2.2. Additionally, this stage is where the introduction of artificial viscosity may be applied. Therefore, it is relevant to dedicate time to a more in-depth study of this part.

The interior penalty method is based on the concept of the boundary penalty method, first introduced by Nitsche [26]. The main idea behind these methods is to avoid the imposition of strong Dirichlet boundary conditions, which may not be suitable when boundary conditions vary rapidly and significantly. Instead, the interior penalty method proposes using penalties to weakly enforce the continuity of the solution and the boundary conditions. To achieve this, the method introduces an additional term with a penalty factor σ_f in the formulation, aiming to minimize the average interpolation error between the numerical solution u and the prescribed boundary condition value u^*. This leads to a reduction in the overall error caused by smoother variations in the solution [27]. By adopting this approach, the general form of the Discontinuous Galerkin Method (DGM) for an elliptic problem with a diffusive flux $\vec{d} = \mu_d \vec{\nabla} \vec{u}$ becomes,

$$\sum_e \int_{\Omega_e} v \frac{\partial \vec{u}}{\partial t} \, dV_e + \sum_e \int_{\Omega_e} \vec{\nabla} v \cdot \mu_d \cdot \vec{\nabla} \vec{u} \, dV_e$$

$$+ \sum_f \int_{\partial \Omega_e} \sigma_f \, [[v]] \cdot [[\vec{u}]] \, dS_e - \sum_f \int_{\partial \Omega_e} [[v]] \cdot \langle \mu_d \cdot \vec{\nabla} \vec{u} \rangle \, dS_e - \sum_f \int_{\partial \Omega_e} \theta \, [[\vec{u}]] \cdot \langle \mu_d \cdot \vec{\nabla} v \rangle \, dS_e = 0.$$

$$\tag{2.17}$$

The penalty term highlighted in green should be followed by the red term to maintain consistency, as emphasized by Hillewaert [27]. The final term in blue is introduced to add or omit symmetry depending on the value of θ. It is important to note that the value of θ does not affect consistency.

However, it is crucial to underline that depending on the specific value of θ, different types of interior penalty methods may come into play.

- $\theta = 1$ the symmetric interior penalty (SIPDG) method.

- $\theta = 0$ the incomplete interior penalty (IIPDG) method.

- $\theta = -1$ the non-symmetric interior penalty (NSIPDG) method.

The incomplete interior penalty method (IIPDG) will be employed to compute the diffusive flux. One of the major disadvantages of the interior penalty methods lies in the penalty factor σ_f, which lacks a precise definition in the literature. This parameter significantly impacts the simulation since it must be chosen carefully: it should be sufficiently high to enforce the boundary conditions accurately while avoiding excessively high values that could lead to poor conditioning of the resulting linear system. However, Koen Hillewaert proposes a relationship to determine an optimal value for the penalty coefficient [27]. ForDGe then applies this specific relationship to a structured mesh composed of quadrangles and a particular diffusive flux $\vec{d}$ defined earlier,

$$\sigma_f = \mu_d(p+1)^2 \frac{A(e)}{V(e)}. \tag{2.18}$$

The diffusivity value, denoted as μ_d, is determined by evaluating the spectral radius of the diffusion matrix D. In the case of ForDGe, this diffusivity value is computed in the following manner

$$\mu_d = \left(2\mu + \beta + \frac{\kappa}{C_v}\right)/\rho. \tag{2.19}$$

The values of μ, β, and κ are determined by taking the maximum value of the considered element and include both the physical quantity and, if applicable, the artificial quantity. Determining the maximum diffusivity per element is crucial to ensure the trace inverse inequality (Eq. 2.20) that guarantees coercivity. This coercivity, in turn, is a fundamental assumption of the Lax-Milgram theorem, ensuring the uniqueness of the solution. By selecting a maximum diffusivity per element, it is ensured that the right-hand side of the inequality is always greater than or equal to the left-hand side, given that the maximum over the element includes the faces.

$$\int_{\partial\Omega_e} \mu_d \, u^2 \, dS_e \le (p+1)^2 \frac{A(e)}{V(e)} \int_{\Omega_e} \mu_d \, u^2 \, dV_e, \quad \forall u \in \mathcal{P}_p(e), \tag{2.20}$$

with $\mathcal{P}_p(e)$ a polynomial function space of order p defined on the element e.

Note   The following equation represents the diffusive flux through the diffusion matrix for a one-dimensional problem.

$$d = \underbrace{\begin{bmatrix} 0 & 0 & 0 \\ -\left(\frac{4}{3}\mu + \beta\right)v/\rho & \left(\frac{4}{3}\mu + \beta\right)/\rho & 0 \\ -\left(\frac{4}{3}\mu + \beta + \frac{\kappa}{C_v}\right)v^2/\rho + \frac{\kappa}{C_v}E/\rho & \left(\frac{4}{3}\mu + \beta + \frac{\kappa}{C_v}\right)v/\rho & -\frac{\kappa}{C_v\rho} \end{bmatrix}}_{D} \frac{\partial}{\partial x} \begin{bmatrix} \rho \\ \rho v \\ \rho E \end{bmatrix}. \tag{2.21}$$

3 Time discretization - Runge Kutta 4

When spatial discretization is performed using the Discontinuous Galerkin method, it leads to a semi-discretized equation Eq. 2.12. To fully solve this equation, it is also necessary to discretize it in the temporal domain. To achieve this, a wide range of methods is available, broadly categorized into two main groups: explicit methods and implicit methods. Each of these methods has its advantages and disadvantages [25].

Explicit methods are conditionally stable and are subject to limitations on the time step used, due to stability conditions such as the CFL (Courant-Friedrichs-Lewy) condition. These methods have a relatively low CPU cost per iteration as they do not involve matrix inversion, but their limited stability range requires a larger number of time steps.
Implicit methods, on the other hand, are unconditionally stable, allowing for a generally significantly larger time step compared to explicit methods. However, their CPU cost per iteration is higher due to the matrix inversion that must be performed at each time step. Additionally, these matrices require extra memory depending on the chosen solution method.
It is important to note that the choice between explicit and implicit methods depends on the specific characteristics of the problem to be solved, such as its numerical stability, computational time constraints, and precision requirements.

In the context of this thesis, an explicit method from the family of Runge-Kutta methods will be employed. Discontinuous Galerkin Runge-Kutta schemes have been extensively studied, notably in the works of Cockburn [28–33]. The fundamental idea of Runge-Kutta methods is to approximate the solution of a differential equation using a weighted linear combination of different slopes calculated at different points in the considered time interval. These slopes are obtained by evaluating the derivative of the solution at each stage of the scheme. By integrating these weighted slopes over the entire time interval, an estimate of the solution at the next time step can be obtained. These methods achieve high accuracy orders by using multiple calculation steps to improve the numerical precision of the solution at the expense of memory, as each solution of the intermediate stages must be stored. The most popular version, and the one used here, is the fourth-order Runge-Kutta method defined by

$$k_i^{(1)} = r_i(u_j^n, t^n) \tag{2.22}$$

$$k_i^{(2)} = r_i(u_j^n + \Delta t k_j^{(1)}/2, t^n + \Delta t/2) \tag{2.23}$$

$$k_i^{(3)} = r_i(u_j^n + \Delta t k_j^{(2)}/2, t^n + \Delta t/2) \tag{2.24}$$

$$k_i^{(4)} = r_i(u_j^n + \Delta t k_j^{(3)}, t^n + \Delta t) \tag{2.25}$$

where r_i is the residual of an element at the node i. The solution at time $t^{n+1} = t^n + \Delta t$ is computed on each element by

$$u_i^{n+1} = u_i^n + \frac{\Delta t}{6} \left(k_i^{(1)} + 2k_i^{(2)} + 2k_i^{(3)} + k_i^{(4)} \right). \tag{2.26}$$

The summation of coefficients in front of the various k_i terms being equal to 1 ensures consistency [25].

4 Artificial viscosity

The amount of added artificial viscosity has a real impact on the simulation: a too high value leads to excessive dissipation, smoothing the solution without providing additional information, while a too low value renders the simulation unable to capture the shock. Ultimately, artificial viscosity represents a delicate compromise between stability and the reliability of the simulation. To approach the topic smoothly, it is useful to have a comprehensive understanding of how artificial viscosity works. The simulation crashes due to a discontinuity generated by the shock. Thus, artificial viscosity is added around the shock to sufficiently smooth the solution, allowing the capture of the shock while preserving smooth regions.

A crucial first step in artificial viscosity is dedicated to locating the shock in the simulation, identifying problematic and smooth areas. For this localization, the use of one or more sensors is necessary. However, the nature of these sensors is not unique. Shock localization can be achieved through physical quantities [2, 34], or through purely numerical quantities based on solution smoothing [35, 36].
The second major step of artificial viscosity concerns its proper injection into the simulation, along with its dissipative quantity. Two main methods will be discussed later: a method based on the Laplacian, which adds a Laplacian proportional to artificial viscosity in the Navier-Stokes equations, and a Newtonian-based method, which directly adds artificial viscosities to physical viscosities, generating resulting viscosities as the sum of physical and artificial components.
Finally, the third and last important step of artificial viscosity concerns its interpolation to quadrature points. In a numerical simulation, calculations are performed through quadrature points. However, artificial viscosity can be calculated directly at these points or interpolated from other points, such as interpolation points, allowing for smoothing of this artificial viscosity. This last step is not mandatory and is only used in the case of a pointwise artificial viscosity.

This section aims to respectively describe the sensors, injection strategies (calculation of artificial viscosity), and interpolation methods that were used to conduct the simulations.

4.1 Sensors

For better comprehension, it will be subdivided into two parts. The first will introduce two sensors (PerssonPeraire and Hennemann) based on a solution regularity criterion, while the second will present three sensors based on a physical criterion.

Although the nature of the various sensors is different, they all share a common characteristic. To ensure stability and accuracy, it is imperative to establish lower and upper bounds. It is evident that the sensor intensity will impact the amount of viscosity added to the simulation; therefore, these bounds ensure the absence of negative values and limit viscosity quantities to reasonable levels. Finally, for the sake of simplicity and comparability, the sensors will be restricted to the range [0,1].

4.1.1 Regularity Criteria

Persson Peraire sensor   This detector relies on the regularity of the solution, achieved by representing the solution within each element as a polynomial expansion. In the case of smooth

solutions, the coefficients in this expansion are anticipated to decay rapidly. Conversely, when the solution lacks smoothness, the discontinuity's intensity will influence the rate of decay of the expansion coefficients [35].

Let consider two expansions for each element, the first one of order p and the second one of order $p - 1$, such that

$$u = \sum_{i=1}^{N(p)} = u_i \phi_i, \tag{2.27a}$$

$$\hat{u} = \sum_{i=1}^{N(p-1)} = u_i \phi_i, \tag{2.27b}$$

where $N(p)$ represents the number of terms in the expansion of order p, and ϕ_i are the basis functions.

The Discontinuity Detector can be defined

$$s_e = log_{10} \frac{(u - \hat{u}, u - \hat{u})_e}{(u, u)_e}, \tag{2.28}$$

with $(\cdot, \cdot)_e$ the standard inner product $L_2(\Omega_e)$.

The sensor is reconfigured in a way that it is defined as

$$\hat{s}_e = \frac{(s_e - (s_0 - \kappa))}{2\kappa} \text{ with } \hat{s}_e = \begin{cases} 0 & \text{if } \hat{s}_e \leq 0, \\ 1 & \text{if } \hat{s}_e \geq 1. \end{cases} \tag{2.29}$$

Here, s_0 is a threshold beyond which the detector s_e should activate, and κ is an interval over which the detector increases from 0 to 1. The only modification made compared to Persson Peraire's article is the reconfiguration of the sensor to achieve a range from 0 to 1. This modification has no impact on artificial viscosity since the minimum and maximum bounds are adjusted accordingly. Its sole effect lies in facilitating the sensor reading (see Fig. 2.2).
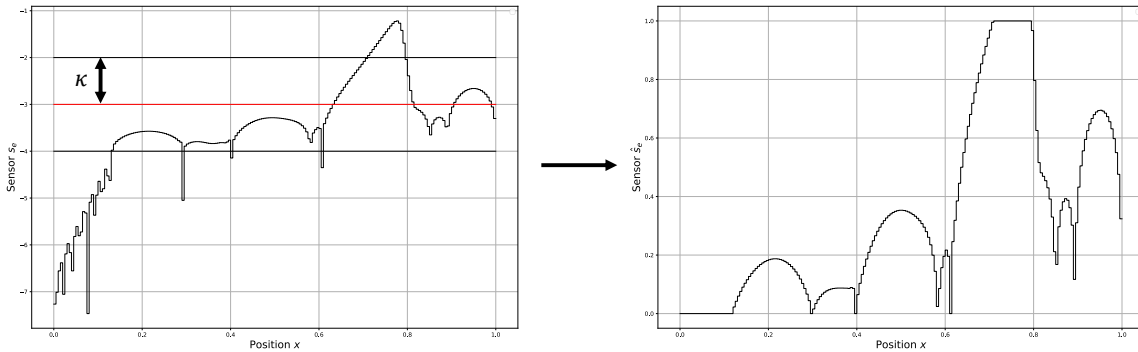
Figure 2.2: Illustration depicting the sensor and its reconfigured version

Hennemann Sensor Hennemann proposes a method to estimate the extent of sub-resolution [36] by drawing from the work of Persson and Peraire. The core idea is to compare the modal energy of the highest-degree polynomial modes with the overall modal energy of the quantity

under consideration. More detailed explanation exceeds the scope of this thesis. For in-depth insights, it is advisable to refer to Hennemann's article.

In order to define the sensor, it is first useful to introduce the following quantity

$$\alpha = \frac{1}{1 + \exp\left(\frac{-s}{T}\right)}, \tag{2.30}$$

$$T(p) = 1.8(p+1)^{\frac{1}{4}}, \tag{2.31}$$

$$s = 9.21024. \tag{2.32}$$

The sensor can then be defined as

$$\hat{s}_e = \begin{cases} 0 & \text{if } \alpha \leq 0.0001, \\ min(\alpha, 0.5) & \text{if } \alpha > 0.0001. \end{cases} \tag{2.33}$$

Note   Even though the maximum value of the sensor only reaches 0.5, it is still bounded between 0 and 1.

4.1.2   Physical Criteria

Given that these sensors rely on the intrinsic behavior of the fluid, it is advisable to apply a similar limiting function to each of them. It is important to note that this function should be smooth enough not to disrupt the numerical simulation with discontinuities. The limiting function $l(s; s_0; s_{max})$ depends on the sensor value s at the considered point, the minimum detection value s_0 and the maximum limit the sensor can reach s_{max}. These values may vary depending on the sensor used. The function can be defined as follows:

$$l(s; s_0; s_{max}) = \begin{cases} 0 \\ l_{min}\left(l_{max}(s - s_0) - s_{max}\right)/s_{max} + 1 \end{cases}, \tag{2.34}$$

where

$$l_{max}(s) = \frac{s}{\pi}\arctan(100s) + \frac{s}{2} - \frac{1}{\pi}\arctan(100) + \frac{1}{2}, \tag{2.35a}$$

$$l_{min}(s) = s - l_{max}(s). \tag{2.35b}$$

Shock sensor $\hat{s}_\beta$ is a combination of the dilatation sensor s_θ and the anti-vorticity sensor s_ω, proposed respectively by Moro [34] and Ducros [37]. The dilatation sensor is activated in response to a shock wave, while the anti-vorticity sensor disappears in regions of the flow dominated by vorticity or in regions that are not shocked, where the sensor remains significant due to sub-resolution effects. The sensor is introduced as follows,

$$\hat{s}_\beta(\vec{x}) = l(s_\beta; s_{\beta,0}; s_{\beta,max}), \qquad s_\beta(\vec{x}) = s_\theta(\vec{x}) \cdot s_\omega(\vec{x}). \tag{2.36}$$

These sensors are formulated as follows,

$$s_\theta(\vec{x}) = -g(h, p)\frac{\vec{\nabla} \cdot \vec{v}}{c^*}, \tag{2.37}$$

$$s_\omega(\vec{x}) = \frac{(\vec{\nabla} \cdot \vec{v})^2}{(\vec{\nabla} \cdot \vec{v})^2 + |\vec{\nabla} \times \vec{v}| + \varepsilon_\omega}, \tag{2.38}$$

with $g(h, p)$ a geometric factor to be defined, c^* the critical speed of sound, and ε_ω representing machine epsilon. The values of the lower and upper bounds are given by $s_{\beta,0} = 0.01$ and $s_{\beta,max} = 2$, respectively. They are chosen to prevent the introduction of artificial bulk viscosity far from shocks, thus keeping minimal impact on the flow behavior [2].

Thermal sensor $\hat{s}_\kappa$ is designed to detect thermal gradients larger than the grid resolution that may lead to nonlinear instabilities. For a structured mesh composed of quadrilaterals, the detector is defined as follows,

$$\hat{s}_\kappa = l(s_\kappa; s_{\kappa,0}, s_{\kappa,max}), \quad s_\kappa(\vec{x}) = g(h, p) \frac{|\vec{\nabla}_\xi T|}{Tt}, \tag{2.39}$$

where $T_t(\vec{x})$ denotes the stagnation temperature, and $\vec{\nabla}_\xi T$ is the temperature gradient under the metric of the reference.

By definition, the sensor should only activate when the thermal gradient cannot be resolved by the grid anymore. Therefore, it is logical to set the lower limit of the sensor as $s_{\kappa,0} = 1$. Furthermore, it is desired that since shocks are stabilized through a mechanism independent of the sensor $\hat{s}_\kappa$, the sensor should vanish in shock waves. Finally, the upper bound $s_{\kappa,max}$ is fixed at 2. to enhance nonlinear stability [2].

Shear sensor The procedure is similar for the shear sensor. It aims to detect velocity gradients larger than the grid resolution. It can be defined as follows,

$$\hat{s}_\mu = l(s_\mu; s_{\mu,0}, s_{\mu,max}), \quad s_\mu(\vec{x}) = g(h, p) \frac{||\mathcal{L}(\vec{v})||_2}{v_{max}} \tag{2.40}$$

where $|| \cdot ||_2$ denotes the spectral norm,

$$\mathcal{L}(\vec{v}) = \vec{\nabla}_x \vec{v} - diag(\vec{\nabla}_x \vec{v}) \tag{2.41}$$

and

$$v_{max}(\vec{x}) = \left(|\vec{v}|^2 + \frac{2}{\gamma - 1} c^2 \right)^{1/2} \tag{2.42}$$

is the *maximum isentropic velocity*, defined as the velocity the flow if all total energy was converted into kinetic energy through an isentropic expansion.

It is noteworthy to observe the presence of the velocity divergence in Eq. 2.41, which is introduced to ensure that the sensor vanishes in shock waves, allowing only the bulk viscosity to stabilize the shock and thereby reducing the method's impact on vortices. Finally, for the same reasons mentioned earlier, the limitations are set as $s_{\mu,0} = 1$ and $s_{\mu,max} = 2$.

Note The thermal and shear sensors described above are expressly designed for shock detection. They only consider areas where the considered gradients cannot be captured by the grid. For test cases that lack any shocks, as is the case with Kelvin-Helmholtz instabilities, the gradients are no longer calculated using the reference metric but instead rely on the physical gradient.

4.2   Injection strategies

Now that the different detectors have been defined, it is important to understand how to calculate and introduce artificial viscosity. The literature offers numerous methods for this purpose. For the current study, two approaches for incorporating artificial viscosity will be employed: a Laplacian-based approach and a Newtonian-based approach.

Considering that artificial viscosity is influenced by various user-related parameters and manipulations aimed at achieving desired results, it may be prudent to establish an upper limit for this viscosity using various physical and/or logical parameters. This approach would help better constrain the user-defined parameters used.

Laplacian-based method   Navier-Stokes equations will receive an additional contribution through the addition of a Laplacian term, resulting in the governing equation of the problem being expressed as

$$\frac{\partial \vec{u}}{\partial t} + \vec{\nabla} \cdot \vec{f}(\vec{u}) + \vec{\nabla} \cdot \vec{d}(\vec{u}, \vec{\nabla}\vec{u}) = \vec{\nabla} \cdot \left(\varepsilon_{AV} \vec{\nabla}\vec{u} \right). \tag{2.43}$$

This artificial viscosity is bounded by a value ε_0, which depends on a magnitude factor ν_0 defined by the user, a geometric factor, and a quantity for obtaining the scaling viscosity λ.

$$\varepsilon_{AV} = \varepsilon_0 \; 0.5 \times \left[1 + \sin \left((\hat{s} - 0.5)\pi \right) \right], \tag{2.44}$$

$$\varepsilon_0 = \nu_0 \; g(h, p) \; \lambda, \tag{2.45}$$

$$\lambda = \sqrt{v^2 + c}. \tag{2.46}$$

The magnitude factor is provided by Ray Vandenhoeck's article [38],

$$\nu_0 = 0.27\delta(M - 1)/M. \tag{2.47}$$

Here, δ represents the shock thickness, and M is the shock strength. These parameters vary depending on the specific test case used.

Newtonian-based method   This approach uses three artificial quantities: shear viscosity μ, bulk viscosity β, and thermal conductivity κ. This artificial viscosity injection technique does not introduce an additional contribution to the Navier-Stokes equations like Laplacian viscosity does. Instead, it directly adds the artificial viscosities to the physical viscosities which contribute to the diffusive flux $\vec{d}$ as shown in Eq. 2.3. For clarity, physical viscosities will be denoted with a subscript $_f$, while artificial ones will be denoted with an asterisk *.

$$\mu^* = k_\beta \; g(h, p) \; \rho \; \sqrt{v^2 + c^{*2}} \; 0.5 \times \left[1 + \sin \left((\hat{s}_\mu - 0.5)\pi \right) \right], \tag{2.48a}$$

$$\beta^* = k_\mu \; g(h, p) \; \rho \; \sqrt{v^2 + c^{*2}} \; 0.5 \times \left[1 + \sin \left((\hat{s}_\beta - 0.5)\pi \right) \right], \tag{2.48b}$$

$$\kappa^* = C_p \; \mu^*/\tilde{Pr}. \tag{2.48c}$$

k_μ and k_β are multiplicative constants whose value is determined based on the sensor used. $\tilde{Pr}$ is a modified Prandtl number that takes into account only artificial viscosity. Assuming that the dynamic and thermal thicknesses are of the same order of magnitude leads to setting this number to 1.

The approach used in this strategy makes the thermal sensor obsolete. This issue arises from the difference in detection between shear and thermal sensors, which can lead to the introduction of shear viscosity without thermal conductivity, and thus without heat flow. This situation can cause the simulation to crash.

The speed of sound can undergo variations through shocks. That is why it is preferable to use the critical speed of sound c^*,

$$c^* = \sqrt{\gamma R \left(\frac{2}{\gamma + 1}\right) T_0}, \tag{2.49}$$

which depends on the total temperature T_0 remaining constant through a stationary shock.

Geometric Scale   The amount of artificial viscosity added depends on the dilation and the state of the fluid, but it must also be influenced by the mesh resolution. In other words, it is entirely logical for artificial viscosity to adjust based on the refinement of the mesh and tend toward zero as the mesh becomes infinitely fine. Thus, to establish this relationship between artificial viscosity and mesh resolution, it is necessary to define a characteristic length. This characteristic length also affects physical sensors, allowing these physical quantities to be brought to a more appropriate scale.

The choice of this scale depends on the need to capture shocks within the mesh elements themselves. In lower-order schemes, this length scale is on the order of the element size, typically denoted as h. However, for higher-order methods, such as those described by Persson and Peraire [35], this length scale becomes on the order of h/p, where p represents the order of the polynomial used to approximate the solution. Generally, the value of h is not constant across the entire mesh but varies from one element to another. However, since simulations are conducted on a structured mesh, a constant characteristic size can be used. Three characteristic sizes have been defined for higher orders to better study their impact on artificial viscosity, named respectively *Linear-*, *Linear+* and *Constant*.

$$g(h, p) = \begin{cases} h \times (2 - Max(d)) & \text{[Linear -]} \\ h \times Max(d) & \text{[Linear +]} \\ h/p & \text{[Constant]} \end{cases} \tag{2.50}$$

where d represents the distance between two interpolation points, p denotes the order of the polynomial utilized, and h represents the characteristic size computed as

$$h = Max(V_e/A_e) \tag{2.51}$$

with V_e and A_e, respectively the volume and surface area of the element.

4.3   Interpolation method

This section aims to explain the interpolation method employed to interpolate artificial viscosity defined at a specific set of points to the volume and face quadrature points used in various simulation calculations, such as flux computation, residuals, and other parameters. The interpolation method relies on the basis of Bernstein polynomials, chosen to maintain positive artificial viscosity throughout the element and avoid the creation of new extremes. Using Lagrange basis functions
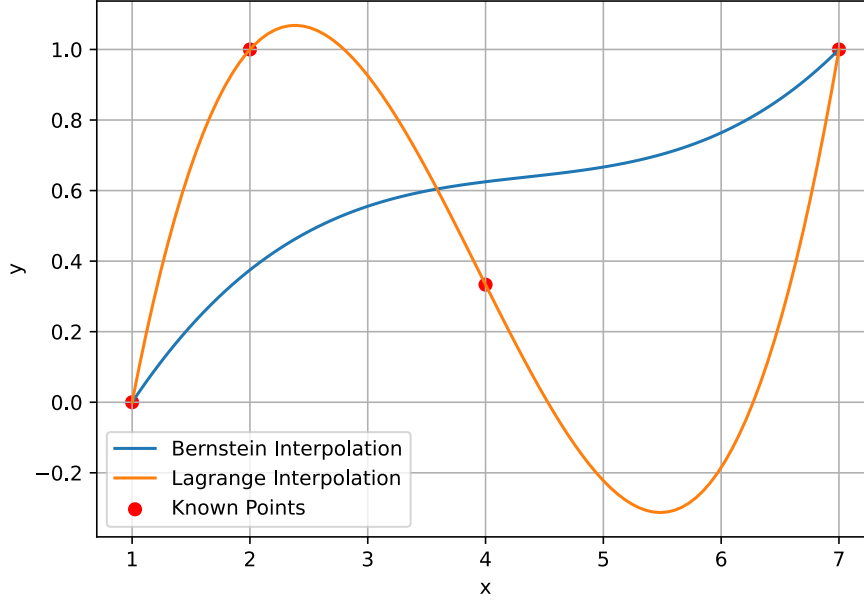
Figure 2.3: Lagrange and Bernstein interpolation

for interpolation could lead to the occurrence of negative viscosity values (see Fig. 2.3), compromising the validity of the results. By opting for Bernstein polynomials, not only is the positivity of quantities ensured, but also the avoidance of creating extremes. This latter property is particularly beneficial when calculating the spectral radius of diffusion, the methodology of which was detailed previously, and inevitably involves the artificial viscosity.

The use of Lagrange interpolation does not guarantee that the calculated artificial viscosity is maximal, which is crucial for its use in the spectral radius to compute the penalty factor. In contrast, Bernstein interpolation, while not interpolating artificial viscosity as rigorously as Lagrange polynomials, offers the advantage of knowing that the maximum amount of artificial viscosity is already defined at one of the points in the set; therefore, the quadrature points will have lower artificial viscosity. It is important to emphasize that the calculation of artificial viscosity remains an approximation, and as such, the use of Bernstein polynomials is considered within the context of an approximation, aiming to obtain results that will be critically evaluated later on.

The definition of the Bernstein polynomials basis is crucial. It is represented in matrix form where each row corresponds to points, and each column represents the associated basis function, as depicted in the following matrix:

$$M_B = \begin{bmatrix} \mathcal{B}_0^{(n)}(x_0^i)\,\mathcal{B}_0^{(n)}(x_0^j) & \mathcal{B}_0^{(n)}(x_0^i)\,\mathcal{B}_1^{(n)}(x_0^j) & \mathcal{B}_0^{(n)}(x_0^i)\,\mathcal{B}_2^{(n)}(x_0^j) & \dots & \mathcal{B}_{n-1}^{(n)}(x_0^i)\,\mathcal{B}_{n-1}^{(n)}(x_0^j) \\ \mathcal{B}_0^{(n)}(x_1^i)\,\mathcal{B}_0^{(n)}(x_1^j) & \mathcal{B}_0^{(n)}(x_1^i)\,\mathcal{B}_1^{(n)}(x_1^j) & \mathcal{B}_0^{(n)}(x_1^i)\,\mathcal{B}_2^{(n)}(x_1^j) & \dots & \mathcal{B}_{n-1}^{(n)}(x_1^i)\,\mathcal{B}_{n-1}^{(n)}(x_1^j) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathcal{B}_0^{(n)}(x_{n-1}^i)\,\mathcal{B}_0^{(n)}(x_{n-1}^j) & \mathcal{B}_0^{(n)}(x_{n-1}^i)\,\mathcal{B}_1^{(n)}(x_{n-1}^j) & \mathcal{B}_0^{(n)}(x_{n-1}^i)\,\mathcal{B}_2^{(n)}(x_{n-1}^j) & \dots & \mathcal{B}_{n-1}^{(n)}(x_{n-1}^i)\,\mathcal{B}_{n-1}^{(n)}(x_{n-1}^j) \end{bmatrix} \quad (2.52)$$

where n is the order used, x^i and x^j respectively represent the x and y coordinates in the cell. The 1D polynomial basis is described as:

$$\mathcal{B}_k^{(n)}(x^i) = \left[2^{-n} \binom{n}{k} (1 + x^i)^k \, (1 - x^i)^{n-k}, \;\; k = 0, ..., n \in \mathbb{N} \right]. \quad (2.53)$$

The artificial viscosity at quadrature points, M_{AV}, is calculated in a matrix form as follows:

$$M_{AV} = M_B \cdot M_{AV,k}. \tag{2.54}$$

The matrices M_{AV} and $M_{AV,k}$ respectively represent the artificial viscosity matrix at quadrature points and control points, where each row corresponds to the points, and the columns represent the different cells. M_B is computed at quadrature points.

5   Common Physical Quantities

This section aims to outline the various quantities and concepts that will be utilized to analyze and compare the results.

5.1   Kinetic energy and enstrophy

The kinetic energy and enstrophy will be essential in assessing how various artificial viscosity methods dissipate these quantities. To achieve this, these quantities will be integrated over the volume and made dimensionless by normalizing them with their initial value. It is important to note, and worth reminding, that this value does not depend on the use of a shock capture method and remains consistent regardless of the sensor or method employed.

The kinetic energy is calculated as follows:

$$K = \frac{1}{K_0} \int_V \frac{1}{2} \rho \, \vec{v} \cdot \vec{v} \, dV, \tag{2.55}$$

where K_0 is the initial kinetic energy.

Enstrophy is a measure of the intensity of vortices in a fluid flow. It is defined as

$$\xi = \frac{1}{\xi_0} \int_V \frac{1}{2} \rho \, \vec{\omega} \cdot \vec{\omega} \, dV, \tag{2.56}$$

where ξ_0 is the initial enstrophy and $\vec{\omega}$ the vorticity.

5.2   Numerical schlieren

The schlieren is a method used to visualize density variations within a fluid. When applied to a strong vortex, this technique reveals density changes associated with the rapid rotation of the fluid. Within the vortex, schlieren highlights compression zones, where density increases due to rotation, and rarefaction zones, where density decreases. Although schlieren does not directly provide information about fluid velocity, it offers a visual representation of the fluid properties within the vortex. In the context of shock capturing, schlieren also plays a crucial role in accentuating spurious oscillations near the shock.
The Schlieren (Sch) is calculated using

$$Sch = \frac{ln\left(1 + ||\vec{\nabla}\rho||\right)}{ln(10)}, \tag{2.57}$$

where $\vec{\nabla}\rho$ represents the gradient of density. This logarithmic expression quantifies local density variations in a fluid, providing a precise visual representation of density gradients.

5.3   Kinetic energy budget

The kinetic energy budget is a fundamental concept in fluid dynamics. It involves studying how kinetic energy is generated, distributed, transformed, and dissipated within a system. This budget is crucial for understanding fluid motion and the complex interactions governing fluid systems. Starting from the Navier-Stokes momentum equation Eq.2.1, taking the dot product with $\vec{v}$ leads to a first form of the kinetic energy conservation equation:

$$\vec{v} \cdot \frac{\partial \rho \vec{v}}{\partial t} = -\vec{v} \cdot \vec{\nabla} p - \vec{v} \cdot \vec{\nabla} \cdot \left(\rho \vec{v} \vec{v} \right) + \vec{v} \cdot \vec{\nabla} \cdot \vec{\tau}. \tag{2.58}$$

By rearranging this form slightly, another form emerges:

$$\frac{\partial k}{\partial t} = \underbrace{p \, \vec{\nabla} \cdot \vec{v}}_{\text{Pressure work}} - \underbrace{\vec{\tau} : \vec{\nabla} \vec{v}}_{\text{Viscous dissipation}} + \underbrace{\vec{\nabla} \cdot \left(-p\vec{v} + k\vec{v} + \vec{\tau} \cdot \vec{v} \right)}_{\text{Transport terms}}. \tag{2.59}$$

where $k = \rho |\vec{v}|^2 / 2$ represents kinetic energy density.

The temporal variation of the kinetic energy density is calculated using a second-order upwind finite difference method [39],

$$\frac{\partial k}{\partial t} = \frac{3k^n - 4k^{n-1} + k^{n-2}}{2\Delta t}, \tag{2.60}$$

where k^n represents the kinetic energy density at time t, k^{n-1} is the kinetic energy density at time $t - \Delta t$, and k^{n-2} is the kinetic energy density at time $t - 2\Delta t$.

For an effective comparison of the budget, all quantities involved will be made dimensionless. Since they all have units of energy dissipation, the dimensionless scaling will be done using the quantity α, defined as

$$\alpha = K_0 / \tau \tag{2.61}$$

where K_0 is the initial kinetic energy and τ a characteristic time.
Although it is always calculated as a distance divided by a speed $\tau = d/v$, these quantities vary depending on the test conducted.
For Kelvin-Helmholtz instabilities, d is the width of the shear layer and v is the maximum initial speed.
In the case of strong vortex-shock wave interaction, d represents the radius b of the vortex, and v is the background velocity v_m.

Chapter 3

Instabilities of Kelvin-Helmholtz

The Kelvin-Helmholtz Instability (KHI) is a phenomenon that occurs when two adjacent fluids move at different speeds relative to their common interface. This instability can also manifest in the presence of velocity shear within a single continuous fluid. The KHI test case garners considerable interest due to its frequent observation in various natural contexts, including oceanic geophysics, hydrodynamics, magnetohydrodynamics, and astrophysics.

This offers a unique opportunity to study the transition to turbulence from laminar flows due to the emergence of ripples prone to secondary instabilities. Understanding the physics related to the resultant turbulent dissipation and mixing is crucial, especially for computational simulations [40].

In the development of shock-capturing methods, this test case allows for exploring the method's response, particularly in terms of artificial viscosity, when the simulation does not conclude due to a shock but due to excessive shear in the layer.

Interestingly, the KHI test case can be studied both viscously and inviscidly. In the former case, where viscosity is present, the problem can be completely resolved thanks to the emergence of the Kolmogorov scale. However, in the inviscid case, simulation crashes can occur due to the emergence of under-resolved structures. The latter case will be dealt with here.

The first part of the chapter aims to present the problem, then to study the two processes of artificial viscosity injections separately, and finally, to compare these two methods of injection.

1   Problem description

Kelvin-Helmholtz instabilities can manifest in various contexts, but here, the test case is studied in an environment where the moving fluids exhibit stratification. In simple terms, stratification implies the presence of distinct layers of fluids with varying properties, such as density or temperature.

The stratified Kelvin-Helmholtz instability test case thus represents an approach to numerically study two layers of fluids with different properties moving at distinct speeds, thereby creating an interface between them. This analysis enables an understanding of the specific characteristics and behaviors of Kelvin-Helmholtz instability in a stratified context. This can have implications in a wide field of applications, involving meteorology and oceanography [41].

In practical applications, it is common to use a double shear layer to simulate the formation of Kelvin-Helmholtz instabilities in a two-dimensional numerical environment. This scenario of

instability within a shear layer is typically selected to demonstrate the transformation of linear disturbances into a transition to nonlinear two-dimensional hydrodynamic turbulence. Initially, the instability generates small-scale vortical structures at the high-density interface, which progressively evolve, through nonlinear interactions, into a complete turbulent field.

The simulations were conducted using a Lax-Friedrichs flux, Gauss-Lobatto-Legendre shape functions, and a Linear- scaling factor. The time step used was equal to $5 \; 10^{-5}$ [s]. The simulation domain is a square defined by $(x, y) \in [0, 2] \times [0, 2]$. The implementation of the initial conditions for the studied case is provided by

$$
\begin{cases}
B & = \tanh\left(15y + 7.5\right) - \tanh\left(15y - 7.5\right), \\
\rho(x, y) & = 0.5 + 3B/4, \\
\rho(x, y)v_x(x, y) & = 0.5(B - 1), \\
\rho(x, y)v_y(x, y) & = 0.1 \sin\left(2\pi x\right), \\
k & = 0.5 \; \rho\left(v_x^2 + v_y^2\right), \\
\rho(x, y)E(x, y) & = \frac{1}{\gamma - 1} + k.
\end{cases}
\tag{3.1}
$$

The initial conditions in a the simulation domain are depicted in Fig. 3.1 depicts the initial conditions in the simulation domain. These figures illustrate the dual symmetry of the test case. The profile of the x-component of velocity clearly demonstrates the generation of shear layers located at $y = 0.5$ and $y = 1.5$. The density and energy density profiles reveal the stratification phenomenon.

Sensors and injection methods are heavily influenced by various user parameters, which may be similar across multiple sections. To avoid excessive repetition. To limit redundancy, a brief description of the parameters related to sensors and injection methods is provided below. If the parameter values vary for a specific test, they will be explicitly stated in the corresponding section.

Sensors

- The Persson Peraire sensor relies on Lagrangian filtering with a detection threshold $s_0 = -3 \log_{10}(\text{order})$ and an interval $\kappa = 1$. The choice of the detection threshold was made in accordance with the guidelines outlined in the referenced article [42].

- The Hennemann sensor does not use any user parameters.

- For the physical sensors (shear, shock and thermal), the detection threshold are defined as follows: $s_{\min,\mu} = 1.2$ and $s_{\min,\beta} = 0.01$, with a maximum value $s_{\max} = 2$ for each of the sensors.

Injection Method

- Laplacian method uses a shock thickness $\delta = 1$ and a shock strength $M = 10$ (see Eq. 2.47).

- Newtonian method employs the multiplicative constants $k_{\mu,\beta} = 0.02$ (see Eq. 2.48).
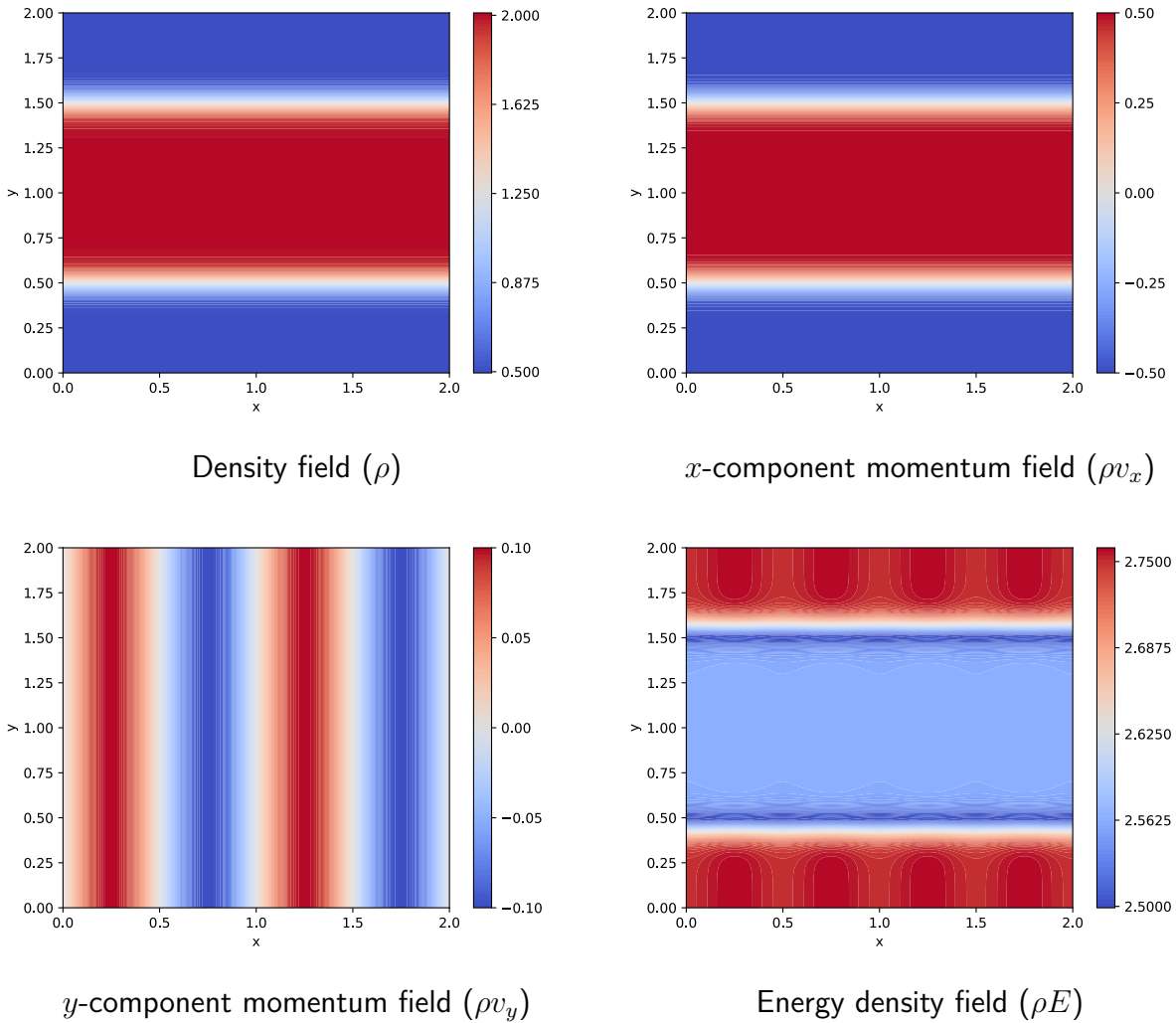
Figure 3.1: Initial condition for the stratified Kelvin-Helmholtz test case, incorporating a double shear layer.

2   Laplacian Injection

The use of the Laplacian introduces additional diffusion, leading to increased dissipation that enhances stability, but at the cost of some information loss. This section aims to examine and compare Laplacian injection with various sensors to determine which one is likely to yield optimal results while ensuring adequate stability.

Study Structure

For this purpose, the section will be divided into two parts: A visual analysis will be given, allowing observation of the behavior of each element in the simulation at a given time t. A quantitative analysis of different quantities, illustrating the impact of the method on the simulation.

This section will focus solely on the regular sensors (PerssonPeraire and Hennemann sensors). The exclusion of the physical sensor is due to its inability to stabilize the simulation when combined with the Laplacian injection and the previously described user parameters.

2.1   Visual analysis

This section, focuses on the spatial aspect with the aim of verifying if the results align with expectations and presenting them. To this end, two time instants have been selected. The first corresponds to a time $t = 2.6[s]$, that allows confirming if the correct physical phenomenon is indeed taking place as viscosity evolves within the solution. At this moment, the solution begins to generate its initial structures, coinciding with the introduction of artificial viscosity shortly before, enabling us to observe its initial effects. The second chosen instant corresponds to the end of the simulation, at $t = 8[s]$, providing an opportunity to visualize a solution stabilized by artificial viscosity. At this point, the impact of the method on the solution becomes more apparent.

Analysis at t = 2.6 [s]

In order to better understand the results that will be discussed, it is necessary to provide some explanations about what can be expected from this test case. The Kelvin-Helmholtz instability arises from turbulence between two adjacent layers of air moving at different velocities. Similarly, the friction between these two masses of air can generate irregularities, manifesting as the penetration of one layer into the other, observed in the form of waves. Due to the continuity of the airflow, air elements near the boundary move faster than those farther away. The pressure next to the layer of faster-moving air is lower than in the surrounding environment. Thus, a force pulls the formation (wave) in the direction of the faster-moving airflow. Near the trough of a wave, the air flows more slowly than in the surrounding area, resulting in higher local pressure. This area experiences a force that pushes it in the opposite direction [43]. The schematic illustration in Fig. 3.2 illustrates the concepts described above.
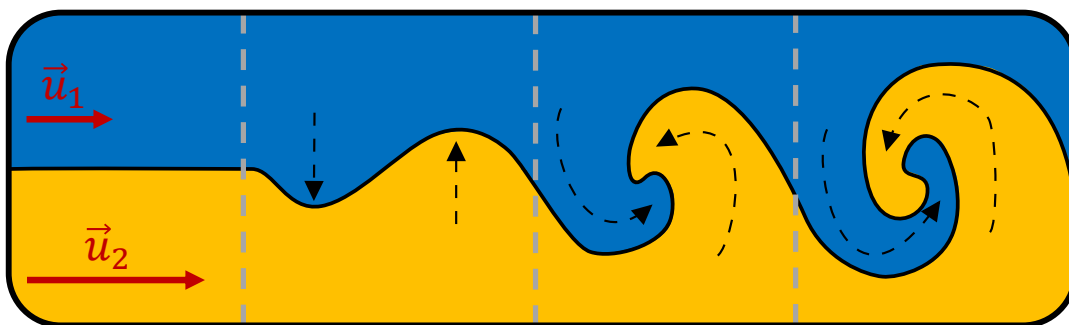
Figure 3.2: Schematic depiction of the generation of Kelvin-Helmholtz instabilities.

Density fields

A similar phenomenon to the one explained earlier is evident in the simulation at time $t = 2.6[s]$ (Fig. 3.3). This figure displays the density obtained for different sensors and configurations. For clarity, it also shows some structures present in either sensor. Initially, the impact of lower viscosity on the solutions from the Hennemann sensor can be observed. Small structures emerge from the vortices that are only noticeable with this sensor (Points 2 and 3).

Artificial viscosity

When examining the artificial viscosity in Fig. 3.4, it is noticeable that artificial viscosity is significantly more pronounced and intense in the PerssonPeraire sensor. However, for each sensor, the most disturbed areas are those within the shear zone. The figure demonstrates that artificial viscosity manages to mimic the deformation of the shear layer, suggesting that artificial viscosity is injected at the right locations throughout the simulation.

Analysis at t = 8 [s]

Density fields

Fig. 3.5 presents the final density fields of the simulation.
Visually, the Hennemann sensor outperforms the PerssonPeraire sensor in terms of results. The former tends to reduce the amount of artificial viscosity injected, as previously observed, thereby minimizing its impact on the simulation.
The Hennemann sensor also shows a significant variation when using similar degrees of freedom but with different orders of interpolation. This suggests that the simulation is quickly affected by the addition of artificial viscosity. It appears that a mesh of the same resolution but of a higher order creates less dissipation, leading to a more realistic result. However, this test case exhibits such pronounced visual variations that it is difficult to determine to what extent a higher order is more advantageous in terms of accuracy. Therefore, we will have to wait for a more quantitative analysis to draw real conclusions.

As for the PerssonPeraire sensor, it does not show significant changes according to the configuration, indicating an excessive addition of artificial viscosity, which ultimately negates the increased precision obtained by increasing the order, despite a constant degree of freedom.

The symmetry in the test case is broken for all configurations, except for the one using the PerssonPeraire sensor with an order 3 scheme. This breaking of symmetry could be due to the asymmetric detection of disturbed cells by the sensors, leading to unequal injection of artificial viscosity. The reason for the preservation of symmetry with the PerssonPeraire sensor at an order 3 scheme would be similar to its consistency between mesh resolutions: excessive numerical dissipation erasing most small structures.

(a) 64×64 - Order 3 (PerssonPeraire sensor)

(b) 51×51 - Order 4 (PerssonPeraire sensor)

(c) 64×64 - Order 3 (Hennemann sensor)

(d) 51×51 - Order 4 (Hennemann sensor)
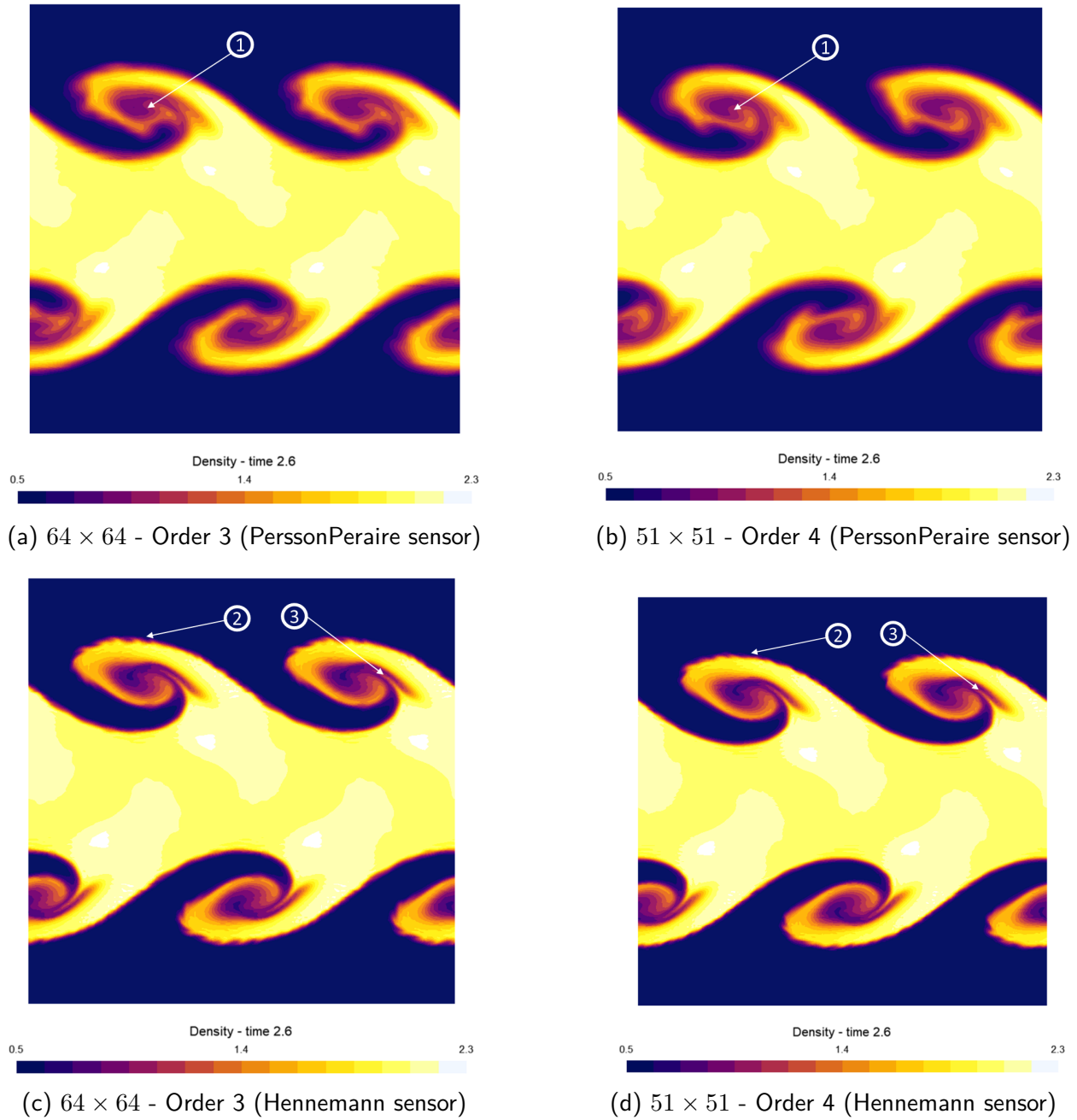
Figure 3.3: Density fields at time t = 2.6 [s] obtained through an artificial viscosity method with Laplacian injection for the PerssonPeraire sensor and Hennemann sensor. The results are presented with grid configurations of 64×64 using a third-order scheme and 51×51 using fourth-order scheme.

(a) 64×64 - Order 3 (PerssonPeraire sensor)

(b) 51×51 - Order 4 (PerssonPeraire sensor)

(c) 64×64 - Order 3 (Hennemann sensor)

(d) 51×51 - Order 4 (Hennemann sensor)
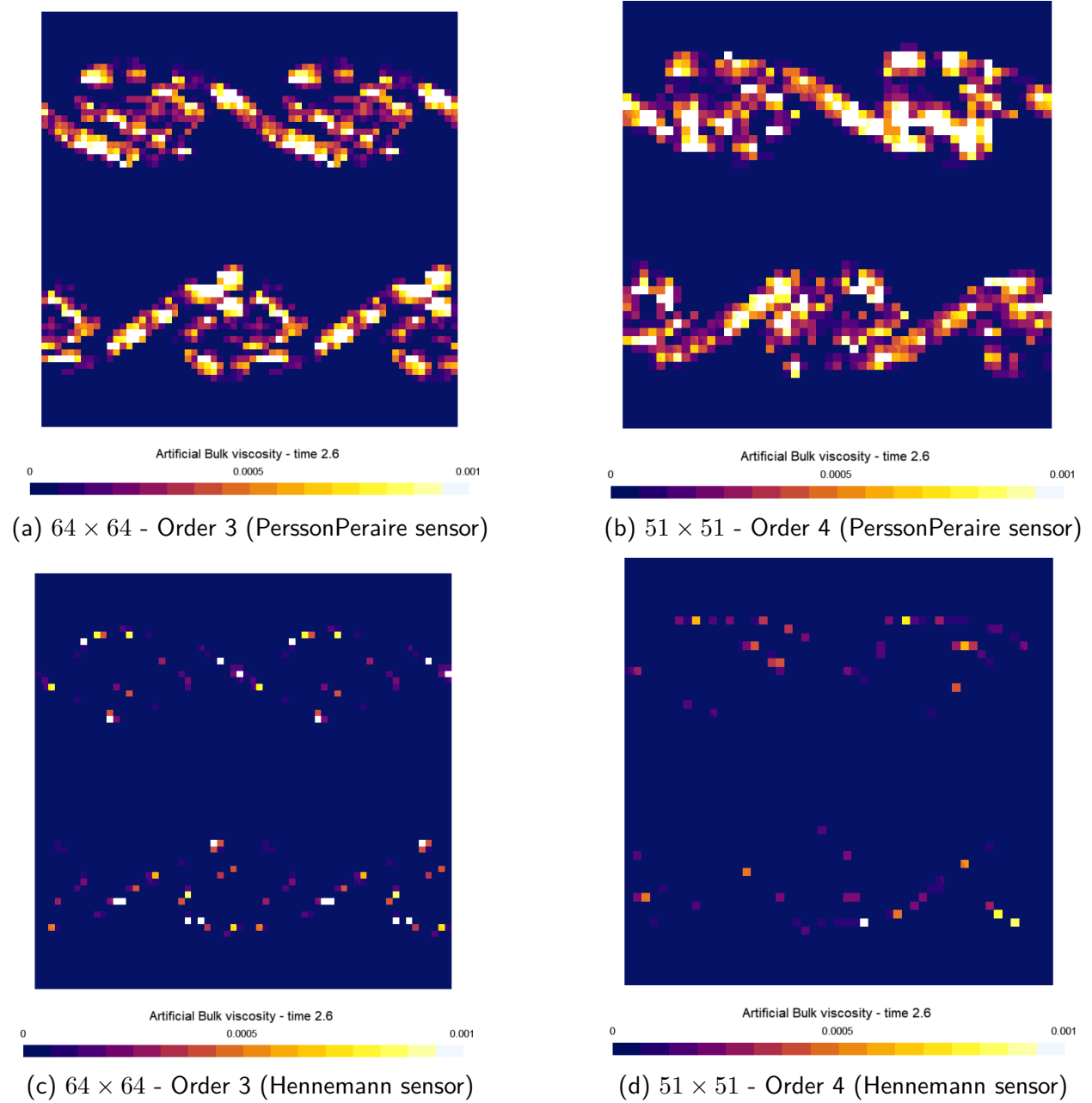
Figure 3.4: Artificial viscosity fields at time t = 2.6 [s] obtained through an artificial viscosity method with Laplacian injection for the PerssonPeraire sensor and Hennemann sensor. The results are presented with grid configurations of 64×64 using a third-order scheme and 51×51 using fourth-order scheme.

(a) 64×64 - Order 3 (PerssonPeraire sensor)

(b) 51×51 - Order 4 (PerssonPeraire sensor)

(c) 64×64 - Order 3 (Hennemann sensor)

(d) 51×51 - Order 4 (Hennemann sensor)
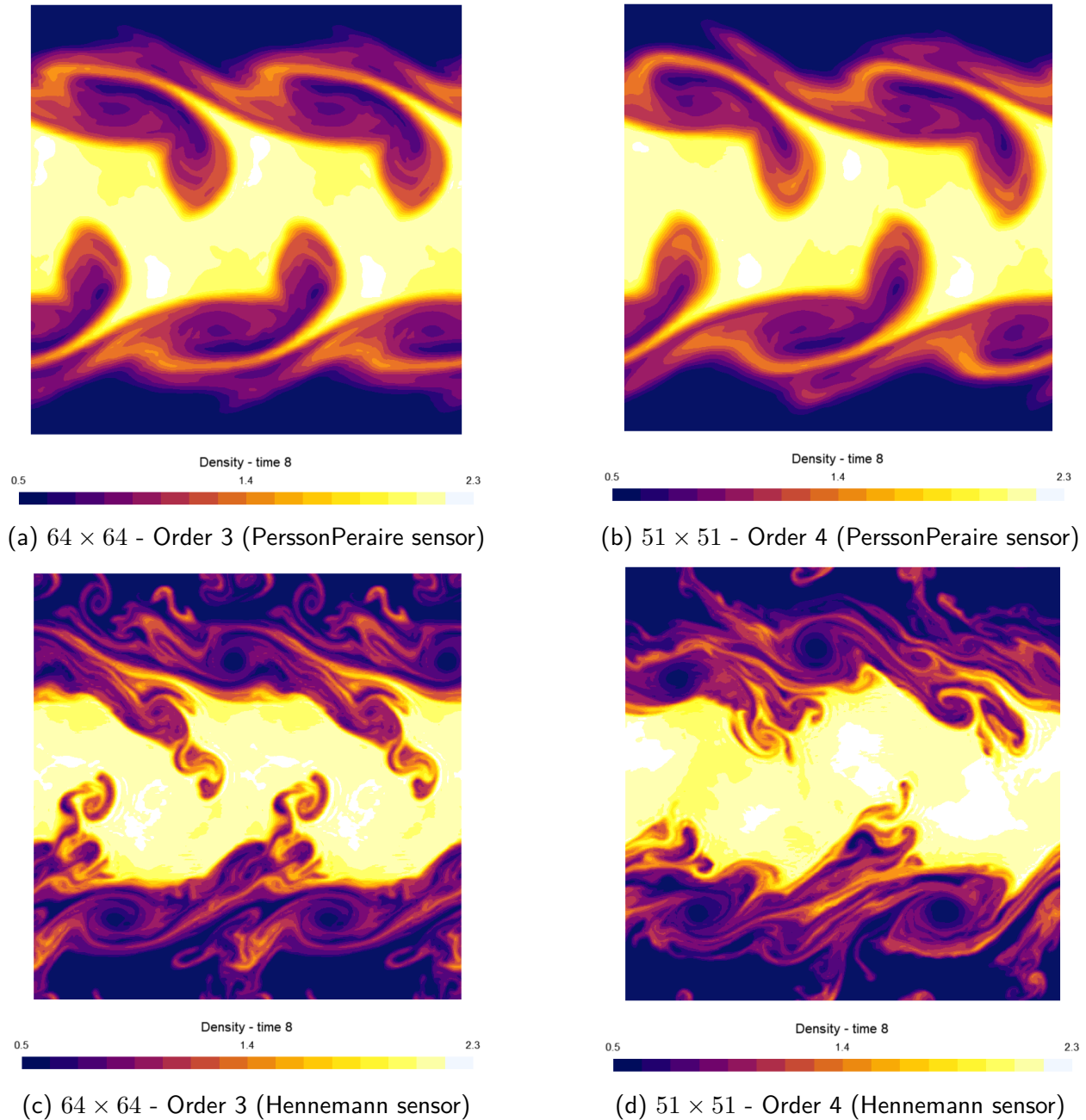
Figure 3.5: Density fields at time t = 8 [s] obtained through an artificial viscosity method with Laplacian injection for the PerssonPeraire sensor and Hennemann sensor. The results are presented with grid configurations of 64×64 using a third-order scheme and 51×51 using fourth-order scheme.

Summary of visual analysis

The artificial viscosity using the Laplacian method, paired with the Hennemann sensor, effectively meets expectations by correctly generating Kelvin-Helmholtz instabilities. The Hennemann sensor seems to offer better performance. These improvements stem from the sensor's greater accuracy, detecting fewer disturbed areas and injecting less artificial viscosity, both in quantity and intensity. This reduced detection leads to less numerical dissipation, allowing the simulation to reproduce more small physical phenomena.

Additionally, it appears that with an identical mesh resolution, a higher-order simulation is more precise when using the Hennemann sensor. In contrast, the PerssonPeraire sensor, with its overly high detection, tends to eliminate many small phenomena due to excessive artificial viscosity.

Finally, a symmetry break was observed, likely due to an asymmetric injection of artificial viscosity that disrupts the symmetrical balance.

In conclusion, the results obtained with the Laplacian artificial viscosity injection method, in combination with the Hennemann sensor, are less dissipative than those obtained with the PerssonPeraire sensor in the used configuration.

2.2 Quantitative analysis

Artificial viscosity

Fig. 3.6 shows the evolution of artificial viscosity integrated over the volume. Initially, no artificial viscosity is applied. It is around the first second that it starts to act for the PerssonPeraire sensor, increasing until a peak at about 3 seconds. Without this viscosity, the simulation would have failed (as indicated by the Vanilla DG curve without a shock capturing method). This increase serves to regulate the solution. For this sensor, although the number of degrees of freedom is the same, order 3 requires less artificial viscosity than order 4, possibly due to greater instability at a higher order.

The Hennemann sensor detects disturbances later, injecting artificial viscosity more gradually, reaching a plateau different from that of the PerssonPeraire sensor. The viscosity applied by this sensor is similar for different orders, although at order 3, the curve is slightly higher than that of order 4. These observations indicate that the detection method significantly influences the behavior of artificial viscosity. The PerssonPeraire sensor suggests that order 4 is more unstable than order 3, while the results from the Hennemann sensor indicate the opposite.

Furthermore, it is noted that the Hennemann sensor generally requires less viscosity than PerssonPeraire. This observation is independent of the mesh resolution used in the simulation. The data show that the artificial viscosity generated by the Hennemann sensor is at least half as much as that generated by the PerssonPeraire sensor. This significant difference highlights the different efficiency and sensitivity of the two sensors in managing artificial viscosity within the simulation.

Kinetic energy

Fig. 3.7 shows the integration of dimensionless kinetic energy over the domain. This measure allows for a visual and quantitative assessment of the impact of artificial viscosity.

Firstly, it is important to note that the Hennemann sensor produces more favorable curves, which is not surprising. It closely follows the Vanilla DG curve for an extended period before diverging
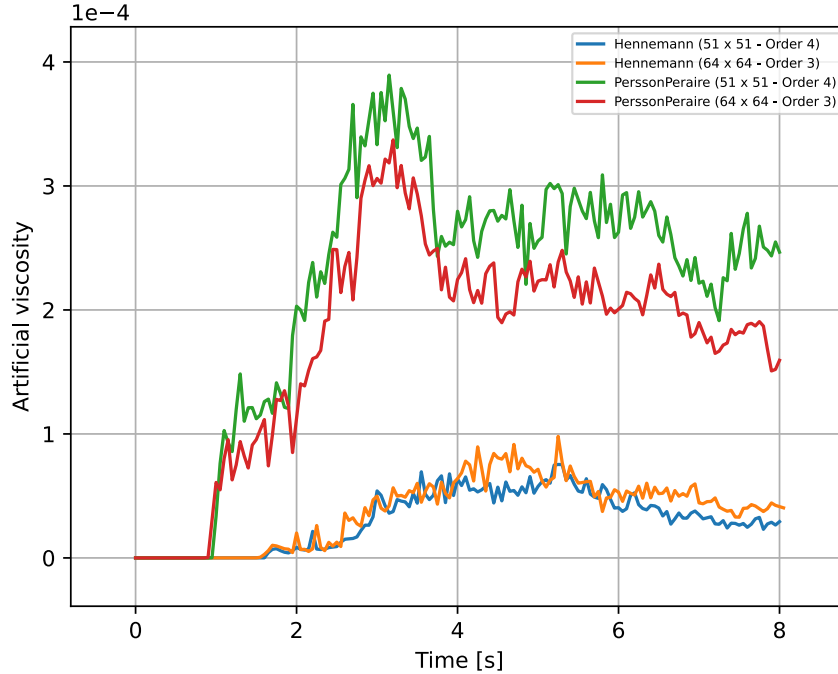
Figure 3.6: Artificial viscosity integrated over the volume obtained through an artificial viscosity method with Laplacian injection for the PerssonPeraire sensor and Hennemann sensor. The results are presented with grid configurations of 64×64 using a third-order scheme and 51×51 using fourth-order scheme.

due to numerical dissipation. However, this numerical dissipation contributes to the stabilization of the system, as clearly demonstrated by this graph.

On closer examination of this graph, it is observed that the dissipation induced by the different sensors depends on the amount of artificial viscosity injected by these sensors. The higher the artificial viscosity, the greater the dissipation, leading to a rapid decrease in kinetic energy.

Moreover, it is interesting to note that this graph reflects the same constant as observed in the artificial viscosity graph. This is understandable considering the relationship between the intensity of artificial viscosity and numerical dissipation.

Kinetic energy budget

After observing and analyzing various quantities, it is interesting to focus on the budget, as presented in Fig. 3.8. It is firstly important to note a striking similarity with the artificial viscosity graph, although it appears to be inverted. This highlights the impact of this viscosity on the simulation.

At the beginning of the simulation, the budgets are almost null until artificial viscosity is injected. Following this injection, a notable divergence in the budget is observed for both types of sensors. This divergence reaches a peak when the simulation becomes most unstable, then gradually attempts to return to zero.
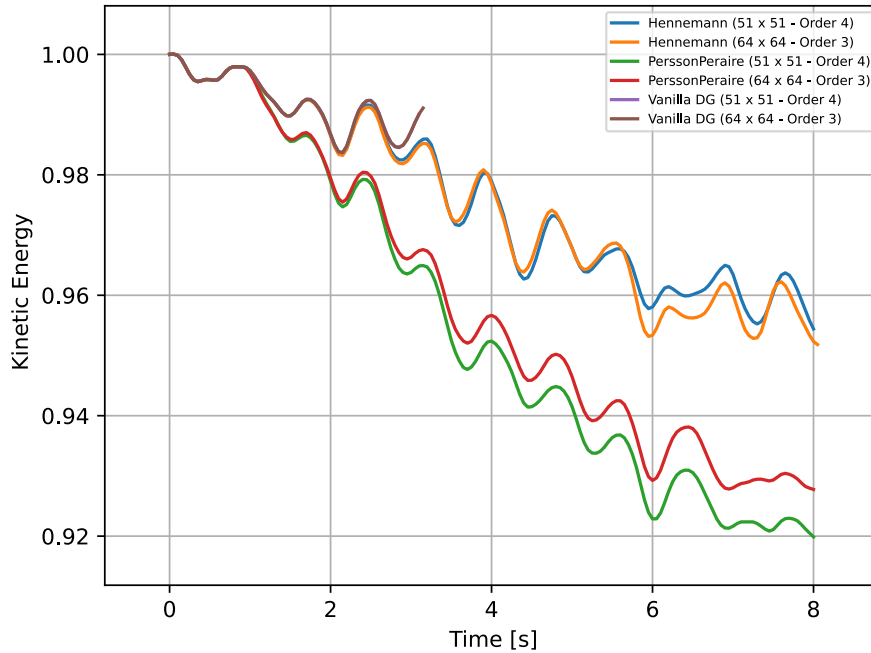
Figure 3.7: Dimensionless kinetic energy integrated over the volume obtained through an artificial viscosity method with Laplacian injection for the PerssonPeraire sensor and Hennemann sensor. The results are presented with grid configurations of 64×64 using a third-order scheme and 51×51 using fourth-order scheme.

2.3   Conclusion

Sensors detect a problem related to the shear layer and consequently inject artificial viscosity. The Hennemann sensor, adapting better, detects and injects a more precise artificial viscosity, avoiding excess. Thus, the final figure from this sensor shows more small structures, unlike its counterpart which, due to excessive detection, creates too much numerical dissipation in the density field and other quantities like kinetic energy. The asymmetric injection of artificial viscosity breaks the symmetry of the test case. Adding artificial viscosity causes a budget imbalance, but the method gradually adapts to rebalance the budget.

The use of artificial viscosity with Laplacian injection varies depending on the sensor used. For example, the PerssonPeraire sensor tends to inject more artificial viscosity for a 4th order calculation compared to a 3rd order at the same resolution. In contrast, the Hennemann sensor reduces the intensity of artificial viscosity for a 4th order compared to a 3rd order, also at the same resolution. This difference in behavior illustrates how each sensor adapts to variations in calculation order to maintain balance and efficiency in the process.

Given the chosen configuration for the sensors and the injection method used, the Hennemann sensor seems to be superior, especially with fewer user parameters. However, this does not mean that the PerssonPeraire sensor is inferior; with different parameters, it could prove to be more suitable depending on the situation.
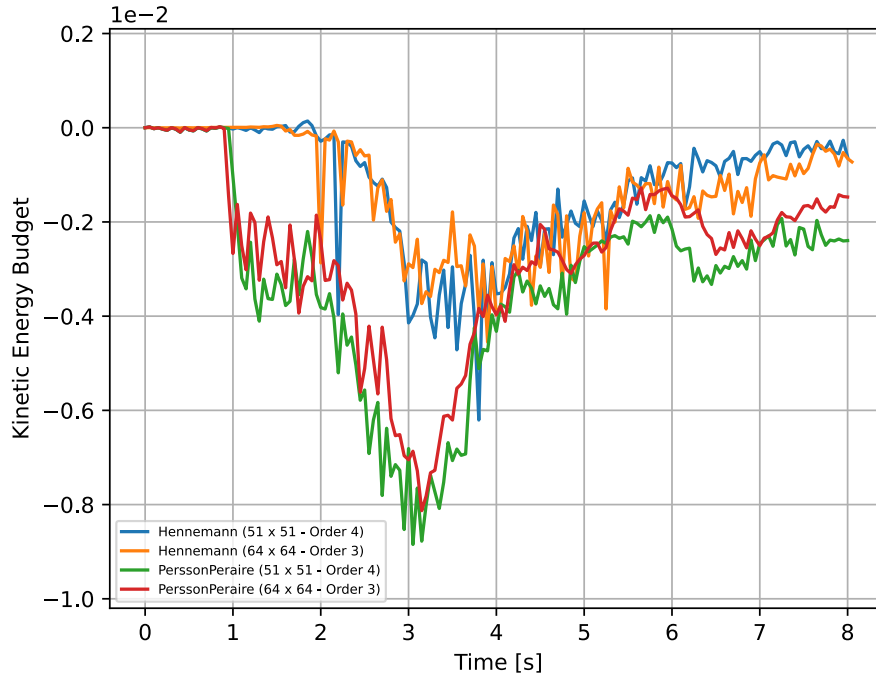
Figure 3.8: Kinetic energy budget obtained through an artificial viscosity method with Laplacian injection for the PerssonPeraire sensor and Hennemann sensor. The results are presented with grid configurations of 64×64 using a third-order scheme and 51×51 using fourth-order scheme.

3   Newtonian Injection

The previous chapter examined a method of artificial viscosity based on the application of a Laplacian. This approach, known for its robustness, has the drawback of increasing dissipation. The current intention is to explore another technique for introducing viscosity, based on physical principles. Unlike the previous method, this one does not add terms to the equations but merges the calculated artificial viscosity with the existing physical viscosity to create a total viscosity. Since the case studied is non-viscous, the total viscosity will correspond to the artificial viscosity.

This method requires the use of three specific variables: shear viscosity (μ), bulk viscosity (β), and thermal conductivity (κ). Three distinct sensors would be ideal for measuring these variables separately. However, in the context of this research, two sensors are sufficient due to the relationship between artificial shear viscosity and artificial thermal conductivity (see Eq. 2.48).

The Newtonian injection method is limited to the use of physical sensors for this case study, as the regular sensors associated with this method did not achieve a stable solution with the user parameters described earlier.

The use of physical sensors is thus crucial for maximizing the efficiency of the Newtonian injection method. These sensors operate on a point-by-point basis and can be calculated in various ways.

Study Structure

The study is structured into three subsections. The first addresses different strategies for evaluating artificial viscosity values at quadrature points. The second subsection examines the influence of changing the detection threshold for the shear sensor, essential for the proper functioning of this case study. The third and last part discusses the potentially problematic relationship between the geometric factor and the detection threshold.

The simulations in this section were carried out on two grids of different dimensions: one of 64×64 with an order 3 scheme (65.536 Dofs) and another of 51×51 order 4 (65.025 Dofs), allowing for the analysis of the effect of increasing the order while maintaining a nearly identical number of degrees of freedom in each subsection.

3.1   Comparative Analysis of Interpolation Methods

The simplest method for calculating artificial viscosity is to determine it directly at the quadrature points. However, this approach can present difficulties, particularly in handling discontinuities. To address these issues, some methods suggested in the literature include the application of smoothing filters [44] or the reconstruction of artificial viscosity [34].

In this study, in addition to the direct method, Bernstein polynomials were used. Thus, artificial viscosity is first calculated at the control points, then interpolated at the quadrature points using these polynomials. Although this method is not perfect, it is important to note that artificial viscosity approaches depend on several user parameters and that artificial viscosity is in itself an approximation. Therefore, using interpolation with Bernstein polynomials amounts to approximating an approximation.
One of the major advantages of Bernstein polynomials, compared to other types of polynomials such as Gauss-Lobatto-Legendre or Lagrange polynomials, lies in their ability not to create new maxima. This characteristic ensures that the artificial viscosity, once interpolated with these polynomials, will always remain positive, thus avoiding the generation of erroneous negative values due to interpolation. This property also ensures that the values of artificial viscosity will not exceed those calculated at the control points. In calculating the diffusive radius, which requires the maximum of artificial viscosity in the element, it is sufficient to refer to the values obtained at the control points to optimize the diffusive radius and effectively ensure coercivity (see Eq. 2.20).

The study of Fernandez [2], which served as a reference for the development of sensors and the injection method, emphasizes the need for continuity of artificial viscosity at interfaces. The effect of this continuity could present a particular interest. Therefore, in the approach using Bernstein polynomials, two scenarios will be examined: one with continuity at interfaces and the other without continuity.

In summary, this section will focus on three distinct methods: the first consists of directly calculating artificial viscosity at quadrature points, the second of determining artificial viscosity at control points and then interpolating, according to Eq. 2.54, to obtain artificial viscosity at quadrature points with continuity at interfaces. Finally, the third method follows the same process but without ensuring continuity at interfaces.

For clarity, these methods will be referred to as Direct Approach (DA), Continuous Bernstein Approach (CBA), and Non-Continuous Bernstein Approach (NCBA), respectively.

3.1.1   Visual analysis

Similar to the approach used in the earlier injection method, the same two moments, 2.6 seconds and 8 seconds, will be analyzed for the reasons already mentioned: the first moment to confirm the occurrence of the expected physical phenomenon with the evolution of artificial viscosity, and the second to observe the final impact of this viscosity on the density field.

Analysis at t = 2.6 [s]

Density

Density fields are presented in Fig. 3.9. Overall, all fields correspond to the explanations provided in the chapter on Laplacian injection at time $t = 2.6$ [s], which confirms their adherence to physical principles and indicates that artificial viscosity does not excessively affect the simulation from the beginning. It is also noteworthy that for all figures, symmetry is preserved.

Regarding the 64×64 order 3 mesh, there are only very slight variations between the Direct Approach and the Bernstein Approaches, indicating a slightly higher precision for the latter. Given the sensitivity of this test case to the injection of artificial viscosity, it could be that the different approaches inject a nearly identical amount of artificial viscosity for this mesh. To confirm this hypothesis, it will be necessary to examine the integration of shear and bulk artificial viscosity over the volume.
For the 51×51 order 4 mesh, the fields are noticeably different. It is plausible that the impact of artificial viscosity is more pronounced for this mesh resolution. The distinction between the Continuous and Non-Continuous Bernstein Approaches is a bit more evident for this resolution, though still moderate. This suggests that, from the beginning of the simulation and with a higher order, artificial viscosity varies depending on the chosen approach.

Examining the differences observed for the same approach but with two different mesh resolutions, more marked disparities are perceived with the Bernstein Approaches, continuous or not, suggesting that these methods are more affected by the order of interpolation than the Direct Approach, which seems to offer similar results regardless of the interpolation order.

Artificial viscosity

The artificial viscosity field presented is visualized at the interpolation points (equivalent to control points in this context) to illustrate the action of the artificial viscosity. The effect of interpolation on the quadrature points is not visually represented due to limitations in their accurate display. Although these figures do not exactly depict the artificial viscosity used at the quadrature points, they nevertheless reveal the modifications induced by the artificial viscosity. A supplementary quantitative analysis will allow for a better understanding of the impact of interpolation.

It is relevant to analyze the behavior of artificial viscosities, both for shear viscosity (μ^*) and for bulk viscosity (β^*). Given that thermal conductivity (κ^*) is related to shear viscosity and

(a) 64×64 - Order 3 (DA)

(b) 51×51 - Order 4 (DA)

(c) 64×64 - Order 3 (CBA)

(d) 51×51 - Order 4 (CBA)

(e) 64×64 - Order 3 (NCBA)

(f) 51×51 - Order 4 (NCBA)

Figure 3.9: Density fields at time t = 2.6 [s] obtained through an artificial viscosity method with Newtonian injection using a physical sensor. The figures have been generated for three approaches: DA, CBA, and NCBA. The results are presented with grid configurations of 64×64 using a third-order scheme and 51×51 using fourth-order scheme.

manifests with increased intensity, its behavior will be similar. Therefore, it is not necessary to spatially represent it.

Artificial shear viscosity   Fig. 3.10 illustrates the artificial shear viscosity according to the mentioned approaches and for the two selected meshes. All graphs seem consistent and show the

injection of artificial viscosity in both shear layers.

For a 51×51 resolution with an order 3 scheme, three observations emerge from these graphs. In the Direct Approach, a greater number of cells are identified as disturbed, leading to artificial viscosity injection in a larger number of cells. However, despite an increased number of disturbed cells, the intensity of artificial viscosity appears similar regardless of the approach for order 3. Additionally, the fields of artificial viscosity seem to retain symmetry at this stage of the simulation. With a 64×64 resolution and an order 4 scheme, observations tend to reverse compared to an order 3 scheme. For this mesh, detection seems similar between different approaches (although density fields showed significant changes for this resolution, preventing the assertion that detection is identical). Moreover, in these approaches, a slight symmetry break is observed. Although this break is not yet evident in the current density fields, it is expected that by the end of the simulation, these fields will also exhibit a symmetry break.

Comparing the same approach with different meshes, as observed for density fields, it appears that the Direct Approach remains similar regardless of mesh resolution. However, the Bernstein polynomial approach shows significant differences, notably an intensification of artificial viscosity. This notable increase in artificial viscosity according to order explains the divergences observed between density fields of order 3 and 4.

Artificial bulk viscosity   After examining the behavior of artificial shear viscosity, it is interesting to look at Fig. 3.11 presenting the bulk artificial viscosity.

Observations regarding this viscosity are rather succinct and general. Firstly, the figure indicates a maximum intensity of bulk artificial viscosity, which is ten times less than that of shear artificial viscosity to be perceived. Secondly, this viscosity is primarily active in the Direct Approach and almost non-existent in other approaches. Bulk artificial viscosity is calculated based on the shock sensor, which in turn is based on the divergence of velocity. The fact that it is present only in the Direct Approach suggests that velocity divergences are more pronounced in this approach than in the Bernstein approaches. This is consistent with the principle of Bernstein approaches, which calculate viscosity at quadrature points through interpolation to attenuate these discontinuities.

It is noteworthy that some points are disturbed in the Bernstein approaches at order 4, which is not the case at order 3. This could indicate two things: the first is that order 4 increases the intensity of artificial viscosity, regardless of the type of physical sensor used, whether for shear or shock. The second possibility is that order 4 is more unstable, leading to more pronounced discontinuities.

(a) 64×64 - Order 3 (DA)

(b) 51×51 - Order 4 (DA)

(c) 64×64 - Order 3 (CBA)

(d) 51×51 - Order 4 (CBA)

(e) 64×64 - Order 3 (NCBA)

(f) 51×51 - Order 4 (NCBA)
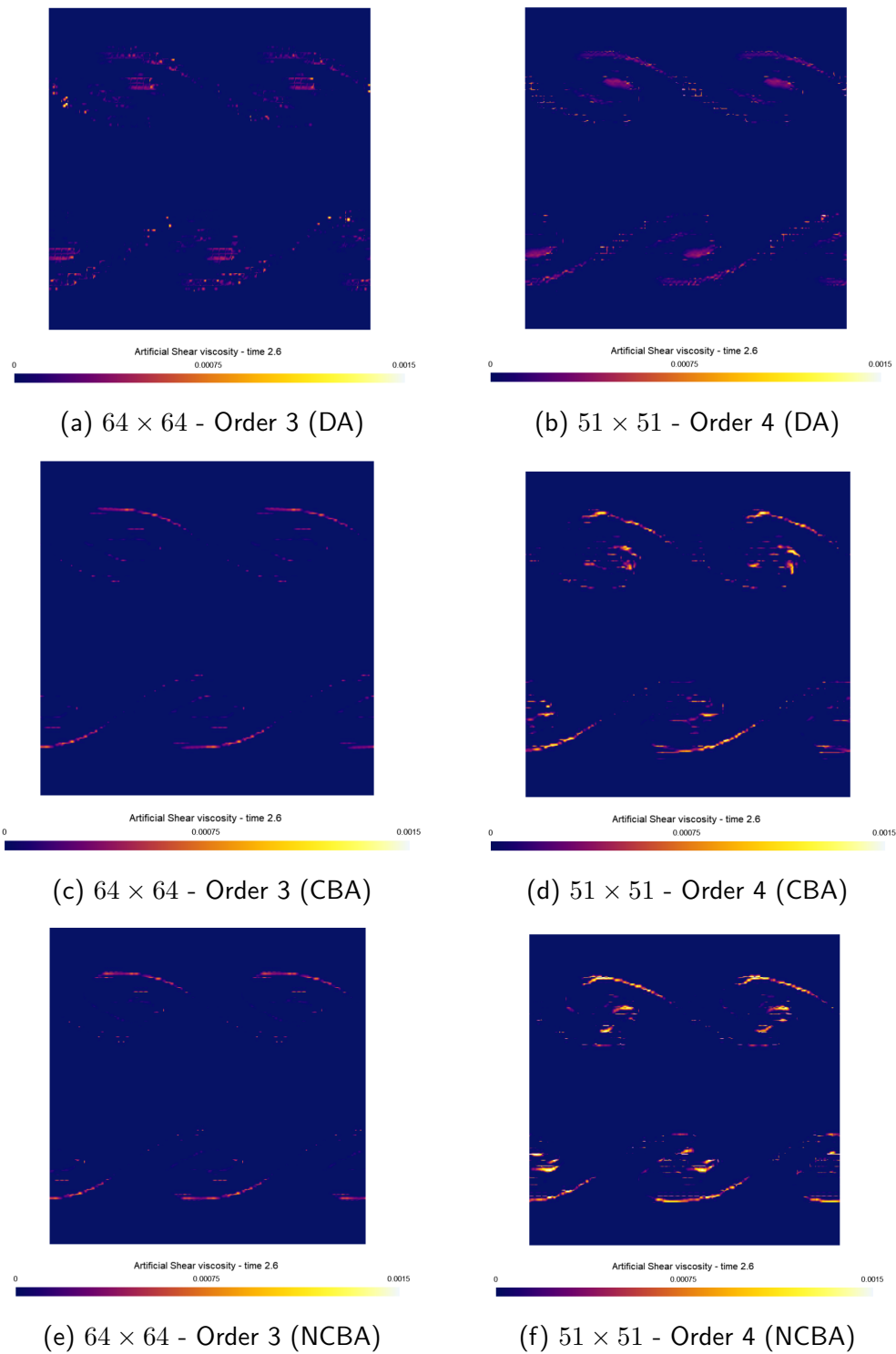
Figure 3.10: Artificial shear viscosity at time t = 2.6 [s] obtained through an artificial viscosity method with Newtonian injection using a physical shear sensor. The figures have been generated for three approaches: DA, CBA, and NCBA. The results are presented with grid configurations of 64×64 using a third-order scheme and 51×51 using fourth-order scheme.

(a) 64×64 - Order 3 (DA)   (b) 51×51 - Order 4 (DA)

(c) 64×64 - Order 3 (CBA)   (d) 51×51 - Order 4 (CBA)

(e) 64×64 - Order 3 (NCBA)   (f) 51×51 - Order 4 (NCBA)
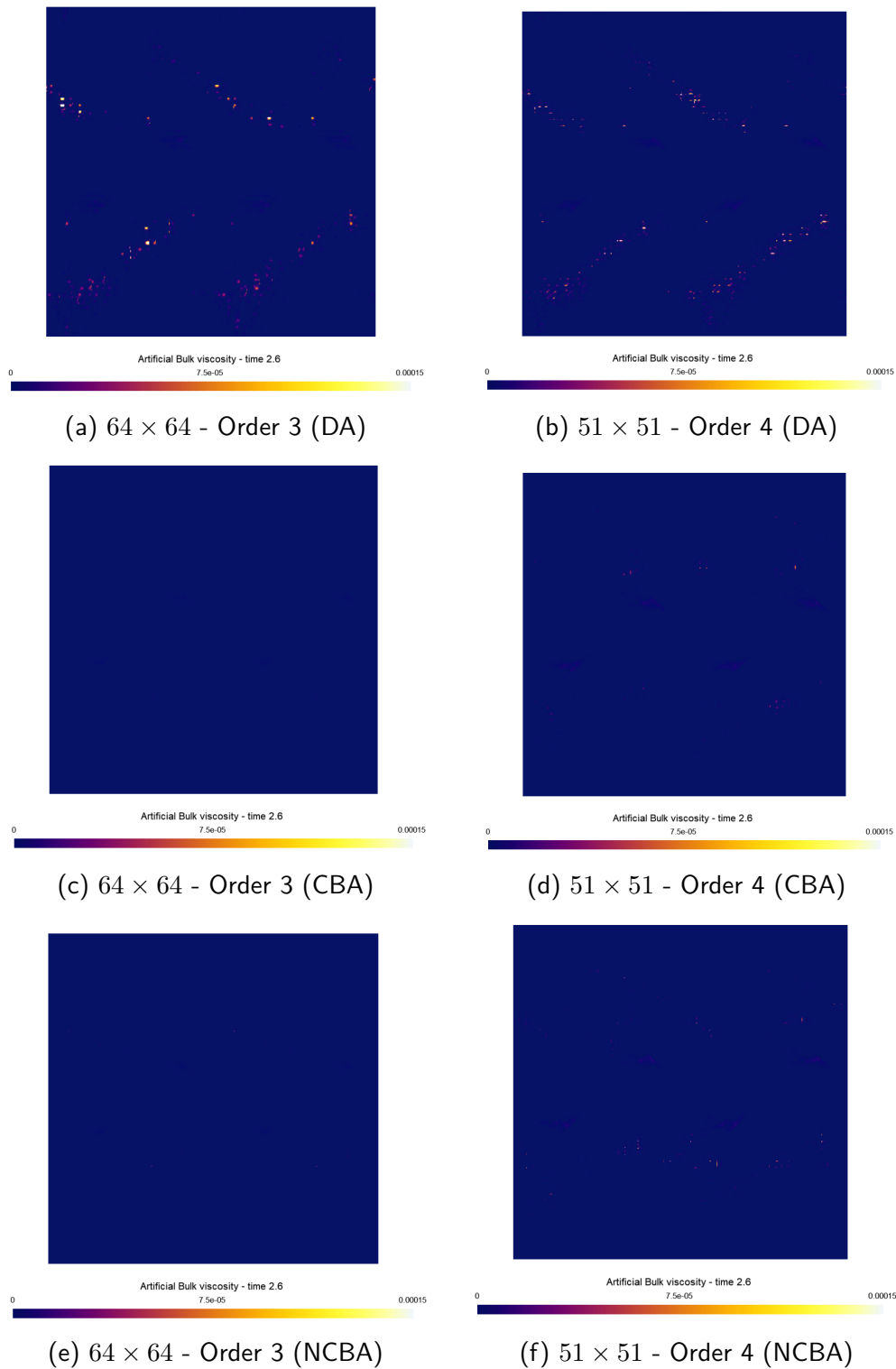
Figure 3.11: Artificial bulk viscosity at time t $=$ 2.6 [s] obtained through an artificial viscosity method with Newtonian injection using a physical shear sensor. The figures have been generated for three approaches: DA, CBA, and NCBA. The results are presented with grid configurations of 64×64 using a third-order scheme and 51×51 using fourth-order scheme.

Analysis at t = 8 [s]

Density

Figure 3.12 illustrates the density profiles obtained by different methods on two distinct mesh configurations.

The test case is noticeably affected by artificial viscosity, leading to rapid visual changes. Regardless of the mesh used, the direct method reveals a significantly higher number of small structures compared to the Bernstein methods, an expected result as these are designed to introduce smoothing.

For both variants of the Bernstein approach, the introduction of continuity at interfaces seems to cause slightly increased dissipation, aligning with expectations as it tends to smooth the method further.

Regarding variations between meshes for the same method, using a higher order appears to lead to more significant numerical dissipation. This reduces the number of visible structures, regardless of the method applied. The impact of the interpolation order on artificial viscosity will be explored in detail in the last section dedicated to the injection method.

Finally, it is noteworthy that, regardless of the chosen approach or mesh configuration, a loss of symmetry is observed in the test case. This asymmetry is likely due to non-symmetrical detection of disturbed cells, highlighting the importance of precision in cell detection to maintain symmetry in simulations.

Artificial viscosity

Figure **??** demonstrates the application of shear and bulk artificial viscosity in a mesh of dimensions 64×64 and order 3, using a direct approach. The selection of this particular image is subjective, but it leads to the same observations regardless of the mesh or method used. Artificial viscosity is primarily applied in the shear zone, influencing the movement of structures.

Regarding bulk viscosity, its impact is almost negligible in the absence of significant shocks. However, it is able to detect low-intensity "shocklets", highlighting its usefulness even in conditions where shocks are not strongly pronounced.

(a) 64×64 - Order 3 (DA)

(b) 51×51 - Order 4 (DA)

(c) 64×64 - Order 3 (CBA)

(d) 51×51 - Order 4 (CBA)

(e) 64×64 - Order 3 (NCBA)
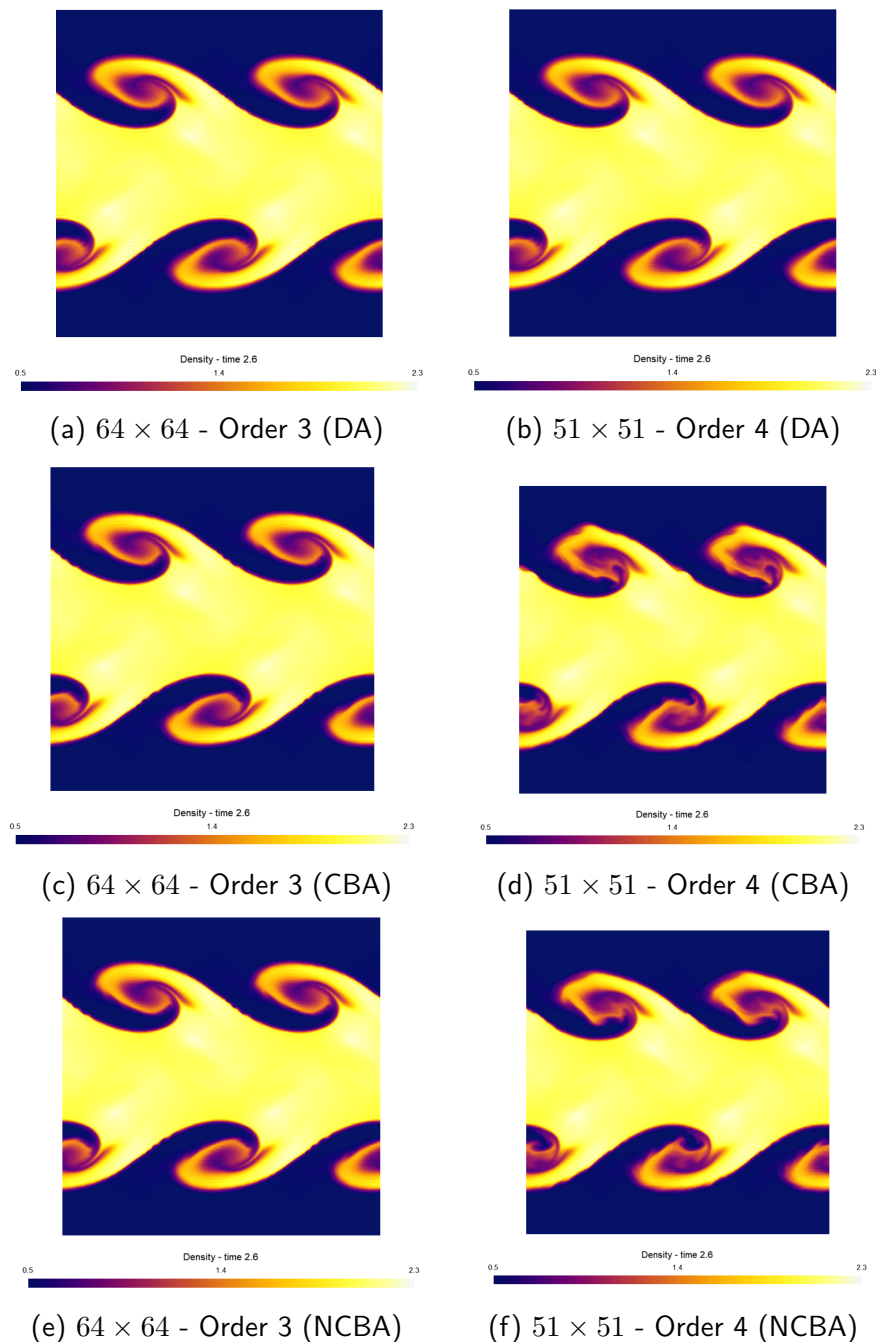
(f) 51×51 - Order 4 (NCBA)

Figure 3.12: Density fields at time t = 8 [s] obtained through an artificial viscosity method with Newtonian injection using a physical sensor. The figures have been generated for three approaches: DA, CBA, and NCBA. The results are presented with grid configurations of 64×64 using a third-order scheme and 51×51 using fourth-order scheme.

(a) Artificial shear viscosity   (b) Artificial bulk viscosity

Figure 3.13: Artificial shear and bulk viscosities at time t = 8 [s] obtained through an artificial viscosity method with Newtonian injection using a physical sensor. The figures have been generated for the direct approach. The results are presented with grid configurations of 64×64 using a third-order scheme.

3.1.2   Quantitative analysis

Artificial viscosity

Fig. 3.13 illustrates the shear and bulk artificial viscosities, integrated over the volume for different approaches and mesh configurations. Regarding shear viscosity, approaches with a 4th order mesh exhibit artificial viscosity from the start of the simulation, even though it is very low in the direct approach. This initial viscosity is attributable to a limitation of the method, related to the geometric factor affecting the shear sensor detection threshold. The details of this issue will be explored more thoroughly in the last section on Newtonian injection.
Apart from this peculiarity, artificial viscosity at equal resolution is higher for Bernstein approaches. These approaches distribute the artificial viscosity over a greater number of points in each cell, thus increasing the overall artificial viscosity.
It is also observed that an increase in order tends to increase artificial viscosity, potentially due to two factors: 4th order induces more instabilities requiring additional artificial viscosities, and the geometric factor increases the amount of viscosity injected.

As for the overall bulk viscosity, it is generally lower than the artificial shear viscosity. It is not affected by the threshold error in the shock sensor detection, which is naturally low. Just like for shear viscosity, the bulk viscosity for a 4th order mesh is higher compared to a lower order.
Furthermore, the direct approach seems to detect more velocity divergences after the halfway point of the simulation. This could be due to the absence of smoothing, as the Bernstein approaches show lower levels of bulk viscosity, and the continuous Bernstein approach appears to require even less bulk viscosity.

Kinetic energy

Fig. 3.15 displays the kinetic energy for different approaches and meshes.

(a) Artificial shear viscosity

(b) Artificial bulk viscosity

Figure 3.14: Artificial shear and bulk viscosities intregrated over the volume obtained through an artificial viscosity method with Newtonian injection using a physical sensor. The results are presented with grid configurations of 64×64 using a third-order scheme and 51×51 using fourth-order scheme.

Figure 3.15: Dimensionless kinetic energy intregrated over the volume obtained through an artificial viscosity method with Newtonian injection using a physical sensor. The results are presented with grid configurations of 64×64 using a third-order scheme and 51×51 using fourth-order scheme.

This quantity clearly illustrates the effect of numerical dissipation due to the addition of artificial viscosity. It is evident that the more artificial viscosity is injected, the more the kinetic energy

decreases, thus indicating a more pronounced numerical dissipation.

The direct approach with a 3rd order mesh shows the least artificial viscosity dissipation. In contrast, approaches using Bernstein polynomials with a 4th order mesh, especially the one with continuity at interfaces, exhibit greater energy dissipation.

Kinetic energy budget

Fig. 3.16 illustrates the kinetic energy budget for various approaches, analyzed in two different ways for a mesh of 64×64 of order 3 and a mesh of 51×51 of order 4.

(a) 64×64 - Order 3   (b) 51×51 - Order 4

Figure 3.16: Kinetic energy budget obtained through an artificial viscosity method with Newtonian injection using a physical sensor. The results are presented with grid configurations of 64×64 using a third-order scheme and 51×51 using fourth-order scheme.

The two graphs show similarities, especially in terms of orders of magnitude.

For the Bernstein approaches, whether continuous or not, the curves are relatively similar regardless of the mesh. They exhibit a maximum drop around 3 seconds, a critical moment where the simulation crashes without stabilization methods. Subsequently, the curves tend to converge in an attempt to rebalance the budget. In these two approaches with a 4th order, the drop occurs more quickly, even from the beginning for the CBA approach, due to the early injection of artificial viscosity affecting the start of the budget.

As for the direct approach, it shows a more pronounced drop. However, like the two Bernstein approaches, it demonstrates a tendency to readjust the budget towards the end of the simulation.

3.1.3   Conclusion

The direct approach displays better accuracy and uses less artificial viscosity than other methods at equal resolution. However, it tends to produce more discontinuities, requiring the application of a more significant bulk viscosity for their management.

The Bernstein methods introduce additional smoothing into the solution. Their advantage lies in the ability of these polynomials to distribute artificial viscosity across the entire cell rather than at a single point, but this comes at the cost of greater numerical dissipation. While this dissipation

reduces precision, it enhances the system's robustness. These approaches remain coherent and valid, particularly in terms of kinetic energy budget management, and offer superior performance compared to the PerssonPeraire sensor associated with Laplacian injection.
The integration of continuity at interfaces also affects the smoothing and consequently the numerical dissipation, especially in cases where the interpolation order is high (unstable).

The detection threshold of the shear sensor varies depending on the mesh used, leading to an early injection of artificial viscosity for a 4th order mesh, thus affecting all quantities produced by the simulations. However, it is important to note that the artificial viscosity method adapts to best regulate this early injection.

Finally, regardless of the chosen configuration or mesh, the density field exhibits a symmetry break, which could be attributed to an asymmetric injection of artificial viscosity in the simulation.

3.2 Effects of Detection Threshold in Shear Physical Sensor

In the previous section, it was observed that artificial viscosities were introduced at the beginning of the simulation, which is not typically desired and indicates a detection issue. To address this, this section focuses on analyzing different detection thresholds for the shear sensor. It has been established that the shock sensor, with an already low threshold, should detect all shocks, while the shear sensor should only respond to extremely intense shear issues that could lead to a simulation crash.

The aim here is to examine five different thresholds for the shear sensor, set at 0.5, 1.0, 1.2, 1.5, and 1.8, respectively. This will involve observing the density field and the field of artificial viscosity at these various thresholds. The analysis will then focus on the behavior of artificial viscosity, kinetic energy, and the kinetic energy budget across the entire volume studied.

The employed mesh is 64×64 of order 3 to avoid the systematic early injection of artificial viscosity and to ascertain whether it originates from a lower detection threshold.

The selected approach is the direct method, which does not introduce additional interpolation that could skew the results.

Note Setting the detection threshold at 1.5 leads to a crash of the simulation, hence visualizing it is not possible. However, this simulation will be analyzed in the quantitative analysis to try and determine the reasons for its crash.

3.2.1 Visual analysis

Density fields

Fig. 3.17 illustrates the density fields at time t = 8 [s] corresponding to various detection thresholds established earlier in the section. Each figure reveals a profile that is similar yet distinct from the others. Notably, the fields with the highest accuracy are associated with detection thresholds of 1.0 and 1.2. This finding is somewhat counterintuitive, as one might expect that the highest detection threshold would yield the most realistic figure (with the least numerical dissipation). However, this is not the case; although the figure with the highest threshold is relatively acceptable, it is still less precise than the other two.

Artificial viscosity

It is prudent to present Fig. 3.18 to provide a spatial understanding, even though it only serves to confirm more marked and intense detection at t = 8 [s] with a shear sensor threshold of 1.8, rather than 1.2 or 1.0. However, direct comparison becomes complex due to the fact that the artificial viscosities are not located in the same places, nor with the same intensity, right from the start of the simulation.

(a) $s_{\mu,0} = 0.5$

(b) $s_{\mu,0} = 1.0$
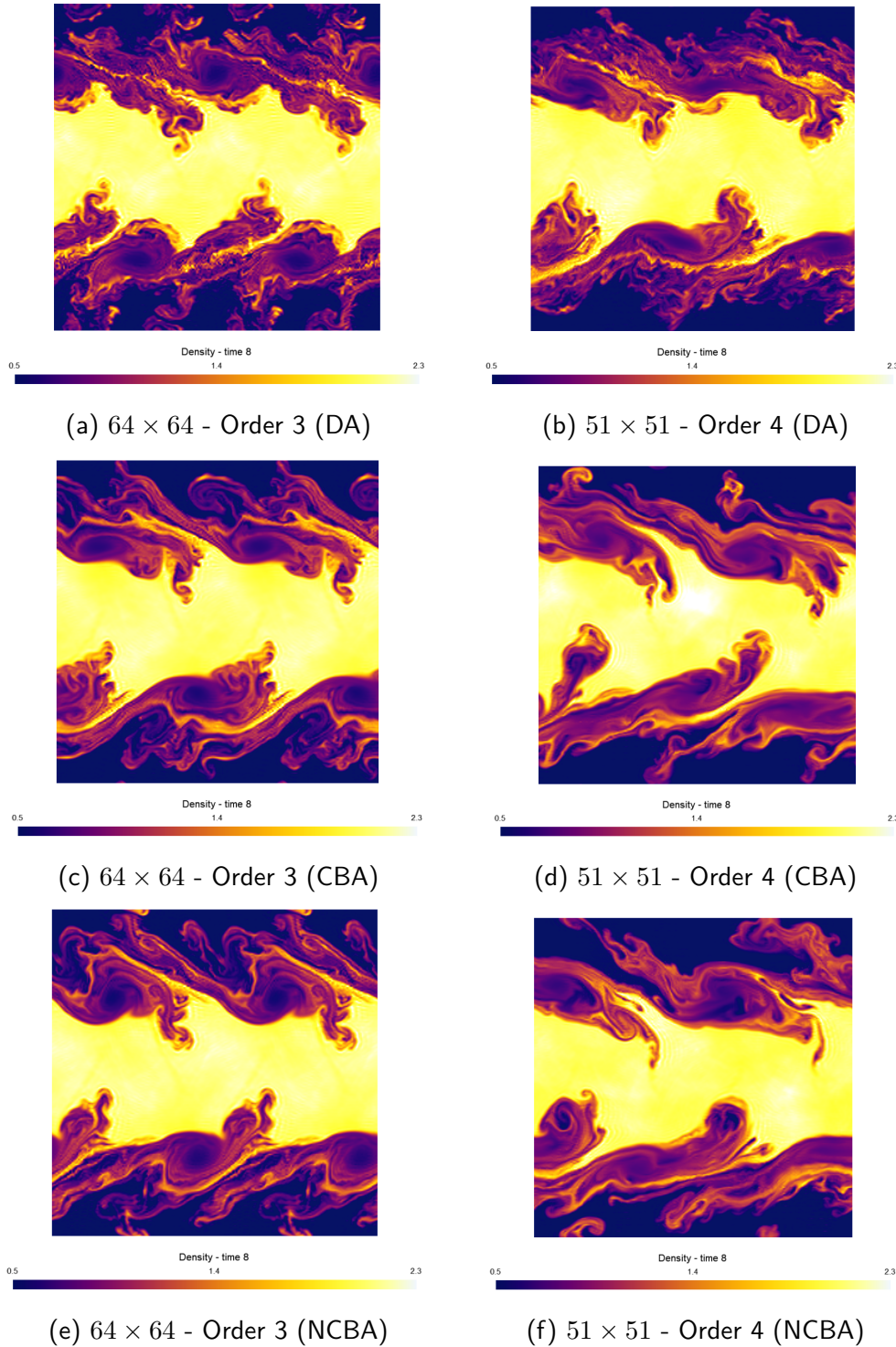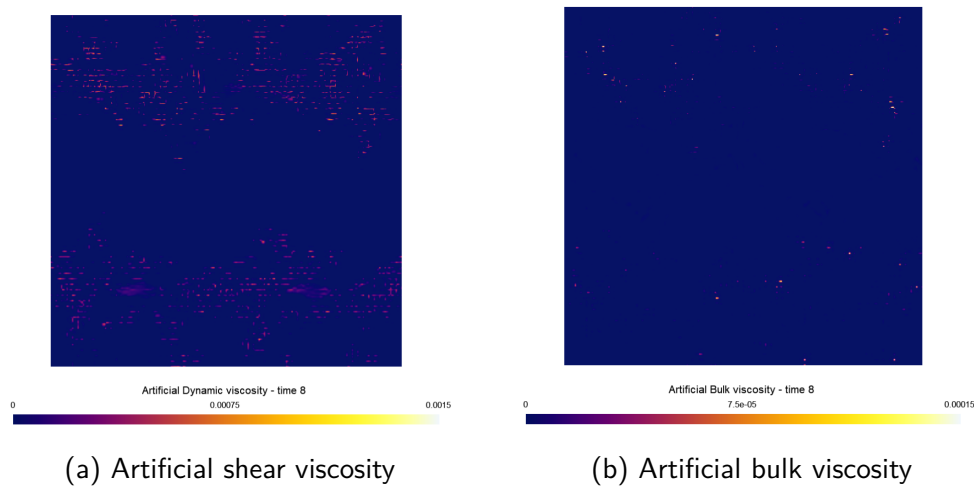

(c) $s_{\mu,0} = 1.2$

(d) $s_{\mu,0} = 1.8$
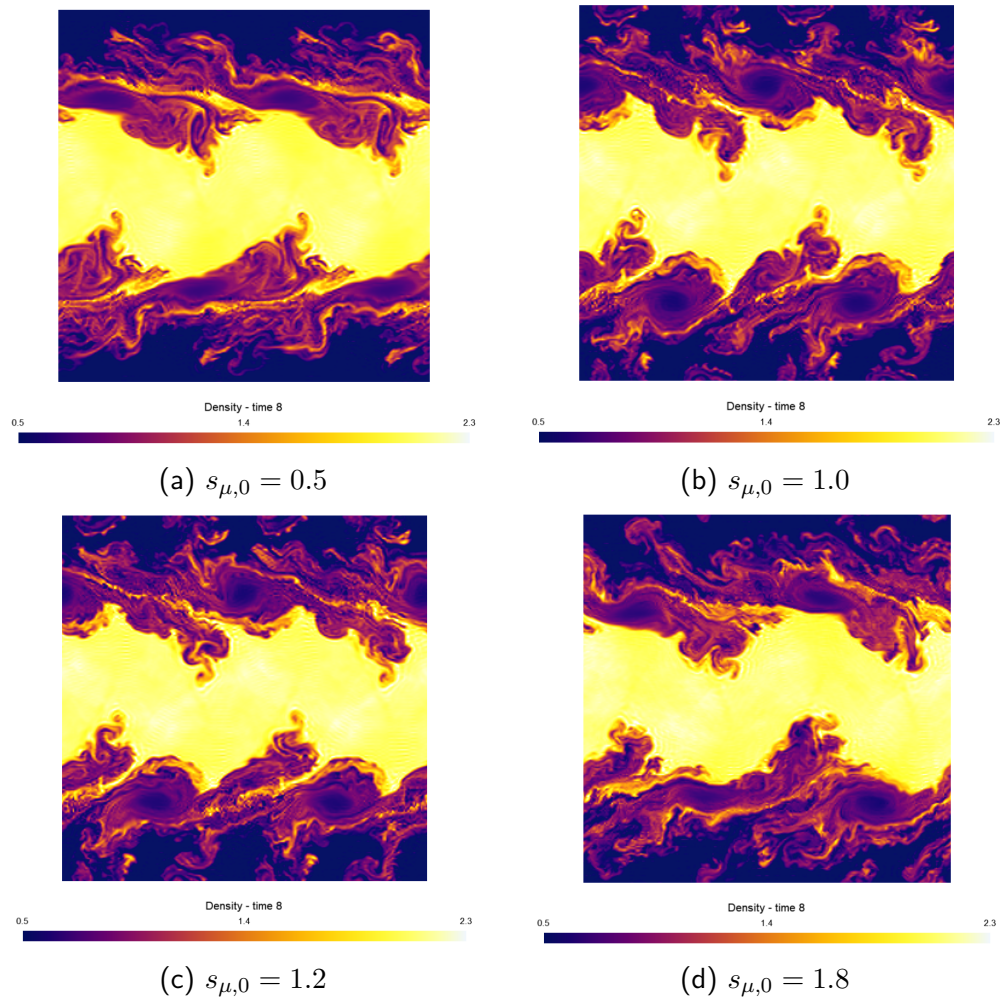
Figure 3.17: Density fields at time t = 8 [s] obtained through an artificial viscosity method with Newtonian injection using a physical sensor. The figures have been generated for four detection thresholds: 0.5, 1.0, 1.2 and 1.8. The results are presented with grid configurations of 64×64 using a third-order scheme.

(a) $s_{\mu,0} = 0.5$

(b) $s_{\mu,0} = 1.0$

(c) $s_{\mu,0} = 1.2$

(d) $s_{\mu,0} = 1.8$
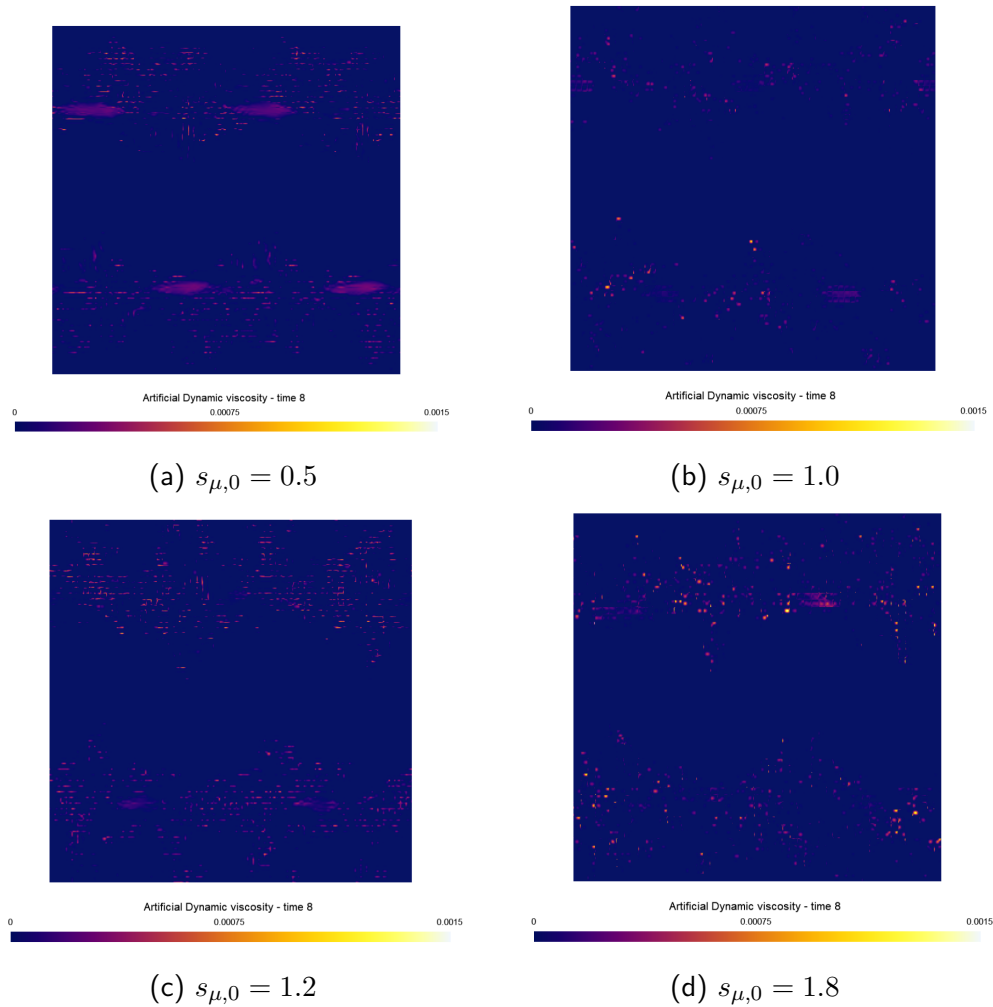
Figure 3.18: Artificial shear viscosity at time t $= 8$ [s] obtained through an artificial viscosity method with Newtonian injection using a physical sensor. The figures have been generated for four detection thresholds: 0.5, 1.0, 1.2 and 1.8. The results are presented with grid configurations of 64×64 using a third-order scheme.

3.2.2   Quantitative analysis

Artificial viscosity

Fig. 3.19 illustrates the artificial viscosities integrated into the volume.

(a) Artificial shear viscosity   (b) Artificial bulk viscosity
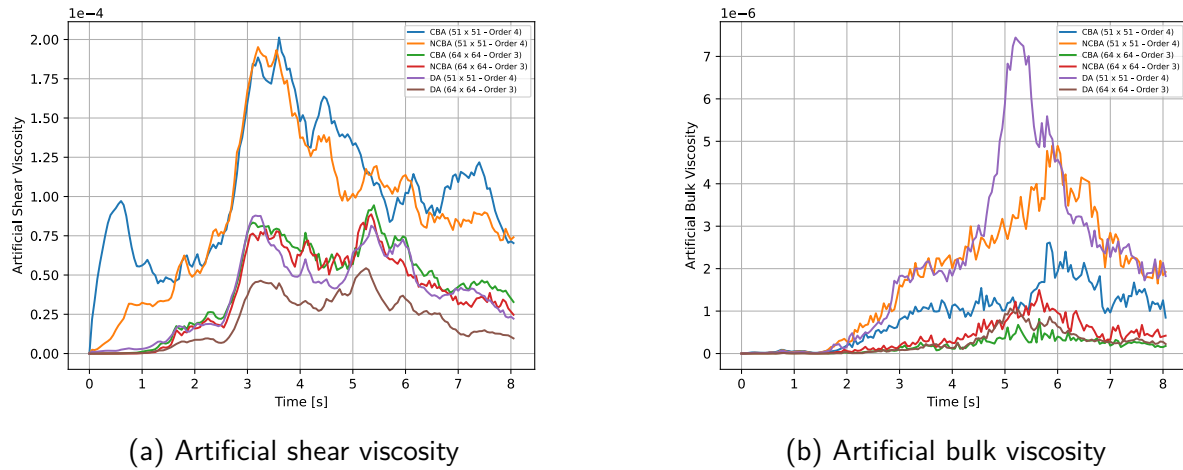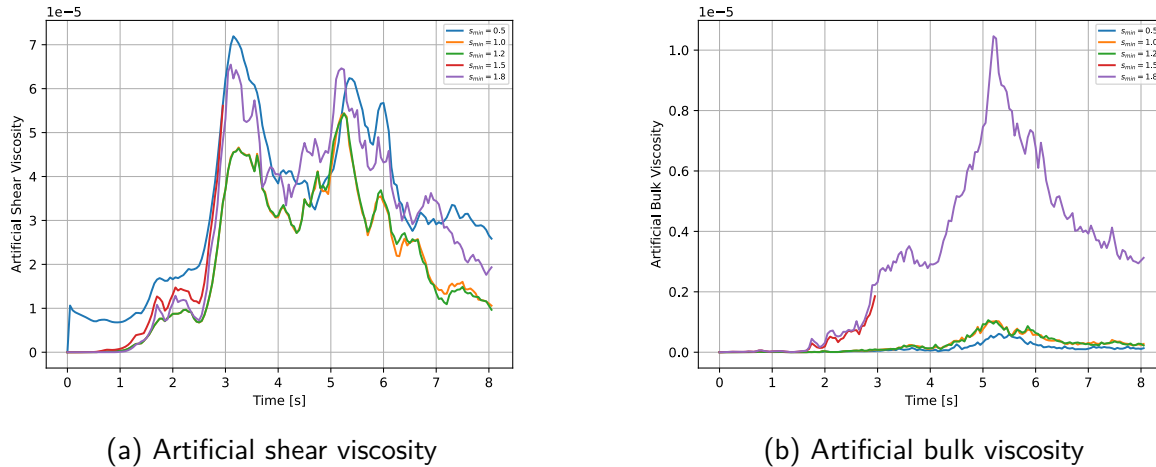
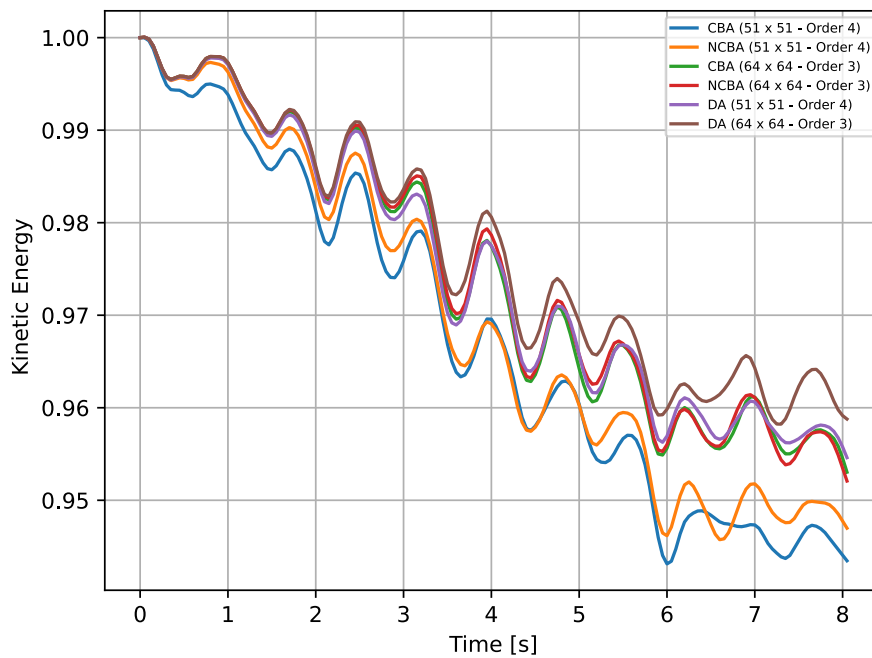Figure 3.19: Artificial shear and bulk viscosities intregrated over the volume obtained through an artificial viscosity method with Newtonian injection using a physical sensor. The figures have been generated for four detection thresholds: 0.5, 1.0, 1.2, 1.5 and 1.8. The results are presented with grid configurations of 64×64 using a third-order scheme.

Regarding artificial shear viscosity, it is immediately noticeable that with a lower threshold, there is an initial injection of artificial viscosity similar to what is observed in Bernstein's approaches. This suggests that the issue lies in the detection threshold, which would therefore require adjustment. With a lower detection threshold, it makes sense to observe an increase in shear artificial viscosities due to broader detection. However, with a detection threshold exceeding a certain value, artificial viscosity also increases, as seen with detection thresholds of 1.5 and 1.8, which lead to an increase in shear artificial viscosities. This rise is attributed to discontinuities.

For these two thresholds, it is also observed that the bulk viscosity is particularly intense. The increase in both shear and bulk artificial viscosities is due to an excessively high detection threshold, leading to instabilities that can crash the simulation, as seen with a threshold set at 1.5. For this reason, the stabilization method adjusts by injecting a larger amount of artificial viscosities.

Kinetic energy

Fig. 3.20 shows the integrated kinetic energy within the volume.

As previously observed, increasing the threshold affects the artificial viscosities, leading to an increase in numerical dissipation. Consequently, detection thresholds with more artificial viscosity dissipate more energy and have lower curves.

Although there is a difference between these curves, the kinetic energy is higher even for the smallest detection threshold compared to the PerssonPeraire sensor with Laplacian injection or the fourth-order Bernstein approach.

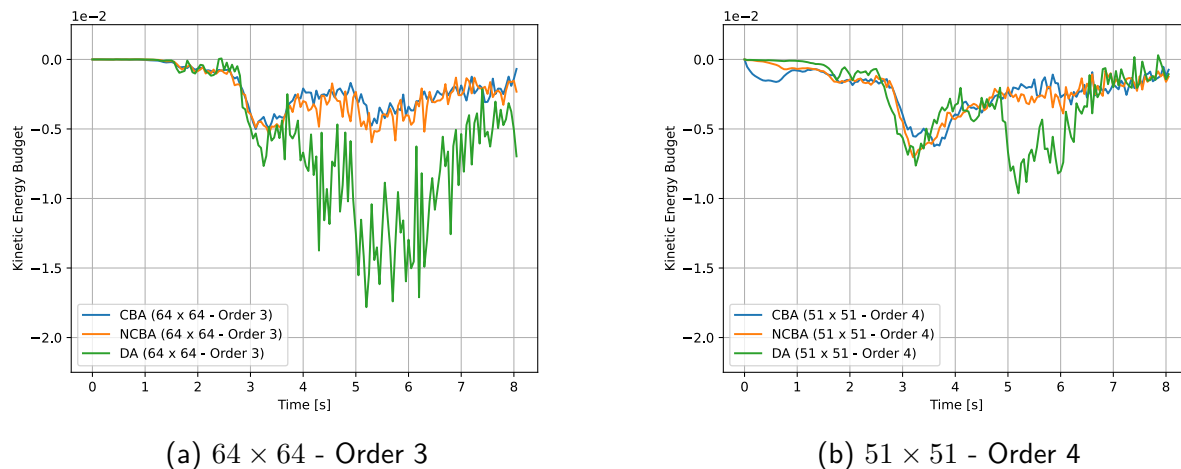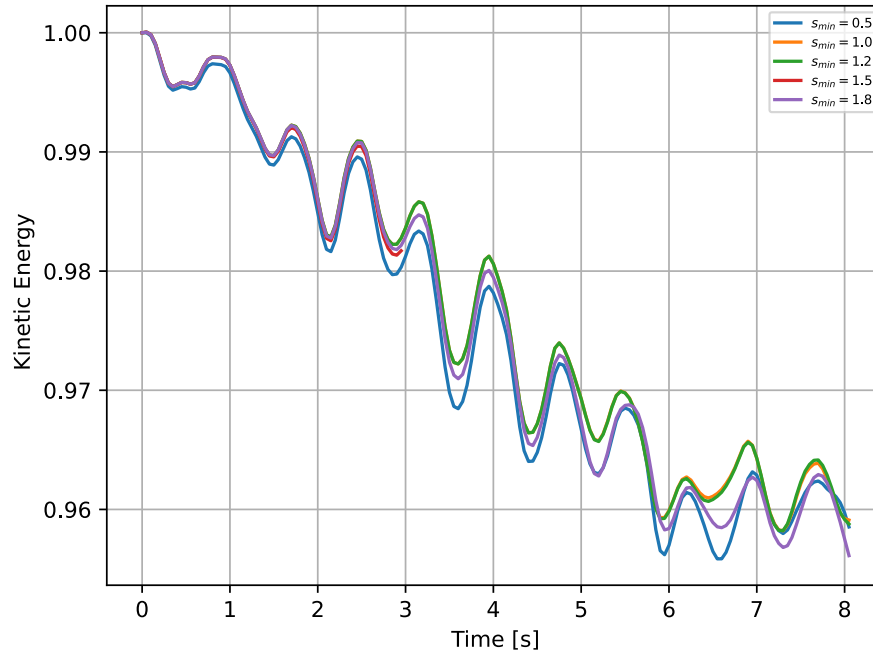Figure 3.20: Dimensionless kinetic energy obtained through an artificial viscosity method with Newtonian injection using a physical sensor. The figures have been generated for four detection thresholds: 0.5, 1.0, 1.2, 1.5 and 1.8. The results are presented with grid configurations of 64×64 using a third-order scheme.

Kinetic energy budget

For the budget depicted in Fig. 3.21, it is noted that different detection thresholds influence the budget while maintaining a similar dynamic. In other words, there is an initial drop in the budget caused by the addition of artificial viscosity, followed by an effort of the simulation to rebalance the budget subsequently. It's also relevant to highlight that the most balanced budget is observed with the threshold of 1.8, likely due to the additional inclusion of bulk artificial viscosity.

3.2.3 Conclusion

The choice of detection threshold for the shear sensor is crucial when it is used in conjunction with an artificial viscosity method. The findings indicate that the size of the threshold does not necessarily ensure optimal performance. On the contrary, it is vital to strike a balance in order to detect destabilizing phenomena before they become unmanageable and require the injection of an increased amount of artificial viscosity, which could then impact the rest of the simulation.

This section also highlighted, in a way, the instability associated with the use of physical sensors such as the shear sensor. In practice, the implementation of a pointwise shear sensor proves to be complex for reliable usage, with its performance potentially varying greatly due to user parameters, the nature of the test cases, and other factors. Consequently, intermediate detection thresholds (like 1.5) might prove to be ineffective, while lower or higher thresholds can be more effective.
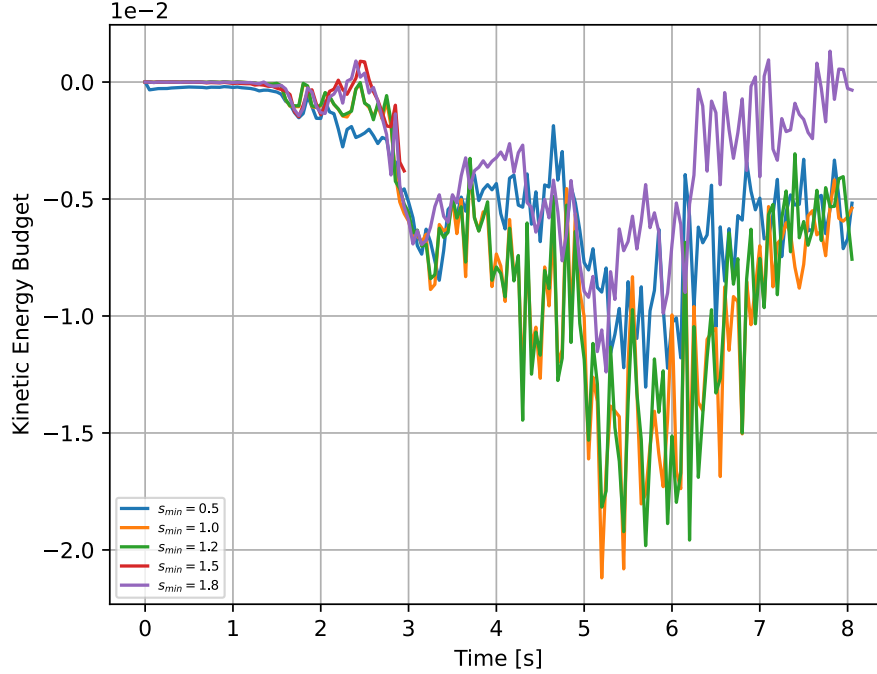
Figure 3.21: Kinetic energy budget obtained through an artificial viscosity method with Newtonian injection using a physical sensor. The figures have been generated for four detection thresholds: 0.5, 1.0, 1.2, 1.5 and 1.8. The results are presented with grid configurations of 64×64 using a third-order scheme.

3.3   Challenges Arising from Geometric Factors

This section can be discussed in general terms, but it becomes particularly significant in the case of Kelvin-Helmholtz instabilities. Artificial viscosity employs a geometric factor to scale the calculated viscosity. This factor plays a role both in the computation of artificial viscosity and in the physical detector. However, the regular sensors discussed here do not require this scaling because they are based on the solution itself. Article [38] proposes three possible geometric factors, as illustrated in the equation,

$$g(h,p) = \begin{cases} h \times (2 - Max(d)) & \text{[Linear -]} \\ h \times Max(d) & \text{[Linear +]} \\ h/p & \text{[Constant]} \end{cases}$$

where d represents the distance between two interpolation points. It is important to note that linear geometric factors present a disadvantage due to their dependence on shape functions. This issue will be discussed in more detail later in this section.

In the case of Kelvin-Helmholtz instabilities, the geometric factor used was *Linear-*, as it involves a more significant scaling factor. Consequently, in the calculation of artificial viscosity, this has the effect of increasing the quantity, thereby adding more dissipation and enhancing robustness. However, regarding the physical detector, the situation presents itself from a different angle.

To better understand the issue, it is essential to revisit the process of calculating physical sensors $\hat{s}$. In this calculation, a physical quantity Q is first evaluated. This quantity is then multiplied by the geometric factor $g(h, p)$, creating a sensor s that has not yet been incorporated into the limiting function. The next step, therefore, is to integrate this sensor s into the limiting function, which yields the actual physical sensor $\hat{s}$. The fundamental objective of this limiting function is to check if the sensor s exceeds the limiting threshold s_{smin}. If so, it returns a non-zero value; otherwise, it returns a zero value.

For simplification, and since the limiting function only changes the value while maintaining the same principle, it is possible to bypass it. The importance of the scaling factor in the detector then becomes more evident. Detection will be made if the following relationship is respected,

$$Q > s_{min}/g(h, p). \tag{3.2}$$

The detection threshold can therefore be re-expressed in terms of the quantity $s_{min}/g(h, p)$. This detection threshold then becomes dependent on the order, exhibiting different behavior depending on the geometric factor used. Table T. 3.1 illustrates the variations in behavior of the geometric factors as the order increases.

Geometric scale	p	$g(h, p)$	$s_{min}/g(h, p)$
Linear-	↗	↗	↘
Linear+	↗	↘	↗
Constant	↗	↘	↗

Table 3.1: Behavior of geometric factors and detection threshold as a function of order.

The *Linear-* geometric factor exhibits an inversely proportional relationship with the detection threshold based on the order, while other factors have a proportional relationship. There is no clearly better or worse geometric factor, as increasing the order inevitably makes the simulation more unstable, so decreasing the detection threshold would allow the introduction of a greater amount of artificial viscosity. On the other hand, as the order of a numerical scheme increases, it means that the scheme is capable of more accurately representing gradients and discontinuities in the solution, including shocks. Consequently, with a high-order numerical scheme, artificial viscosity can be reduced compared to a lower-order scheme. This is because the high-order scheme can more faithfully represent the behavior of the shock without needing excessive artificial viscosity to stabilize the solution. However, it is important to note that reducing artificial viscosity with order is not necessarily an absolute rule.

Now that the impact of order dependence has been mentioned, as it affects the detection threshold, it is important to quantify this impact and determine whether it is negligible or not. To do this, a numerical example is presented with the Kelvin-Helmholtz test case, using a mesh of 64×64 order 3 and a minimal detection threshold of $s_{min} = 1$, using Gauss-Lobatto-Legendre shape functions. Table T.3.2 shows a significant difference in the detection threshold between the constant factor and linear factors. Additionally, the intensity of artificial viscosity is much more reduced using the constant factor compared to the linear factors for the same resolution.

The choice of the scaling factor has a significant impact on the simulation, affecting both the intensity of artificial viscosity and the detection threshold of sensors that require scaling. The

Geometric scale	h	$g(h,p)$	$s_{min}/g(h,p)$
Linear-	0.03125	0.0345	28.99
Linear+	0.03125	0.0279	35.84
Constant	0.03125	0.0104	96.15

Table 3.2: Impact of the scaling factor on the detection threshold for the Kelvin-Helmholtz test case.

main drawback lies in the dependence on the basis functions for linear scaling factors, although they show good results [38]. Therefore, it would be prudent to recommend the use of the *constant* scaling factor, which unfortunately does not always allow detection due to its very high threshold, as was the case for the Kelvin-Helmholtz instability test case.

In conclusion, the three scaling factors each have their advantages and disadvantages. When combined with a well-calibrated method of artificial viscosity, each of these scaling factors could be equivalent with proper calibration.

4 Evaluating Laplacian and Newtonian Injection Techniques

The objective of this section is to compare two methods of artificial viscosity injection. Although this comparison has been individually addressed in different sections, it is wise to examine them together for a better understanding of their differences.

To do this, the best results from each method will be considered. On one hand, there is the Laplacian injection method using the Hennemann sensor, and on the other, the Newtonian injection method using physical sensors with a direct approach.

It is relevant to retain the same meshes that were used in previous discussions. These meshes reveal the distinct behavior of each method, particularly in terms of the impact of increasing the interpolation order.

4.1 Visual analysis

The visual analysis focuses exclusively on two key elements. The first is the visualization of artificial viscosity at the specific time $t = 2.6$ [s]. This moment has been selected to ensure an accurate understanding of where the artificial viscosity needs to be applied. This precision is crucial because, although the disturbances caused by the sensors are observable at the end of the simulation, it remains difficult to judge their effectiveness due to the chaotic nature of the simulation.

The second element is the observation of the density field at the final moment of the simulation, $t = 8$ [s]. This observation aims to determine the real impact of the artificial viscosity on the evolution of the simulation.

Artificial viscosity at 2.6 [s]

While direct comparison of artificial viscosities is not feasible, their associated sensors can be compared. Fig. 3.22 demonstrates the artificial viscosity fields for the two injection methods discussed.

The physical shear sensor captures the deformation in the shear layer more accurately. Due to its pointwise nature, it refines the field by concentrating intensity on specific points, which are typically the most likely to cause issues.

On the other hand, the Hennemann sensor, though based more on a numerical approach than a physical one, also manages to faithfully replicate the problematic areas. It adapts well to the injection of artificial viscosity. Its elementwise nature does not allow for as precise a delineation of problematic areas. However, by adding extra viscosity to disturbed zones, it enhances the robustness of the simulation.

Density fields

Fig. 3.23 illustrates the density fields obtained by two distinct injection methods. The impact of the Laplacian method is evident when comparing the two visualizations. There is a clear effect of the Laplacian, which tends to homogenize the density field, resulting in a smoother and more uniform representation. In contrast, the approach based on Newtonian viscosity, while presenting similar characteristics, reveals a more hatched texture and more abrupt transitions.

All the figures break the symmetry due to the asymmetrical injection of artificial viscosity.

(a) 64×64 - Order 3
(Laplacian injection)

(b) 51×51 - Order 4
(Laplacian injection)

(c) 64×64 - Order 3
(Newtonian injection)

(d) 51×51 - Order 4
(Newtonian injection)
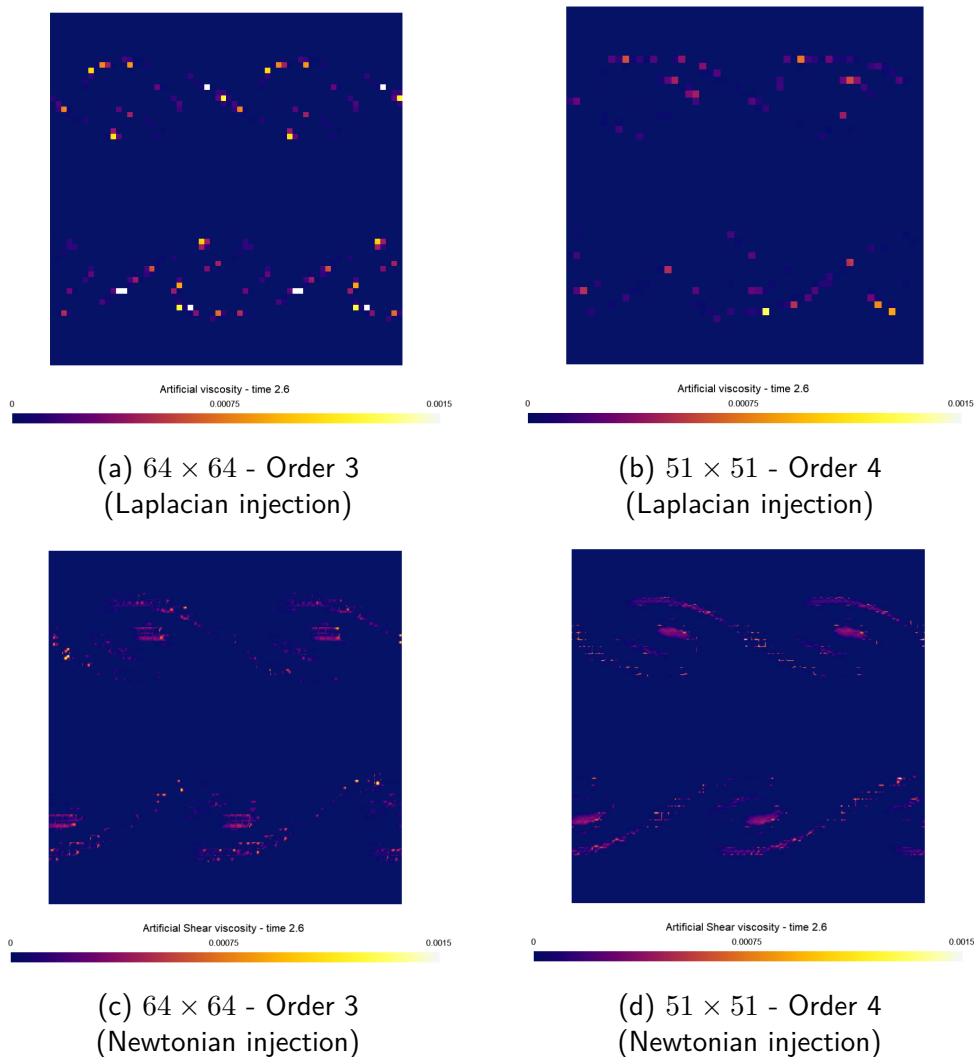
Figure 3.22: Artificial shear viscosity at time t = 2.6 [s] obtained through an artificial viscosity method with Laplacian and Newtonian injection. The results are presented with grid configurations of 64×64 using a third-order scheme and 51×51 using fourth-order scheme.

It is important to remember, as mentioned in the previous chapter, that the use of a higher order leads to a significant reduction in precision. This issue arises from the integration of the geometric factor "*Linear-*", which tends to lower the detection threshold of the shear sensor, thus increasing artificial viscosities with the rise in interpolation order, thereby negatively impacting the quality of the simulation.

4.2   Quantitative analysis

Kinetic energy

The kinetic energy Fig. 3.24 remains within a relatively narrow range. In comparison to the PerssonPeraire sensor, where the final kinetic energy had decreased by 7-8% from its initial value, here the loss is consistently around 4.5%. This indicates that the kinetic energy generated, regardless of the method used, stays within a closely similar range.

(a) 64×64 - Order 3
(Laplacian injection)

(b) 51×51 - Order 4
(Laplacian injection)

(c) 64×64 - Order 3
(Newtonian injection)

(d) 51×51 - Order 4
(Newtonian injection)
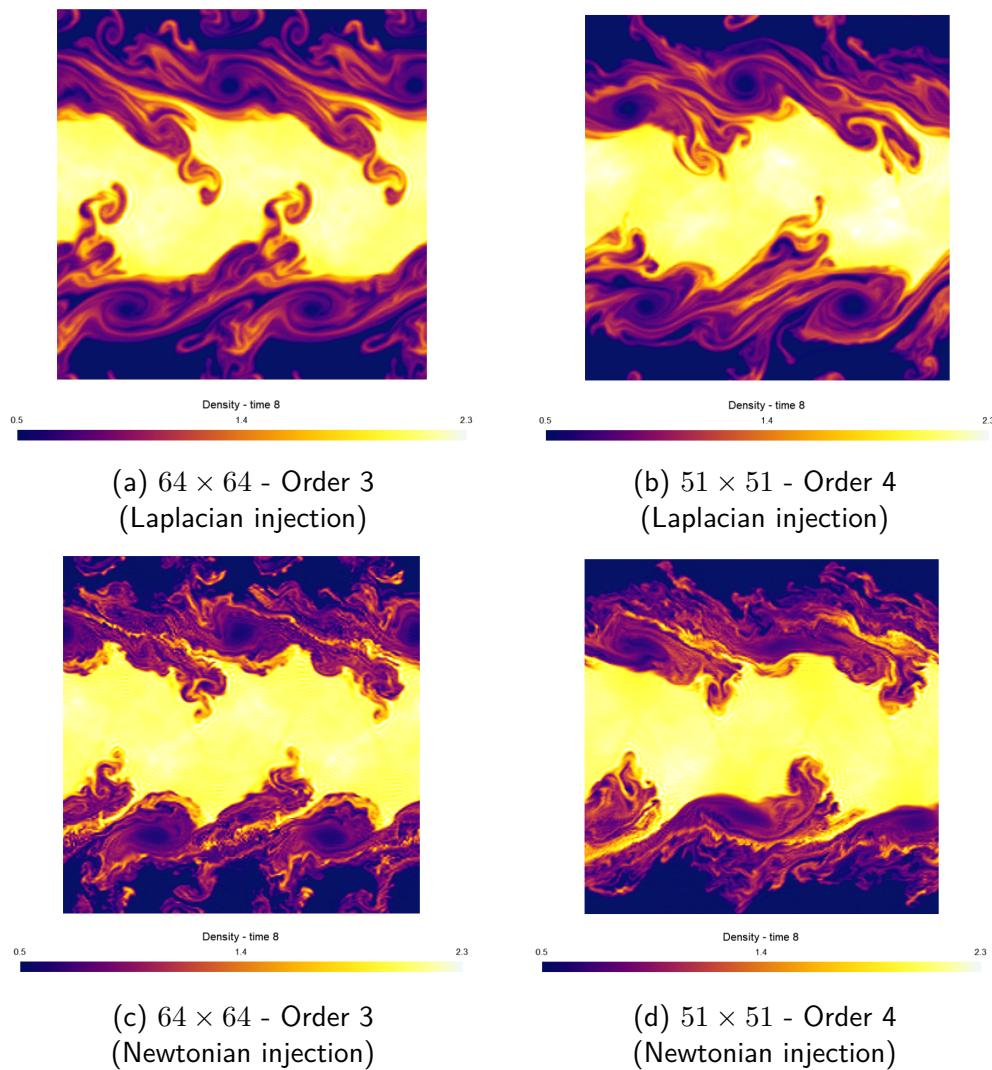
Figure 3.23: Artificial shear viscosity at time t $= 2.6$ [s] obtained through an artificial viscosity method with Laplacian and Newtonian injection. The results are presented with grid configurations of 64×64 using a third-order scheme and 51×51 using fourth-order scheme.

The use of Newtonian injection at the third order exhibits slightly less numerical dissipation than that of the Laplacian injection. However, at the fourth order, this method suffers from degradation due to a sensor failure, affecting the simulation throughout its course. It tends, however, to stabilize towards the end of the simulation, reaching values similar to those of the Laplacian injection at the same resolution. Nevertheless, the initial impact prevented the formation of certain structures, thereby altering the overall dynamics of the test case.

Kinetic energy budget

The kinetic energy budget, illustrated in Fig. 3.25, shows that a Laplacian injection, although leading to a drop due to the addition of artificial viscosity, causes a less pronounced drop than that observed with a Newtonian injection.
This more pronounced drop could be attributed to several factors, including the use of the New-
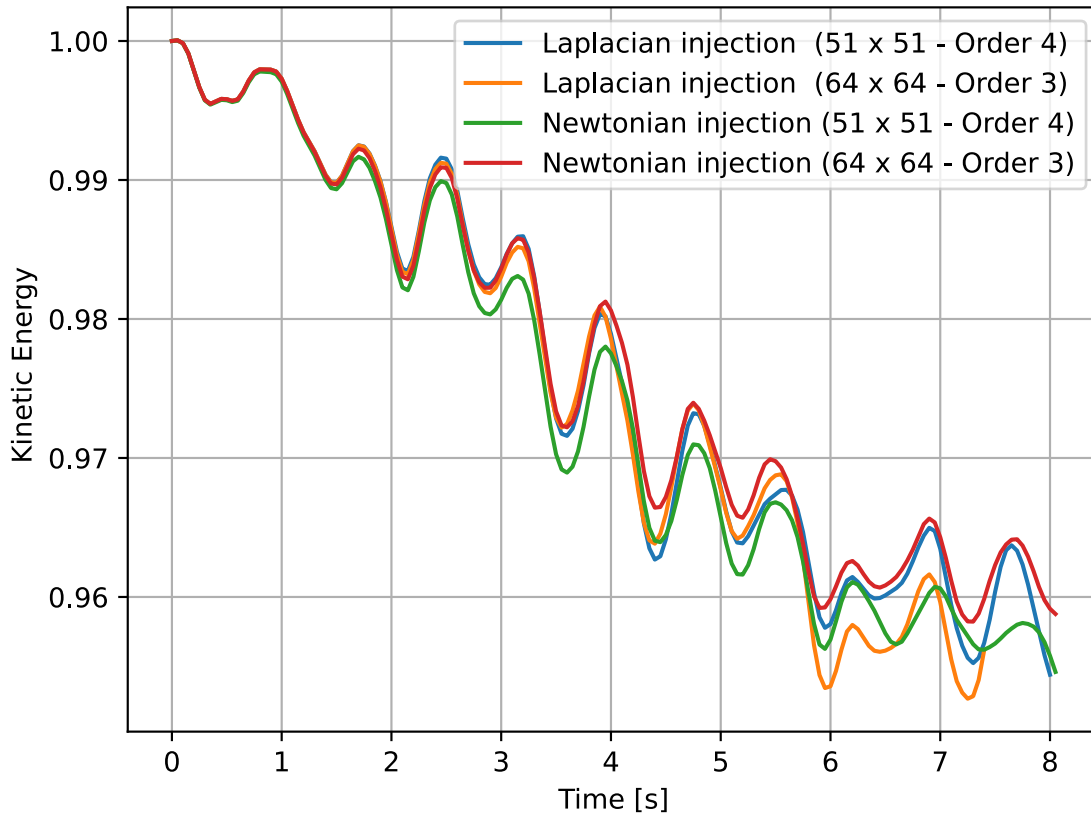
Figure 3.24: Dimensionnless kinetic energy obtained through an artificial viscosity method with Laplacian and Newtonian injection. The results are presented with grid configurations of 64×64 using a third-order scheme and 51×51 using fourth-order scheme.

tonian injection method. As previously observed in the density field, this method is characterized by more abrupt transitions, unlike the Laplacian injection which tends to smooth the solution. Additionally, the use of sensors of different natures may also play a role in this phenomenon. However, all methods, following this initial drop, will gradually attempt to rebalance the kinetic energy budget throughout the course of the simulation.

4.3   Conclusion

The two injection techniques used in the simulation create visually distinct outcomes. The Laplacian injection leads to more homogeneous and streamlined structures, suggesting a viscosity injection that promotes smoothness and uniformity. On the other hand, the Newtonian injection results in more erratic and chaotic patterns, with sharp density variations and intricate, fine details. This approach may yield outcomes that more closely replicate natural phenomena but with a greater propensity for instability.

Regarding the numerical dissipation of kinetic energy, both simulations exhibit relatively comparable patterns. However, when examining the kinetic energy budget, a significant difference
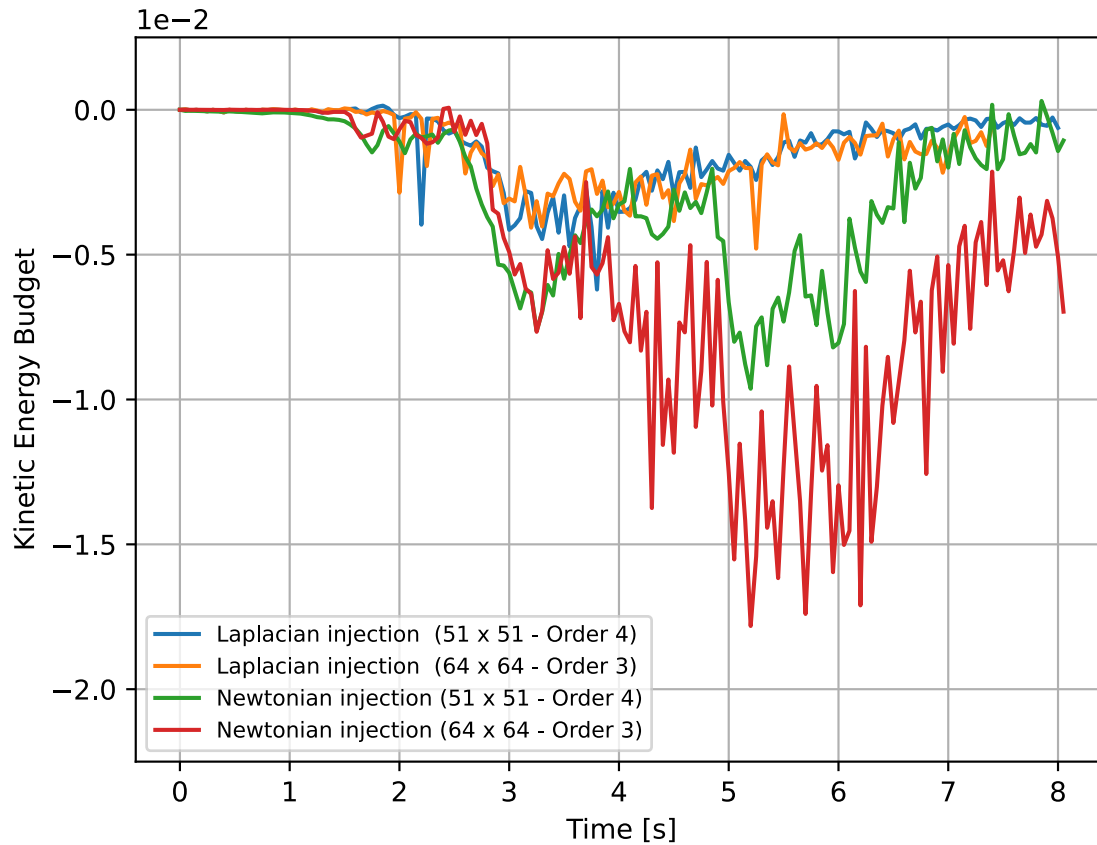
Figure 3.25: Kinetic energy budget obtained through an artificial viscosity method with Laplacian and Newtonian injection. The results are presented with grid configurations of 64×64 using a third-order scheme and 51×51 using fourth-order scheme.

emerges: the Laplacian injection tends to result in narrower budget variances compared to the Newtonian injection, possibly indicating greater robustness. Nonetheless, both methods demonstrate self-regulation over time to rebalance the budget.

Chapter 4

Inviscid Strong Vortex-Shock Wave Interaction

The interaction between the shock wave and the strong vortex is crucial for understanding various phenomena, such as the stability of supersonic and hypersonic vehicles, the properties of high-speed flows, and the characteristics of flows in the Earth's atmosphere and other astrophysical environments. It also ensures that the scheme can capture unsteady compressible flows and complex physical phenomena resulting from this interaction.

In a context of inviscid flow, the interaction between a shock wave and a strong vortex is one of the fundamental test cases in computational fluid dynamics (CFD) and numerical aerodynamics. This classical study of the interaction between two major phenomena in fluid dynamics sheds light on the complex and aspects of compressible fluid mechanics. In this test case, a strong vortex collides with a stationary shock wave characterized by a sudden increase in pressure. This collision generates a range of dynamic phenomena, including shock wave reflections, vortex structures, and severe pressure gradients. As a result, the interaction between these two entities causes complex disturbances in the velocity and pressure fields, as well as significant density variations.

When the strong vortex encounters the powerful shock wave, it leads to a significant deformation of the shock wave configuration, resulting in the formation and propagation of both linear and nonlinear waves downstream in the flow. Two distinct physical phenomena are observable in this context. First, the strong vortex is split into two distinct vortex structures due to the compression induced by the passage of the shock wave (Clearly, the post-shock vortex configuration depends greatly on the relative strength of the shock wave and the vortex). Second, a cylinder-shaped acoustic wave structure appears downstream of the stationary shock wave. Sound waves centered on the core of the moving vortex are partially interrupted by the shockwave. As a result, an alternation of expansion and compression zones becomes observable.

The first part of the chapter aims to present the problem, then to study the two processes of artificial viscosity injections separately, and finally, to compare these two methods of injection.

1   Problem description

The flow fields contain a stationnary shock with a strength of the shock (denoted M_s) equal to 1.5 and a strength of the vortex (denoted M_v) equal to 0.9. The problem domain is defined as $\Omega = (0,2) \times (0,1)$, with the stationary normal shock located at $x = 0.5$, and the vortex center initially positioned at $(x_c, y_c) = (0.25, 0.5)$ (see Fig. 4.1). The simulations were conducted using a Lax-Friedrichs flux, Gauss-Lobatto-Legendre shape functions, and a Linear- scaling factor. The time step used was fixed to $5\ 10^{-5}$ [s] and the final time is set to 0.7 [s]. The mesh consists of uniform quadrilateral elements.
The upstream flow is initialized as follows

$$\begin{bmatrix} \rho_u \\ u_u \\ v_u \\ p_u \end{bmatrix} = \begin{bmatrix} 1.0 \\ M_s\sqrt{\gamma} \\ 0.0 \\ 1.0 \end{bmatrix}. \tag{4.1}$$

All initial conditions can be determined solely from this information. More detailed calculations are provided in the appendices.
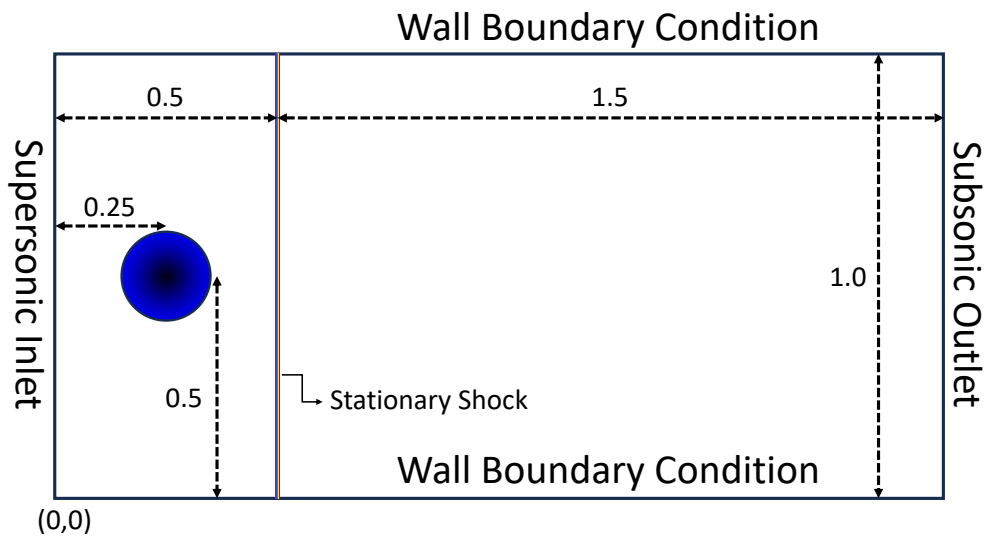
Figure 4.1: Schematic representation of the inviscid strong vortex-shock wave interaction test case.

The different sensor user parameters are briefly described below, along with those of the injection method.

Sensors

- The Persson Peraire sensor relies on Lagrangian filtering with a detection threshold $s_0 = -3 \log_{10}(\text{order})$ and an interval $\kappa = 1.5$. The choice of the detection threshold was made in accordance with the guidelines outlined in the referenced article [42].

- The Hennemann sensor does not use any user parameters.

- For the physical sensors (shear, shock and thermal), the detection thresholds are defined as follows: $s_{\min,\mu} = 1$, $s_{\min,\beta} = 0.01$, $s_{\min,\kappa} = 1$, with a maximum value $s_{\max} = 2$ for each of the sensors. The choice of the detection threshold was made in accordance with the guidelines outlined in the referenced article [2].

Injection Method

- Laplacian method uses a shock thickness $\delta = 0.5$ and a shock strength $M = 1.5$ (see Eq. 2.47).

- The Newtonian method employs the multiplicative constants $k_\mu = 1$ and $k_\beta = 0.1$ (see Eq. 2.48).

2   Laplacian Injection

In this section, the detailed application of Laplacian injection is explored, specifically in the context of inviscid interactions of vortex shocks. This segment aims to establish a solid foundation for the study of this particular test case. The focus will be exclusively on the impact of various regular sensors as well as the physical sensor. It is important to note that, unlike Newtonian injection which involves the use of three artificial quantities, Laplacian injection concentrates solely on one. Consequently, only the shock sensor, crucial in this scenario, will be employed.

2.1   Comparative Analysis of Sensor Effects

The comparison of various sensors and their impact on different quantities will be conducted using a 4th-order mesh with dimensions of 120×60, totaling 180 000 degrees of freedom.

This section will be divided into two subsections. The first will be a visual analysis to observe the behavior of artificial viscosity and the density field, with a focus on details using Schlieren imaging. The second subsection will examine integrated quantities over the volume, such as artificial viscosity, kinetic energy, and others.

2.1.1   Visual analysis

Density Fields

Fig. 4.2 presents the density field at a time $t = 0.7$ [s] for the various sensors.

The resolution achieved with the PerssonPeraire sensor is found to be less accurate. While it is capable of capturing vortex separation, its precision falls short when compared to the other two sensors. As for the regular Hennemann sensor, its performance closely mirrors that of the physical sensor, though it appears to be slightly less precise. Additionally, it introduces extra structures which might be considered as numerical artifacts.

The physical shock sensor, despite not being entirely suitable for use with a Laplacian injection viscosity method due to its inability to handle shear or thermal issues, still appears to yield better results under these specific conditions.

(a) PerssonPeraire sensor

(b) Hennemann sensor

(c) Physical sensor

Figure 4.2: Density fields at time $t = 0.7$ [s] obtained through an artificial viscosity method with Laplacian injection for the PerssonPeraire sensor, the physical sensor and Hennemann sensor. The simulation is conducted on a mesh 120×60 using a fourth-order scheme.

Artificial viscosity

It is now relevant to focus on the analysis of the detection of disturbed cells and the integration of viscosity within them. This study aims to evaluate the behavior of various detectors and to confirm whether artificial viscosity is correctly applied to the disturbed cells.

For this study, two critical moments have been selected. The first, at t $= 0.23$ [s], corresponds to the moment when the vortex completely crosses the shock. The objective here is to spatially analyze the sensors to see if they adequately adjust to the altered shape of the shock caused by the passage of the vortex. The second important moment is at t $= 0.7$ [s], the end of the simulation. At this stage, the shock should have almost returned to its original configuration, providing an opportunity to observe the behavior of the artificial viscosity in the absence of shock disturbances by the vortex.

Analysis at t $= 0.23$ [s]

Fig. 4.3 displays the disturbed cells and the injected artificial viscosity. A positive observation is that all sensors successfully capture the shock deformation and accordingly inject artificial viscosity.

It should also be noted that while regular sensors inherently detect cell disturbances, the injection of artificial viscosity is not quite the same. Indeed, artificial viscosity uses a quantity (see the

(a) Troubled cells (PerssonPeraire sensor)

(b) Artificial viscosity (PerssonPeraire sensor)

(c) Troubled cells (Hennemann sensor)

(d) Artificial viscosity (Hennemann sensor)

(e) Troubled cells (Physical sensor)

(f) Artificial viscosity (Physical sensor)

Figure 4.3: Regions of troubled cells and artificial viscosity fields at time $t = 0.23$ [s], obtained through an artificial viscosity method with Laplacian injection for the PerssonPeraire sensor, the physical sensor and the Hennemann sensor. The simulation is conducted on a 120×60 mesh using a fourth-order scheme.

scaling viscosity λ in Eq. 2.44) calculated at each point, potentially leading to minor variations within a cell. The decision to use a point-based quantity rather than a constant one, by taking the maximum value for instance, aims to enhance the precision of the Laplacian viscosity method slightly. In the case presented here, this improvement does not significantly impact the results and is not particularly relevant to the discussion but still merits mention.

The PerssonPeraire sensor significantly detects a higher number of troubled cells. Contrary to the preconceived idea that more extensive detection could lead to reduced detection intensity per cell, especially with the injection of artificial viscosity in the vicinity, this hypothesis does not hold true in this case. Indeed, both Hennemann and physical sensors show detection intensity equivalent to that of PerssonPeraire while reporting a lower number of disturbed cells or points.

The physical sensor enables more accurate intracellular detection of disturbed regions, but this comes with the addition of significantly higher artificial viscosity in that area, risking excessive discontinuity in the field of artificial viscosity. While this issue is mitigated by the smoothing of the Laplacian injection, it is entirely legitimate to have heightened concerns regarding Newtonian injection.

The Hennemann sensor provides an efficient illustration of the shock deformation with the vortex. Although it achieves optimal detection, it is important to note that this sensor has a maximum detection intensity set at 0.5 by definition, while the others can go up to 1. In this particular context, this characteristic seems to align with the trends observed by the other two sensors. However, it remains to be determined whether this peculiarity is intentional or if the sensor is already saturated. It will be observed later that, with the Newtonian injection, this approach does not allow for the successful completion of the simulation.

Analysis at t = 0.7 [s]

Fig. 4.4 presents the detections according to the different sensors for this time.
Similarly to the previously examined moment, many cells are disturbed at the shock, forming a detection line strongly centered on the shock with slight extensions on the sides. There is also weak detection associated with structures resulting from the interaction between the shock and the vortex for PerssonPeraire sensor. This clearly appears to be a drawback of this sensor in this context. Indeed, these structures are not disruptive enough to crash the simulation, as demonstrated by the other two sensors.

Regarding the physical sensor, its detection is much finer than the others. Its true added value lies in its pointWise nature and, therefore, its ability to provide a more realistic view of detection and destabilizing structures, such as the shock. For example, it reveals that the shape of the shock has not fully returned to its initial configuration, still exhibiting slight deformations.
Additionally, similar to the PerssonPeraire sensor, it has very weak detection of acoustic waves near the shock, which, once again, are not destabilizing and indicate slight sensor calibration inaccuracies. Although this is completely negligible in this test case, other test cases are more sensitive to artificial viscosity and could yield entirely different results (Instabilities of Kelvin-Helmholtz test case). However, this detection remains entirely correct and consistent.
It is also noteworthy that the sensor does not detect the vortex at all due to its association with an anti-vorticity sensor.
Finally, it is interesting to note that the sensor identifies spurious oscillations near the shock, sometimes with relatively high intensity. This is a positive aspect for this sensor, highlighting its role in stabilizing these destabilizing oscillations for the shock. This observation also explains why the PerssonPeraire sensor has a rather wide thickness compared to the shock since it captures the spurious oscillations in addition to the shock.
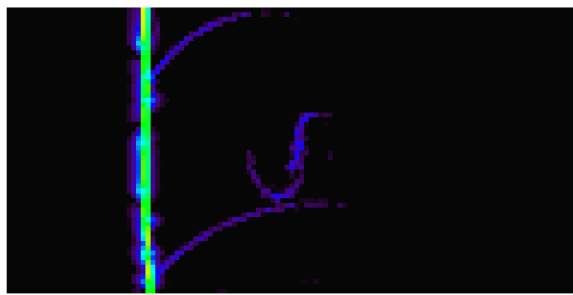
(a) Troubled cells (PerssonPeraire sensor)
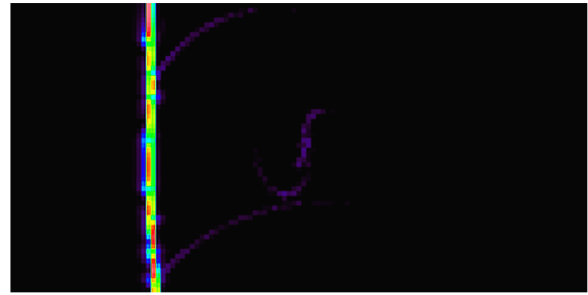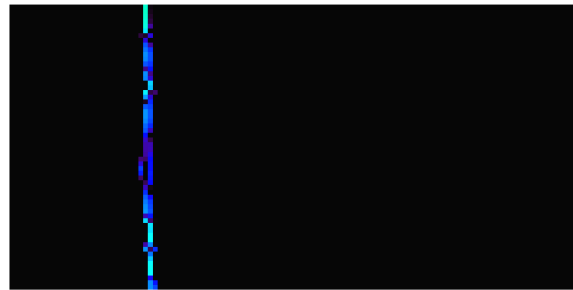
(b) Artificial viscosity (PerssonPeraire sensor)

(c) Troubled cells (Hennemann sensor)
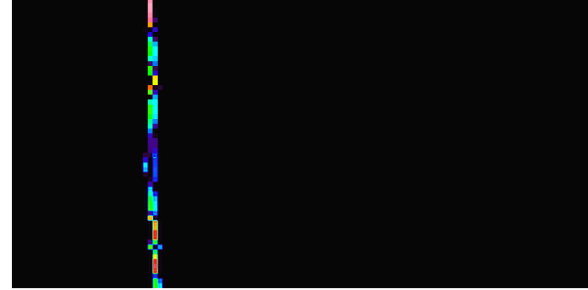
(d) Artificial viscosity (Hennemann sensor)
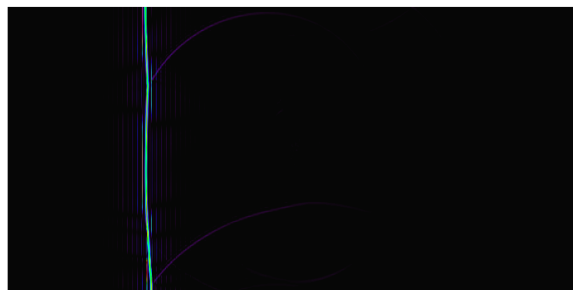
(e) Troubled cells (Physical sensor)
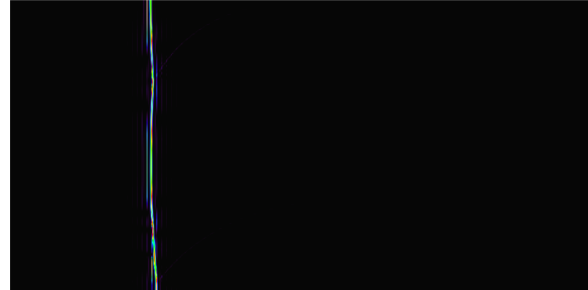
(f) Artificial viscosity (Physical sensor)
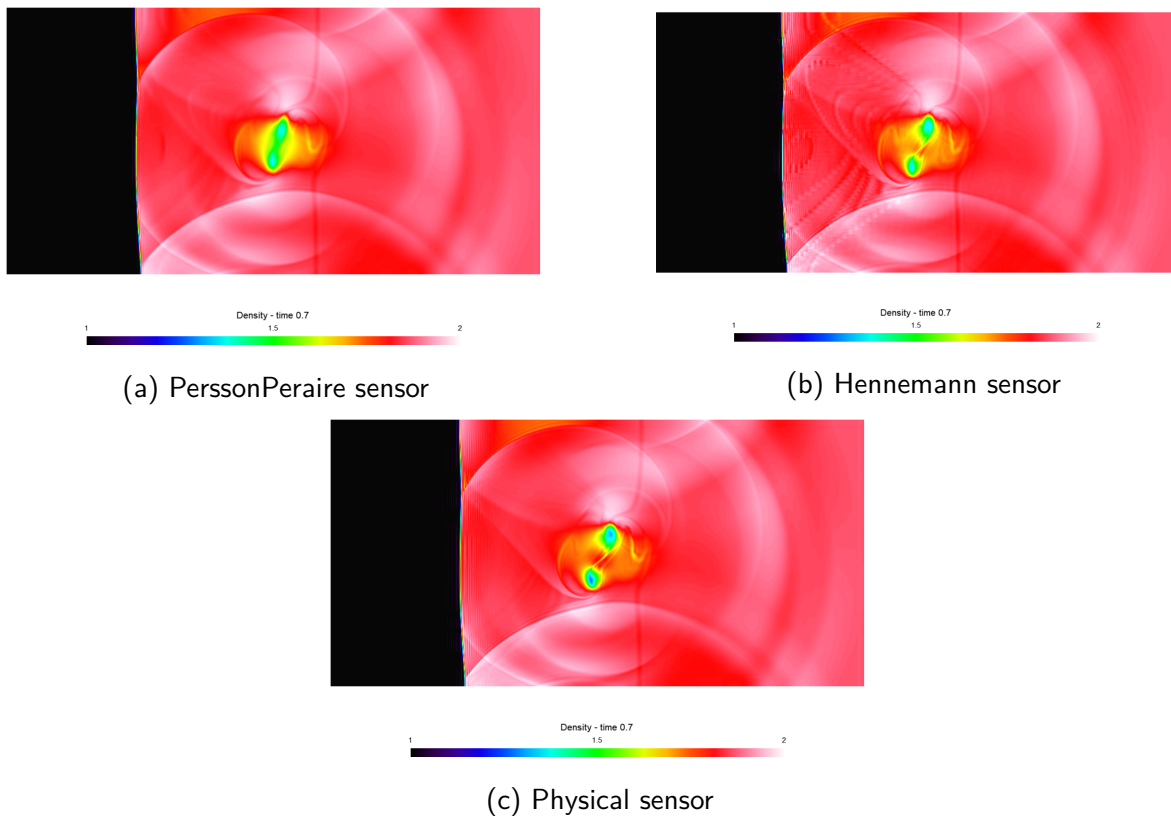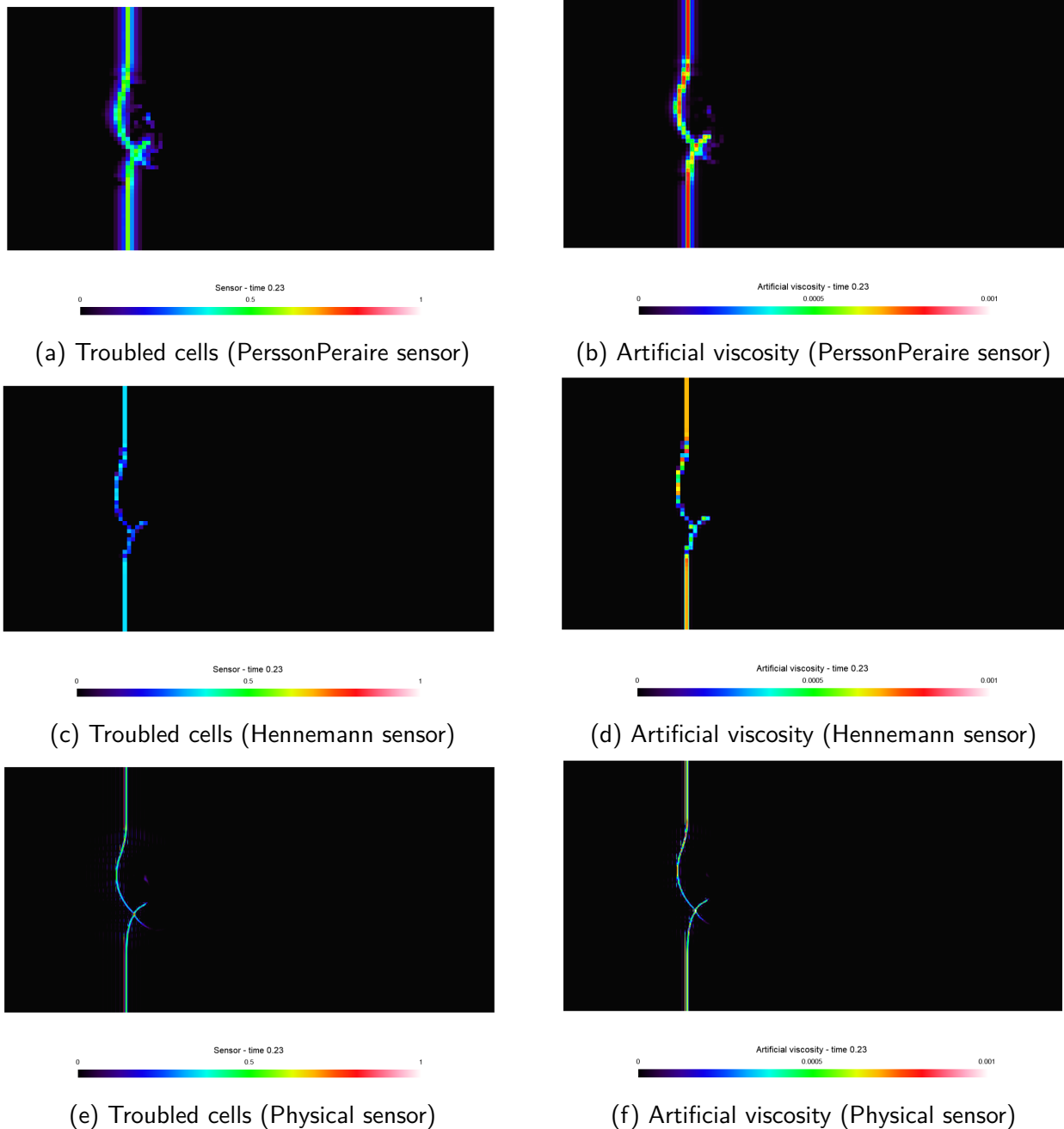
Figure 4.4: Regions of troubled cells and artificial viscosity fields at time $t = 0.7$ [s], obtained through an artificial viscosity method with Laplacian injection for the PerssonPeraire sensor, the physical sensor, and the Hennemann sensor. The simulation is conducted on a 120×60 mesh using a fourth-order scheme.

The Hennemann sensor provides sufficient detection to stabilize the simulation. Its detection intensity is relatively lower than that of the physical sensor, even though the latter should display a higher intensity due to its PointWise nature. This sensor contributes to the stability of the simulation but is stingy in terms of detection and, consequently, artificial viscosity, which can lead to difficulties for other configurations (change of test case, user parameters, shock intensities and others).

Numerical schlieren

The numerical schlieren is proving to be an exceptional tool for the analysis of the strong vortex test case. It highlights the structures resulting from shock-vortex interaction, as well as shock-related oscillations. This tool is widely used in the literature to better visualize shocks, which justifies the importance of its visualization and comparison. Fig. 4.5 presents a reference schlieren [45] as well as schlieren generated by various sensors using the Laplacian injection method.

(a) Reference

(b) PerssonPeraire sensor

(c) Hennemann sensor

(d) Physical sensor

Figure 4.5: Numerical schlieren at time $t = 0.7$ [s], obtained through an artificial viscosity method with Laplacian injection for the PerssonPeraire sensor, the Hennemann sensor and the physical sensor. The simulation is conducted on a 120×60 mesh using a fourth-order scheme.

It is important to note that the interest here lies in the differences observed when using different sensors, rather than exploring other aspects such as choice of numerical flux, mesh type, or grid resolution. It is highly likely that the various criteria introduce artifacts. Nevertheless, no criticism will be addressed to them if all figures include them.

The use of high artificial viscosity is evident in the schlieren obtained with the PerssonPeraire sensor. This sensor exhibits fewer oscillations and less pronounced intensities. It also provides adequate visualization of most phenomena, even though they are less contrasted (instabilities observed near the vortex).

The other two sensors exhibit a greater number and higher intensity of oscillations, but they offer better precision. It is important to note that there is a trade-off between attenuating these spurious oscillations and the accuracy of the simulations.

For the Hennemann sensor, structures between the shock and the vortex are observed with much higher intensity than in the other two sensors. These structures are commonly observed in many shock-capturing methods [46, 47]. Comparison with the reference indicates that these structures are numerical artifacts, thus penalizing the Hennemann sensor for their amplification.

The physical sensor also exhibits these same artifacts, but to a lesser extent, and it fairly represents the reference.

Density profile

Schlieren has revealed the presence of very pronounced density oscillations, especially for sensors injecting the least amount of viscosity, namely Hennemann and the physical sensor. To analyze these oscillations in more detail, a graph was generated by taking a cut in the density domain equivalent to $y = 0.4 + \varepsilon$, where $\varepsilon^1 = 10^{-4}$. This location was chosen to establish a basis for comparison with a reference obtained in the literature [48].

The graph in Fig. 4.6 highlights excessive fluctuations near the shock, with much higher values observed for the Hennemann sensor. This trend can be attributed to its economy in terms of detection and injection of artificial viscosity. In this context, the physical sensor manages the oscillations better as it detects them at the sensor level. Although some oscillations persist, they are somewhat attenuated but appear quite early. On the other hand, the PerssonPeraire sensor injects a significantly higher amount of artificial viscosity, and therefore shows much fewer oscillations than the other two, with their appearance occurring later than the other two, which is a notable advantage.

Other oscillations occur a little later around $x = 0.8$, before entering a region where the density decreases abruptly, marking the entry into the core of one of the two vortices. It is worth noting that the curves corresponding to the physical and Hennemann sensors come very close to the reference solution but do not quite reach it. As for PerssonPeraire, it encounters more difficulties and stops well before, which is normal given that the method used with this sensor is not precise enough at this resolution.

Summary of visual analysis

The sensors successfully detect the problematic shock and continuously inject artificial viscosity into it. Various observation moments have confirmed that the artificial viscosity is correctly injected at the shock location, and even as the shock deforms, the artificial viscosity faithfully follows it.

The PerssonPeraire sensor detects the highest number of disturbed cells and injects artificial viscosity of a similar magnitude as the other sensors, but in a more widespread manner. This spread-out detection results in greater numerical dissipation and limits its ability to accurately replicate the complex separation of the vortex. However, it should be noted that the oscillations

---

[1]This value was chosen to avoid taking density only at the edges of the elements, considering the regularity of the quadrangular mesh.

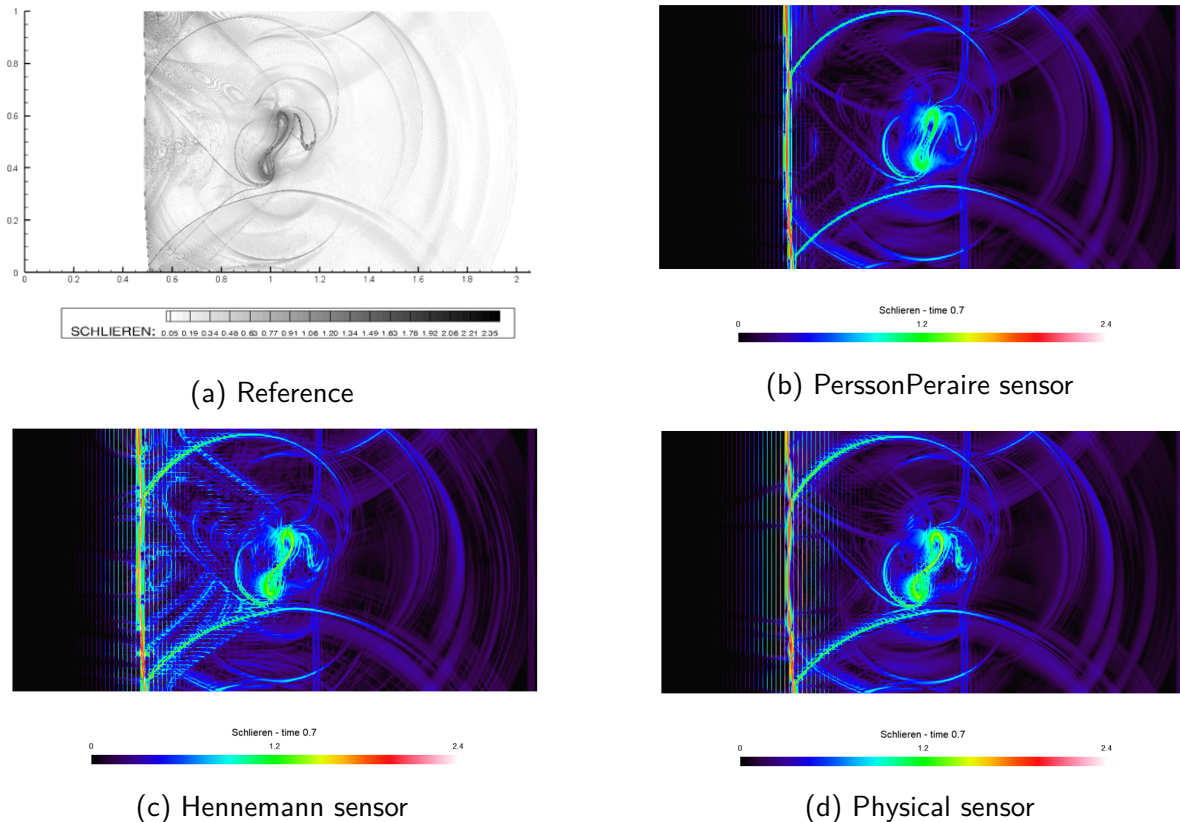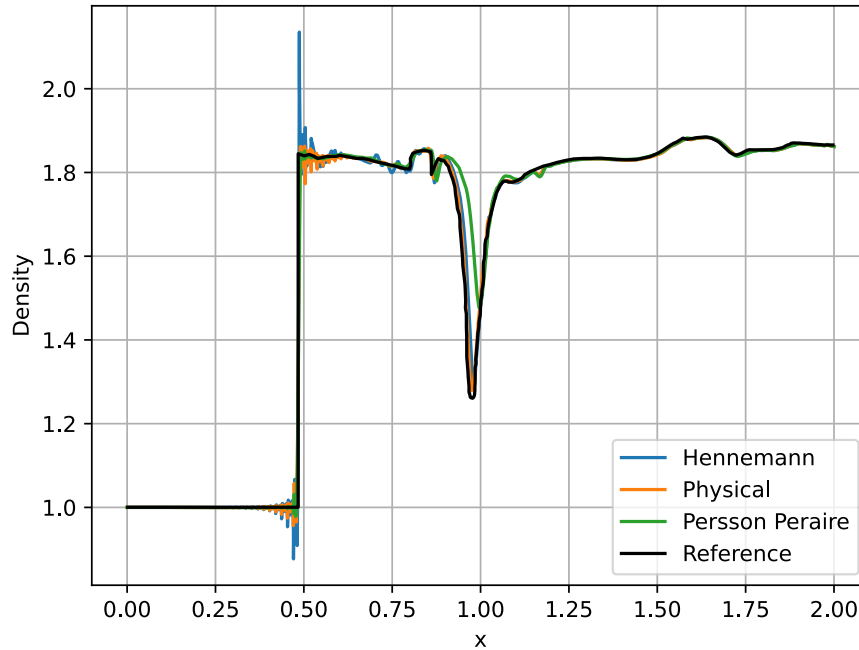Figure 4.6: Density profile along $y = 0.4$ at time $t = 0.7$ [s] obtained through an artificial viscosity method with Laplacian injection for the PerssonPeraire sensor, the Hennemann sensor and the Physical sensor. The simulation is conducted on a 120×60 mesh using a fourth-order scheme.

associated with the shock are most attenuated with this sensor, indicating that a larger amount of artificial viscosity reduces oscillations but affects the accuracy of the results.

On the other hand, the Hennemann sensor detects fewer disturbed cells and injects a viscosity better adapted to the situation. It provides better results than the PerssonPeraire sensor, although it still does not match the precision of the physical sensor. It also presents more significant oscillations and structures not seen in the other figures and the reference.

The physical shock sensor offers the best result, although it is not entirely suitable for the method. Its pointwise nature allows it to detect, with superior precision, even slight deformations of the stationary shock in the flow. Its detection is not perfect and may lead to the identification of non-destabilizing structures. Nevertheless, thanks to its pointwise nature and the use of a natural filter (the use of Sinus in Equation Eq. 2.44), these erroneous detections are minimized and become less noticeable.

The density profile observed along $y = 0.4$ reveals that these oscillations appear quickly and take time to fade, although their intensity is less pronounced than that observed with the Hennemann sensor. However, this result should be interpreted cautiously, as it represents only a cross-section at a specific location and time. The oscillations may be more intense at other moments or in other places.

The referenced solution indicates that, even though the mesh is not fine enough to allow for a precise comparison regarding the validity of the figures, the representation obtained with the

physical shock sensor is nevertheless quite faithful.

2.1.2   Quantitative analysis

Artificial viscosity

Fig. 4.7 illustrates the integration of artificial viscosity over the volume for different sensors. Each curve remains relatively stable over time, which is consistent with expectations, as the major disturbance in this case is the stationary shock. Therefore, only this part of the domain should receive additional viscosity, as shown in Fig. 4.3 and Fig. 4.4.
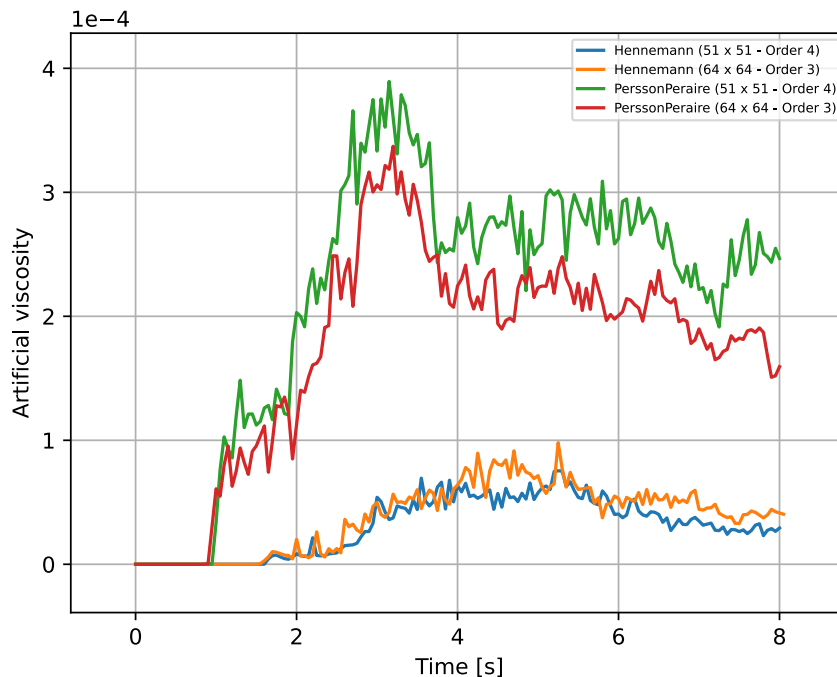
Figure 4.7: Artificial viscositiy integrated over the volume obtained through an artificial viscosity method with Laplacian injection for the PerssonPeraire sensor, the Hennemann sensor and the Physical sensor. The simulation is conducted on a 120×60 mesh using a fourth-order scheme.

The PerssonPeraire sensor injects more artificial viscosity overall due to its detection of larger disturbed areas. Furthermore, the slight increase in the curve comes from the detection of other non-destabilizing structures.

As for the Hennemann sensor, it exhibits higher viscosity than the physical sensor, which is explained by its elementWise nature. This nature also partially explains the more frequent variations observed with regular sensors compared to the physical sensor. Indeed, a constant sensor means that if it detects a disturbed area, an entire cell will inject a quantity of artificial viscosity. The second reason stems from the fact that it depends on the regularity of the solution, which is more prone to variation over time than the speed divergence used by the physical sensor. Since this divergence remains largely unchanged over time, as it is mainly induced by the shock, it is consistent that the viscosity curve associated with this sensor remains constant.

Kinetic energy and enstrophy

Fig. 4.8 presents data related to kinetic energy and enstrophy in the simulation. An initial disturbance due to the shock generation at the beginning of the simulation affected the kinetic energy results, causing a decrease. This initial error complicates the comparative analysis between cases with and without shock capture. However, a general trend is still observable: kinetic energy increases as soon as the vortex enters the shock.

(a) Kinetic energy   (b) Enstrophy
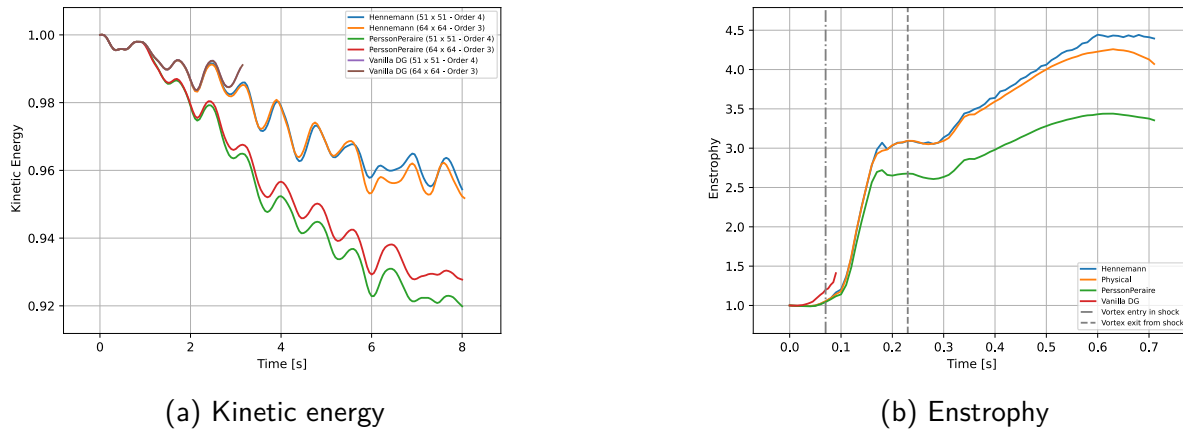
Figure 4.8: Dimensionless kinetic energy and enstrophy integrated over the volume obtained through an artificial viscosity method with Laplacian injection for the PerssonPeraire sensor, the Hennemann sensor and the Physical sensor. The simulation is conducted on a 120×60 mesh using a fourth-order scheme.

Until about halfway through the shock traversal, the curves from different simulations appear to overlap, regardless of the amount of artificial viscosity injected. However, after this midpoint, significant differences arise due to numerical dissipation, with an increasing gap between the curve generated by the PerssonPeraire sensor and those of the other two sensors. On the other hand, the curves of the latter two sensors seem to converge until the end of the simulation. This phenomenon is intriguing, especially since the Hennemann sensor injects twice as much artificial viscosity, but its impact appears to be attenuated.
The main explanation probably lies in the detection method itself. The Hennemann sensor only detects the shock, while the physical sensor also identifies emerging structures, albeit of low intensity. They still obtain numerical dissipation reducing the overall kinetic energy.

Regarding enstrophy, the impact of numerical dissipation induced by artificial viscosity is apparent from the beginning of the simulation. Increasing gaps also manifest around the halfway point of the shock traversal, for similar reasons to those of the kinetic energy. A major difference lies in the results of the Hennemann and physical sensors, which coincide for kinetic energy but diverge in the middle of the simulation regarding enstrophy. This suggests that the hypothesis of adding numerical dissipation in emerging structures detected only by the physical sensor is plausible.

Kinetic energy budget

Figure 4.9 illustrates the kinetic energy budget for each sensor. From the outset, the budget shows a discontinuity due to the stationary shock. Initially, the curves of the physical sensor exhibit a

general trend that is opposite to those of the regular sensors, similar to what is observed in the overall budget. A plausible hypothesis is that the pointWise nature of the physical sensor, coupled with its generally lower artificial viscosity injection compared to the other two, leads to a different calculation of gradients. Indeed, the curves of the regular sensors overlap in a consistent manner, which reinforces this assumption.

All sensors appear to be striving to achieve budget closure. Furthermore, even the physical sensor, starting from a lower point than the others, tries to adjust in order to reach this closure.
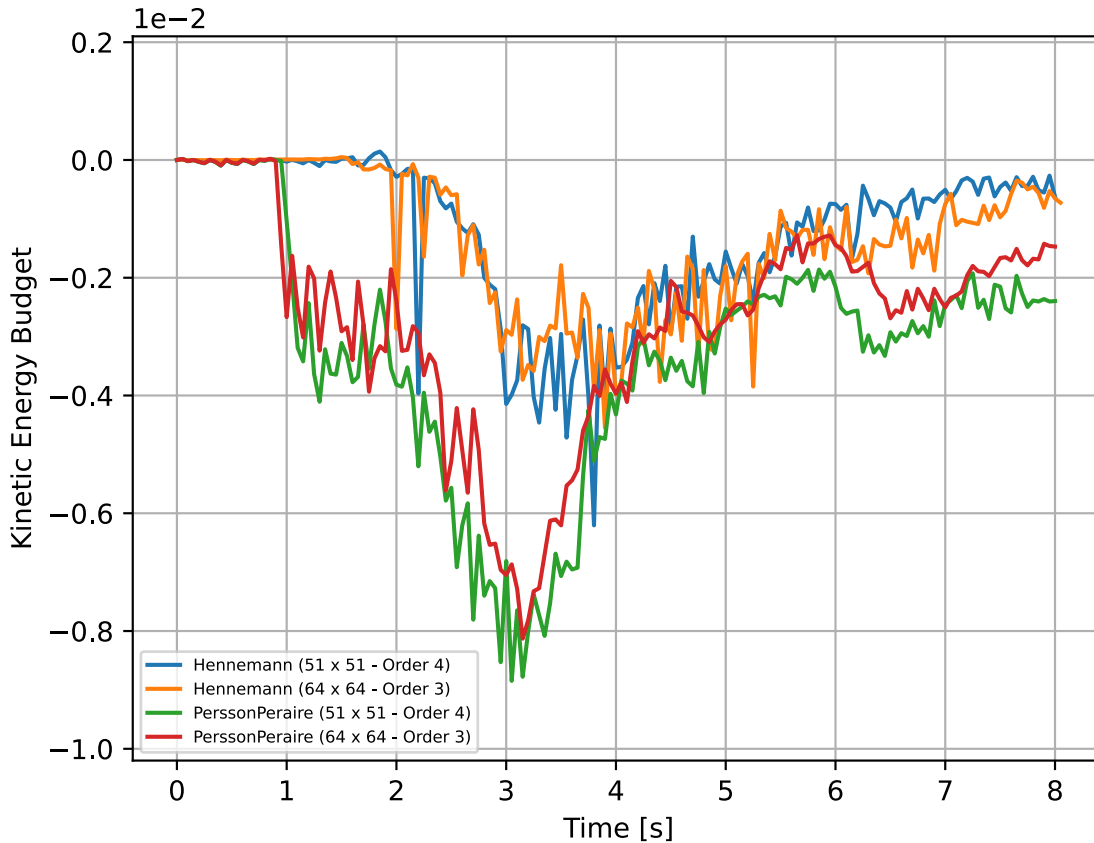
Figure 4.9: Kinetic energy budget obtained through an artificial viscosity method with Laplacian injection for the PerssonPeraire sensor, the Hennemann sensor and the Physical sensor. The simulation is conducted on a 120×60 mesh using a fourth-order scheme

2.1.3   Conclusion

In terms of pure visualization, the physical sensor is the one that most closely approximates reality, thereby offering enhanced precision. However, upon closer examination of the quantities, it is evident that although the Hennemann sensor injects a greater artificial quantity, the curves it produces are very close, if not identical, to those obtained with the physical sensor.

The oscillations for the Hennemann sensor appear more intense, though the Schlieren visualization suggests oscillations of similar intensity for the physical sensor. While these oscillations are present, they nonetheless allow for the successful progression of the simulation and contribute to the attempt to close the budget.

Finally, the PerssonPeraire sensor displayed the least favorable results due to its higher dissipation. However, it has shown that a more substantial addition of artificial viscosity leads to greater numerical dissipation but also to a reduction in these oscillations. With the current parameters, this sensor may not be ideal, but with recalibration, it could yield better performance.

3   Newtonian Injection

In the previous section, artificial viscosity was introduced using a Laplacian of the solution. This method has advantages such as robustness, but it also has drawbacks, notably the increase in dissipation due to this Laplacian.

Now, the goal is to explore a second method of injecting viscosity, this one being based more on a physical criterion. Unlike the previous method, this approach does not involve adding terms to the equations. The calculated and injected artificial viscosity will be directly combined with the existing physical viscosity, thus forming a resultant artificial viscosity.

This method makes use of three distinct terms: two viscosity terms, namely shear viscosity (μ) and bulk viscosity (β), as well as a third quantity, thermal conductivity (κ). In practice, it would be necessary to have three distinct sensors to measure these terms separately. However, in the context of this study, only two sensors are required since there is a relationship between the artificial shear viscosity and the artificial thermal conductivity (see Eq. 2.48).

Therefore, the use of physical sensors is the only option to fully take advantage of the Newtonian injection method. These sensors are inherently point-by-point and can be calculated in various ways.

Study structure

In the first subsection, different approaches will be examined to obtain the values of artificial viscosity at quadrature points, in order to determine the best method to apply in the following subsections.

The second subsection will focus on the analysis of the impact of the detection and application of artificial viscosity at the point level versus detection and application at the element level.

Finally, the third and last subsection aims to examine the influence of different sensors when using Laplacian injection. Although regular sensors are not inherently suited for the use of Newtonian injection, a proposal is to use the regular sensor as a shock sensor, while the other two sensors would remain inactive. The goal would then be to assess whether this approach is viable, to study the impact of the injection method on these sensors, and finally to compare the results obtained with those achieved using the physical sensor, in order to determine if the results are similar to those of Laplacian injection.

Each subsection will be divided into two distinct parts: the first part will consist of a visualization of the density profiles, numerical schlieren, detected areas, and areas where artificial viscosity is present. The second subsection, on the other hand, will focus more on the quantification of results using volumetric integrations, allowing for analysis of the behavior of the kinetic energy balance. All simulations in this section were conducted on a grid of dimension (120×60) with a precision of order 4.

3.1   Comparative Analysis of Interpolation Methods

The fundamental concept of deriving artificial viscosity at quadrature points through a type of interpolation using Bernstein polynomials is considered promising. This method tackles crucial issues like ensuring positive values, preventing the creation of new maxima in artificial viscosity, and introducing smoothing to artificial viscosity. However, it is noteworthy that Bernstein polynomials are not universally applicable in all scenarios. For instance, in the case of a 120×60 mesh of order 4, an approach employing Bernstein interpolation without interface continuity does not converge to a satisfactory solution. This aspect will be addressed in the quantitative analysis section.

3.1.1   Visual analysis

Density fields

Fig. 4.10 shows the density fields obtained using two different approaches: the direct approach and the approach using Bernstein interpolation while applying continuity at interfaces.
Both density fields are very similar, though the direct approach allows for a slightly better visualization. However, it reveals the presence of numerous oscillations near the shock, which propagate over a certain distance. These oscillations could pose a potential problem and it is plausible that they could lead to a crash if not carefully managed.
The attenuation of oscillations near the shock by an interpolation cannot be visually observed in the density profile due to these numerical artifacts intervening.
The approach using Bernstein polynomials results in a grid from the shock to the two vortices. This grid is clearly a numerical artifact due to the use of this interpolation.

(a) Direct approach   (b) Continuous Bernstein approach

Figure 4.10: Density fields at time $t = 0.7$ [s] obtained through an artificial viscosity method with Newtonian injection for the physical sensor. Two approaches are used, a direct approach and a continuous Bernstein approach. The simulation is conducted on a mesh of 120×60 using a fourth-order scheme.

Artificial viscosity

The visual representation of the artificial viscosity field is displayed at the interpolation points (which correspond to control points in this context), demonstrating the activity of the artificial viscosity. The influence of interpolation on the quadrature points is not visually depicted, owing

to challenges in accurately displaying them. While these images do not precisely show the artificial viscosity applied at the quadrature points, they do indicate the alterations caused by the artificial viscosity. Additional quantitative analysis is required for a deeper comprehension of the effects of interpolation.

Only the final time $t = 0.7$ [s] will be considered. The previous section, which evaluated the different sensors, has already sufficiently demonstrated that the physical shock sensor meets expectations. Fig. 4.11 shows the artificial shear and bulk viscosities for the different approaches.

(a) Artificial shear viscosity (Direct approach)

(b) Artificial bulk viscosity (Direct approach)

(c) Artificial shear viscosity
(Continus Bernstein Approach)

(d) Artificial bulk viscosity
(Continus Bernstein Approach)

Figure 4.11: Artificial shear and bulk viscosities at time $t = 0.7$ [s] obtained through an artificial viscosity method with Newtonian injection for the physical sensor. Two approaches are used, a direct approach and a continuous Bernstein approach. The simulation is conducted on a mesh of 120×60 using a fourth-order scheme.

The scale for shear viscosities has been reduced to illustrate their absence at the end of the simulation. This observation is consistent with expectations regarding the sensor and viscosity. Since the shock is the only source of disturbance in the tested case, the shear sensor should not be triggered.

Consequently, it is only the bulk viscosity, related to the velocity divergence and therefore to the shock, that contributes to stabilization. The discussion about the physical sensor is set aside, as it has already been sufficiently addressed in the context of Laplacian injection. However, it is worth noting that the detection exhibits strong behavioral similarity (not in intensity but in behavior) with that observed in the Laplacian injection.

Despite the fact that interpolation does not come into play in the figure, it is naturally observed that the artificial bulk viscosity is saturated for the Bernstein Continuous Approach (CBA), highlighting the fact that over the course of the simulation, a greater amount of artificial viscosity is injected due to this approach.

Schlieren

The numerical schlieren, shown in Fig. 4.12, clearly reveals the numerical anomalies previously observed in the density fields. It indicates that the grid-like appearance between the shock and the vortex is present in both methods, likely due to the use of Newtonian injection with physical sensors. However, this effect is more pronounced with the Bernstein polynomial approach.

One possible explanation for this amplification is that the Bernstein approach disperses viscosity throughout the cell, thereby reducing the intensity of artificial viscosity at quadrature points compared to interpolation points. However, at the quadrature points of interfaces, the intensity of artificial viscosity remains identical to that calculated at interpolation points, leading to a reduction in the intensity of artificial viscosity only at the volume quadrature points. This might explain why a grid-like pattern appears over the domain after the vortex has passed through the shock.

A second reason could be that the use of simple C_0 continuity at interfaces is insufficient, and it might be necessary to increase it.

(a) Direct approach   (b) Continuous Bernstein approach

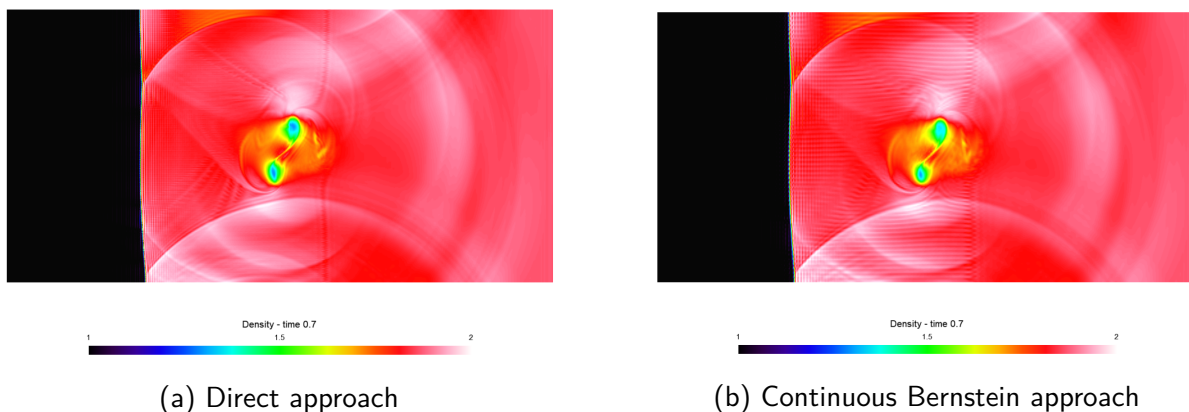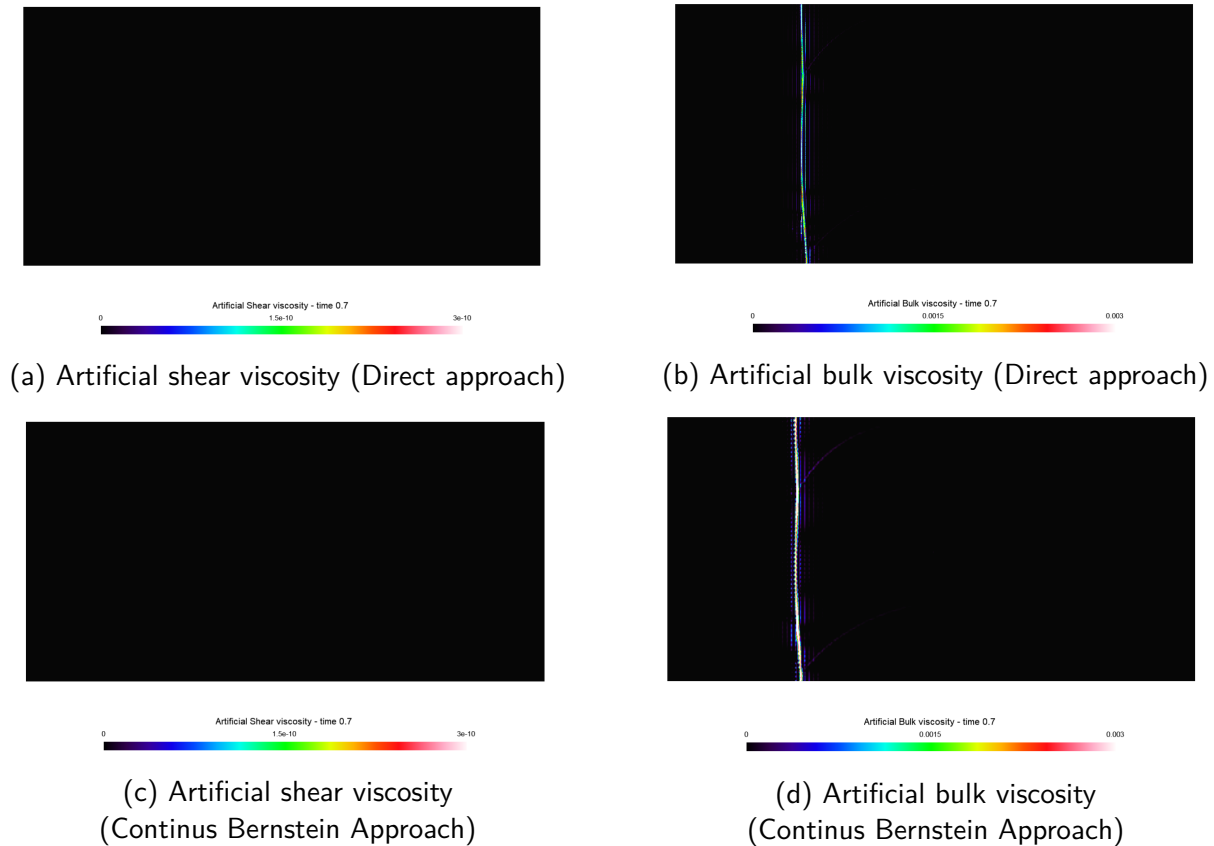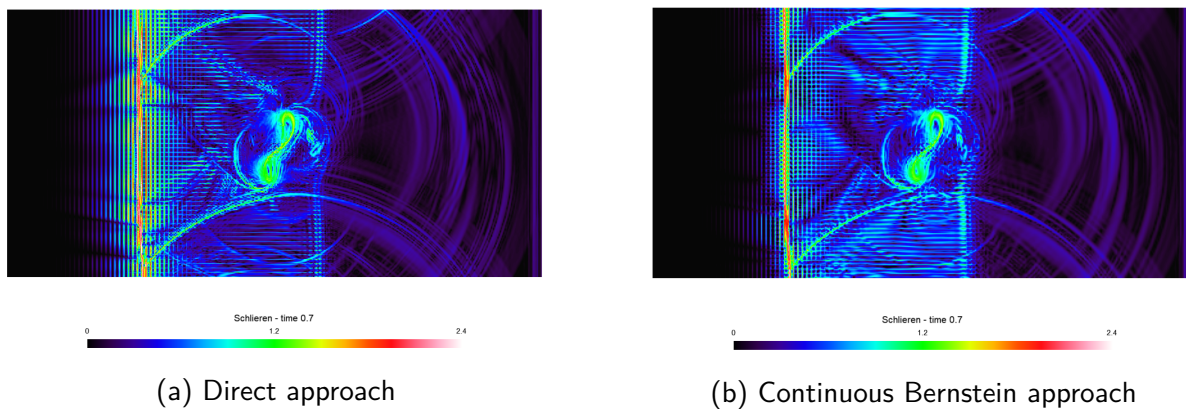Figure 4.12: Numerical schlieren at time $t = 0.7$ [s] obtained through an artificial viscosity method with Newtonian injection for the physical sensor. Two approaches are used, a direct approach and a continuous Bernstein approach. The simulation is conducted on a mesh of 120×60 using a fourth-order scheme.

Density profile

Fig. 4.13 demonstrates that the use of Bernstein polynomials succeeds in reducing the intensity of oscillations near the shock. This observation suggests that the utilization of these polynomials is prudent, and with some adjustments, this method could offer improved performance.
Apart from this aspect, the two methods examined appear relatively similar, but they fail to fully capture the maximum of the density well.
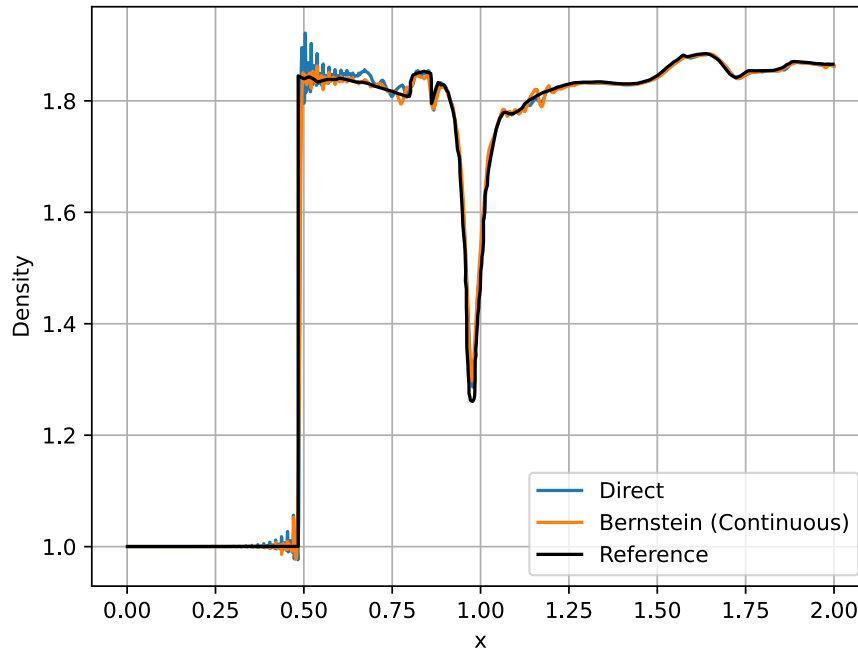
Figure 4.13: Density profile along $y = 0.4$ obtained through an artificial viscosity method with Newtonian injection for the physical sensor. Two approaches are used, a direct approach and a continuous Bernstein approach. The simulation is conducted on a mesh of 120×60 using a fourth-order scheme.

3.1.2   Quantitative analysis

Artificial viscosity

Fig. 4.14 illustrates the shear and bulk artificial viscosities calculated over the entire volume during the simulation. The shear artificial viscosity is shown on a semi-log graph due to its low intensity in the direct approach.

The shear and, by extension, thermal artificial viscosities (linked by the equation Eq. 2.48) exhibit non-zero values in the Bernstein approaches when a vortex crosses the shock, suggesting that structures are impacted differently depending on the chosen approach. This observation underscores the immediate and prolonged impact of adding artificial viscosity in a simulation. It indicates that one method of artificial viscosity will have distinct effects from another, altering

the evolution of the simulation.

The bulk artificial viscosity suggests that the Bernstein techniques increase the amount of artificial viscosity distributed per cell. In other words, through interpolation, artificial viscosity is not only injected at disturbed points but is spread over a larger number of points within the cell while maintaining a high intensity of viscosity. Although there is no creation of new maximum peaks of viscosity, this diffusion leads to an overall increase in the amount of viscosity present.
Finally, the Bernstein approaches, whether continuous or not, generate the same total amount of artificial viscosity. This also suggests that continuity at interfaces is likely more robust.

(a) Artificial shear viscosity   (b) Artificial bulk viscosity
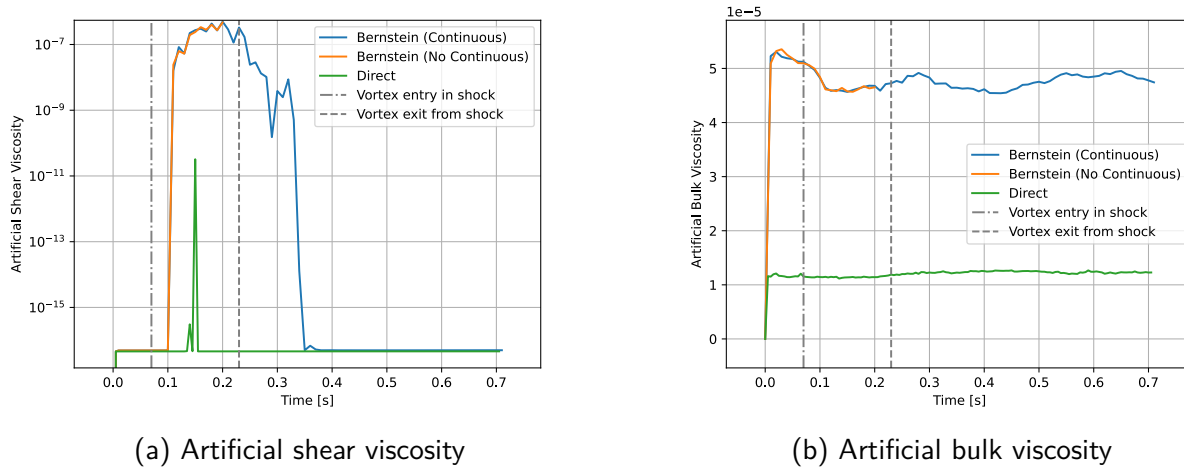
Figure 4.14: Artificial shear and bulk viscosities integrated over the volume obtained through an artificial viscosity method with Newtonian injection for the physical sensor. Two approaches are used, a direct approach and a continuous Bernstein approach. The simulation is conducted on a mesh of 120×60 using a fourth-order scheme.

Kinetic energy and enstrophy

Fig. 4.15 displays the kinetic energy as well as the dimensionless enstrophy integrated over the volume.
The observation that the kinetic energy remains unchanged, whether the continuous Bernstein approach is used or not, aligns with the notion that the same amount of artificial viscosity is injected in both cases, resulting in a similar impact in terms of numerical dissipation. This suggests that the differences in modeling methods with Bernstein do not significantly affect the overall level of kinetic energy.
Furthermore, the comparison with the direct approach, which injects a lesser amount of artificial viscosity, reveals the effect of numerical dissipation on the results.

The significant increase in enstrophy, particularly with the use of Bernstein polynomials and even more so when continuity is not accounted for at interfaces, highlights a key difference between the methods. The higher enstrophy in the Bernstein approaches, especially with discontinuity, might indicate increased instability, as suggested by the observation that the curve with the highest intensity is the one that exhibits stability issues.
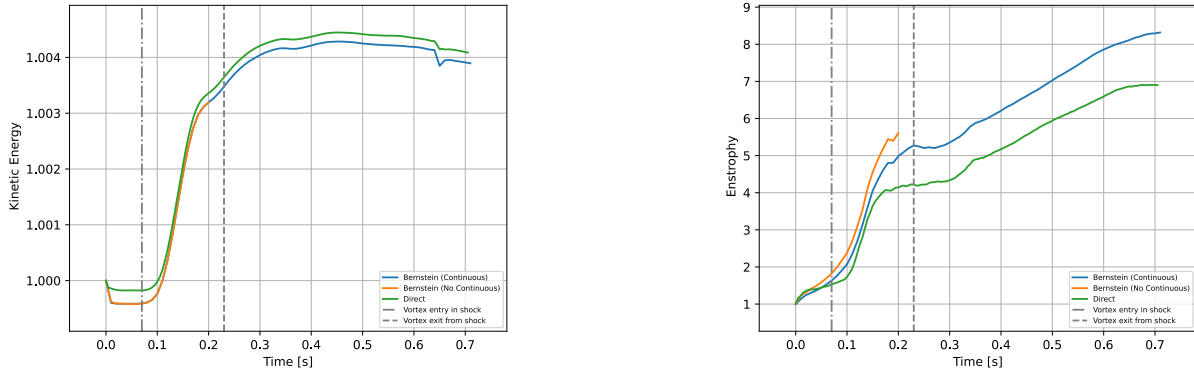
Figure 4.15: Dimensionless kinetic energy intregrated over the volume obtained through an artificial viscosity method with Newtonian injection for the physical sensor. The simulation is conducted on a mesh of 120×60 using a fourth-order scheme.

Kinetic energy budget

Fig. 4.16 shows that methods based on Bernstein polynomials fail to achieve budget closure, despite their use of increased artificial viscosity and their ability to attenuate Gibbs oscillations near the shock. This issue could be due to difficulties with interfaces, resulting from the specific interpolation of these polynomials.

Similar to the Laplacian injection using a physical shock sensor, the direct approach initially shows a deviation from budget closure. However, as the simulation progresses, this gap tends to diminish, and the approach almost reaches budget closure by the end of the simulation.
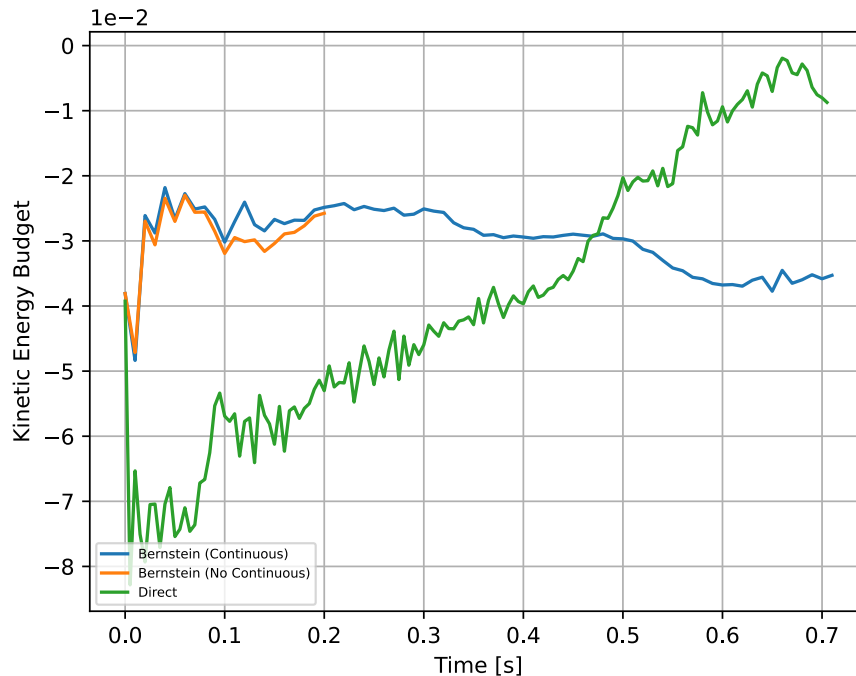
Figure 4.16: Kinetic energy budget obtained through an artificial viscosity method with Newtonian injection for the physical sensor. The simulation is conducted on a mesh of 120×60 using a fourth-order scheme.

3.1.3   Conclusion

The artificial shear viscosity is particularly low or even non-existent, which is a positive point since the simulation is supposed to be unstable only due to the shock. This shock is detected with as much precision as in the Laplacian method.

The impact of Bernstein interpolation becomes apparent when comparing viscosities, where those from Bernstein are higher. This significantly influences the results of the simulation. Additionally, this interpolation tends to create a grid pattern in the numerical Schlieren, which compromises the quality of the solution obtained in the simulation.

Furthermore, oscillations are more attenuated with the Bernstein approach. This suggests that this method could offer better stability and greater precision in simulations, especially in scenarios where shock management and disturbance reduction are crucial.

The Non Continuous Bernstein approach does not prove effective, suggesting that continuity contributes to the stability of the simulation. However, this conclusion should be interpreted cautiously, as the failure of this method in this specific case does not mean it would be ineffective in other configurations or with other parameters.

In summary, the Bernstein approach, with its unique characteristics, seems to offer advantages for this test case but needs further refinement to achieve better results.

3.2   PointWise and ElementWise Artificial Viscosity

This section is dedicated to exploring the impact of pointWise versus elementWise artificial viscosity, as well as their respective strengths and weaknesses. To concentrate the analysis, the study will be conducted on a single mesh resolution of (120 x 60) of order 4. It will also be done using the direct approach, so that the analyses are not disturbed by the effects of interpolation by Bernstein polynomials.

3.2.1   Visual analysis

Density fields

Fig. 4.17 depicts the visualization of density fields at the final moment, $t = 0.7$ [s].

(a) Elementwise   (b) Pointwise

Figure 4.17: Density fields captured at time $t = 0.7$ [s], using an elementwise artificial viscosity method and a pointwise artificial viscosity method. Both methods use a physical sensor of the same nature as the artificial viscosity. The simulation is conducted on a mesh 120×60 using a fourth order scheme.

It is observed that there are significantly more oscillations (point 1) in the pointWise density profile compared to the elementWise profile. This outcome is expected, as the use of elementWise artificial viscosity inherently acts as a basic form of smoothing, which naturally reduces oscillations. Points 2 and 4 highlight the presence of visible filaments in the pointWise figure that are absent in the elementWise figure. These filaments are also observed when using the Laplacian injection method with the Hennemann sensor.

Points 3 and 5 indicate changes near the vortex separation. Indeed, there is a slight improvement in precision with pointWise artificial viscosity, consistent with the notion that it introduces less dissipation and therefore provides a more accurate representation of reality.

Observations at points 3 and 5 reveal alterations near the vortex separation zone. This scenario is characterized by a subtle improvement in precision when using the pointwise artificial viscosity method. Due to its more refined detection capability, the pointwise artificial viscosity generates less dissipation, thereby leading to a more accurate representation of reality.

Artificial viscosities

Fig. 4.18 and Fig. 4.19 showcase the troubled cells detected by the physical sensors (elementWise and pointWise) and the artificial bulk viscosity injected into the simulation at the respective moments $t = 0.23$ [s] and $t = 0.7$ [s]. In the previous section, it was observed that the shear sensor detected very little in a direct approach. The use of an elementwise artificial viscosity method does not alter this fact, hence the discussion will not focus on artificial shear viscosity.

(a) Troubled cells (Elementwise)   (b) Artificial bulk viscosity (Elementwise)

(c) Troubled cells (Pointwise)   (d) Artificial bulk viscosity (Pointwise)
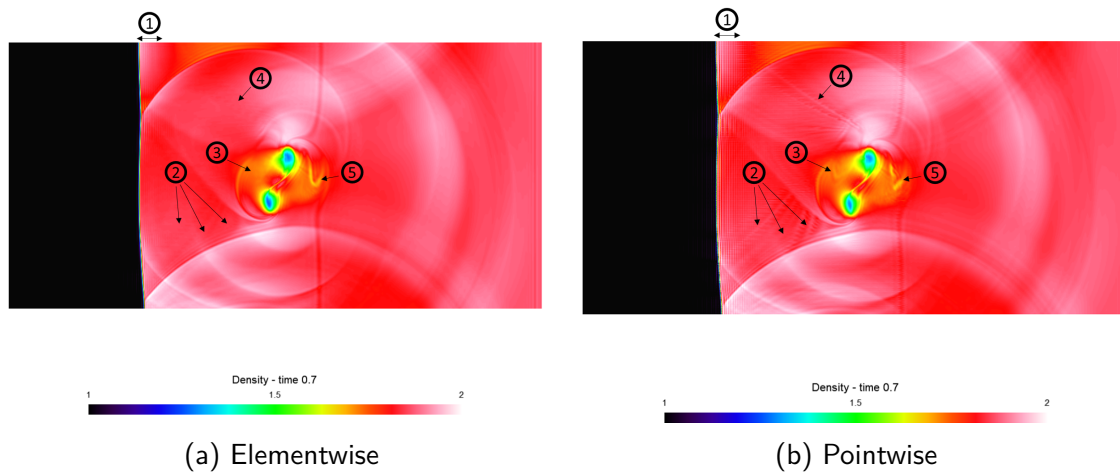
Figure 4.18: Regions of troubled cells and artificial bulk viscosity fields at time $t = 0.23$ [s], using an elementwise artificial viscosity method and a pointwise artificial viscosity method. Both methods use a physical sensor of the same nature as the artificial viscosity. The simulation is conducted on a mesh 120×60 using a fourth-order scheme.

An initial notable observation is that, from a detection standpoint, the elementWise figures seem to align with the pointWise detections.

As previously mentioned, the use of a physical sensor shows that it naturally detects oscillations near the shock with the aim of attenuating them. Regarding the elementWise sensor, the situation is a bit more complex. It can be noticed that, like at the first moment, it tends to capture almost all the oscillations behind the shock. However, at the second moment, only some oscillation fragments are detected. This variation can be attributed to two main reasons. First, by injecting a greater amount of artificial viscosity at the shock and its surroundings, it tends to reduce the oscillations over time. Secondly, there can sometimes be extremely low detection that is barely

(a) Troubled cells (Elementwise)

(b) Artificial bulk viscosity (Elementwise)

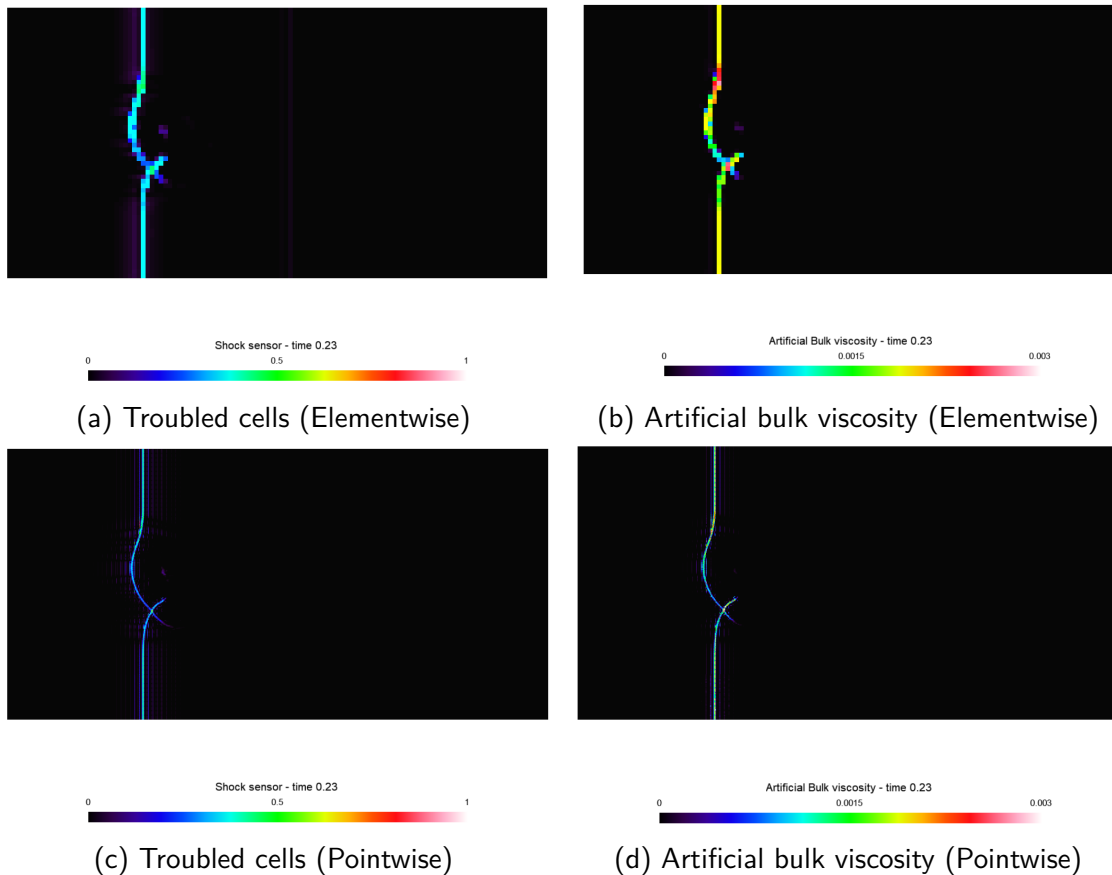(c) Troubled cells (Pointwise)

(d) Artificial bulk viscosity (Pointwise)
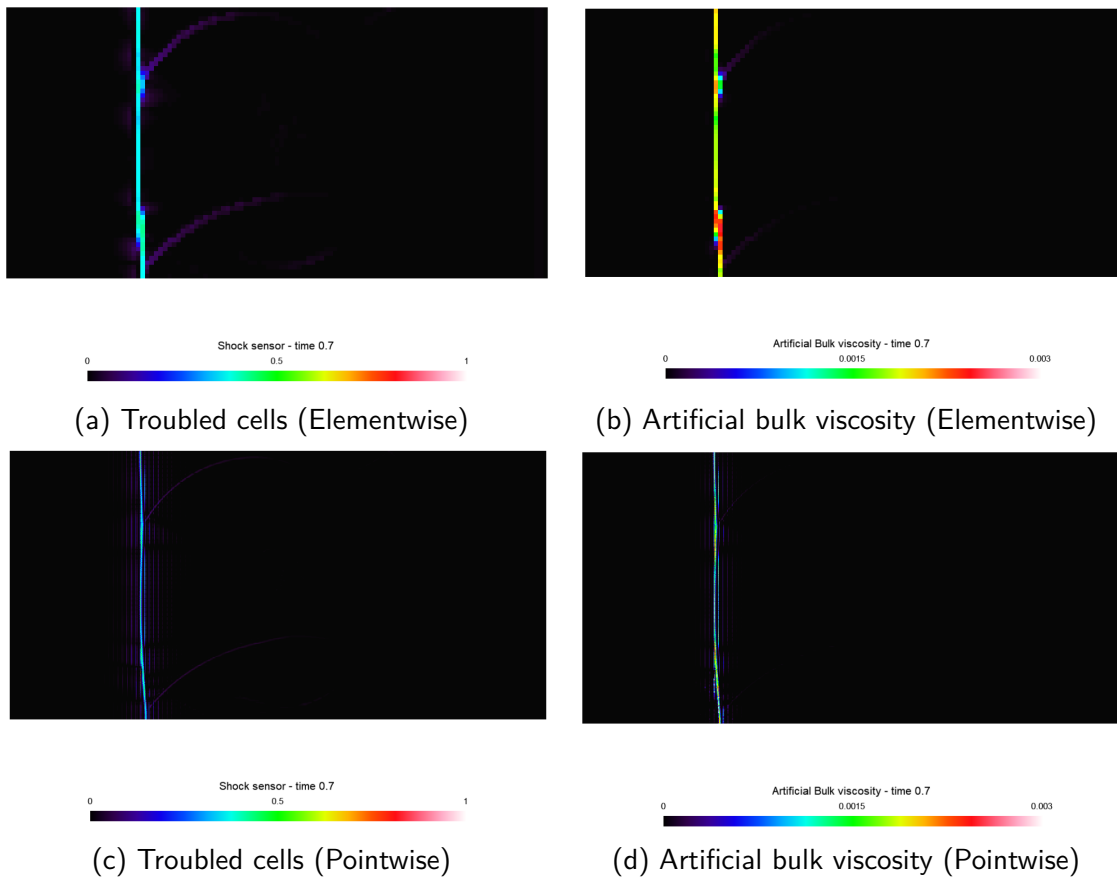
Figure 4.19: Regions of troubled cells and artificial viscosity fields at time $t = 0.7$ [s], using an elementwise artificial viscosity method and a pointwise artificial viscosity method. Both methods use a physical sensor of the same nature as the artificial viscosity. The simulation is conducted on a mesh 120×60 using a fourth-order scheme.

perceptible in the sensor's visualization.

The physical detector, regardless of its nature, also detects some structures that are not directly the shock and should not be detected by the sensor. Despite this detection, the injected artificial viscosity diminishes. This results from incorporating the sine function in equation Eq. 2.48, which facilitates a filtering effect. This effect reduces the viscosity when it is below half of the maximum detection threshold and increases it when it exceeds this halfway point. Therefore, for the elementWise artificial viscosity, it injects very little or no artificial viscosity for the oscillations compared to the pointWise.

Regarding intensity, it remains substantially in the same order of magnitude, with some areas exhibiting higher intensity for the pointWise while others are more pronounced for the elementWise. The figure at $t = 0.7$ [s] suggests a more pronounced intensity for the elementWise artificial viscosity, although more precise conclusions require additional information.
Overall, the intensity remains similar, with some areas more accentuated than others depending on the moment considered.

Numerical schlieren

The Schlieren visualization shown in Fig. 4.20 reveals a significant difference in intensity between the two types of artificial viscosity. It appears that there are fewer oscillations, but the real distinction lies in their intensity. It becomes clear that using a constant element-level artificial viscosity results in a drastic reduction in the intensity of these oscillations.

Schlieren - time 0.7
0   1.2   2.4

Schlieren - time 0.7
0   1.2   2.4

(a) Elementwise   (b) Pointwise

Figure 4.20: Numerical schlieren at time $t = 0.7$ [s], using an elementwise artificial viscosity method and a pointwise artificial viscosity method. Both methods use a physical sensor of the same nature as the artificial viscosity. The simulation is conducted on a mesh 120×60 using a fourth order scheme.

Additionally, there is a grid-like appearance in the pointwise case between the shock and the vortex separation, which is barely present in the elementwise scenario. The grid-like appearance is further emphasized by the pointwise nature of the artificial viscosity, highlighting discontinuities between elements and thus producing such a pattern. This phenomenon is not surprising since Schlieren visualization is based on the gradient of density.

Furthermore, most structures appear in both figures (points 2 and 4 in Figure Fig. 4.17), but they exhibit a significantly higher intensity in the pointwise case, indicating that density variations are stronger and may achieve a level of precision unmatched by the elementwise approach. This increased precision also leads to the emergence of new structures that were observed in Figure Fig 4.17 (points 3 and 5).

Density profile

It is relevant to perform a cross-section through the domain at $y = 0.4 + \varepsilon^2$ (see Fig. 4.21) to examine the oscillations near the shock and determine if this cross-section aligns with a reference [48].
In the figure, it is observable that, as noted in the Schlieren, the oscillations start very early in the case of pointwise viscosity and have a higher intensity. In contrast, for the elementwise viscosity, there are only a few oscillations which attenuate much more quickly.

$^2 \varepsilon = 10^{-4}$

Besides that, the curve associated with the elementwise viscosity appears to better match the reference solution, even reaching a density depth equal to that of the reference at the core of one of the vortices. However, at $x = 2$, there is an aberration caused by the reflection of the wave generated by the establishment of the shock, which induced the formation of the shock itself.

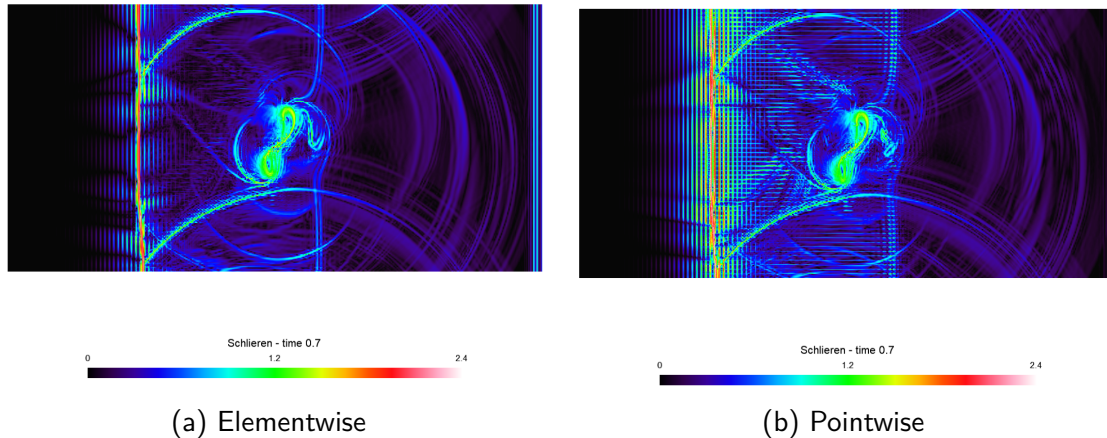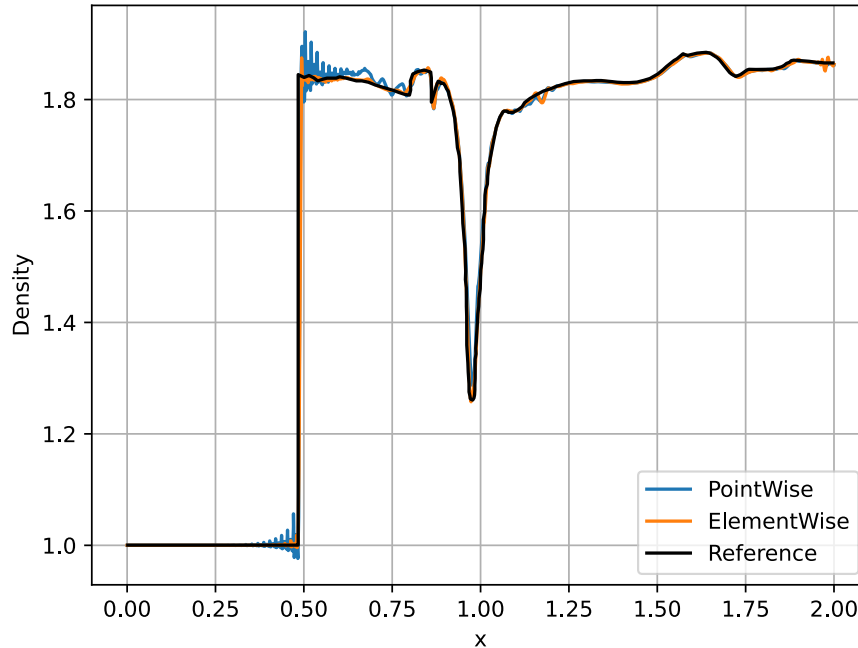Figure 4.21: Density profile along y $= 0.4$ at time t $= 0.7$ [s], using an elementwise artificial viscosity method and a pointwise artificial viscosity method. Both methods use a physical sensor of the same nature as the artificial viscosity. The simulation is conducted on a mesh 120×60 using a fourth order scheme.

Summary of visual analysis

For this mesh, the behavior of artificial viscosity and the observed quantities appear similar regardless of the nature of the artificial viscosity. Nevertheless, the visual analysis suggests that a greater quantity of small structures is generated through the pointwise detection and application of artificial viscosity. Several factors might explain this phenomenon. Firstly, the use of a constant artificial viscosity per element inevitably increases the numerical dissipation resulting from the addition of artificial viscosity. Secondly, it is possible that the sensors identify structures that do not necessarily destabilize the simulation. Although a natural filtering is applied during the calculation of artificial viscosity, some of these structures may persist. The approach of pointwise detection and injection of artificial viscosity can reduce the impact of these destabilizing structures on the simulation, thereby improving overall accuracy.

A drawback of the pointwise approach is the increased presence of Gibbs oscillations near shock areas, which can potentially lead to the failure of the simulation. These oscillations are significantly more marked and intense when using artificial viscosity applied in a pointwise manner. Adopting

a smoothing technique could be beneficial to mitigate these oscillations. This is illustrated by the element-by-element approach to artificial viscosity, which naturally smooths the solution.

3.2.2 Quantitative analysis

The graphs in Fig. 4.22 illustrate the integrated artificial bulk viscosity over the entire volume.

Figure 4.22: Artificial bulk viscosity integrated over the volume, using an elementwise artificial viscosity method and a pointwise artificial viscosity method. Both methods use a physical sensor of the same nature as the artificial viscosity. The simulation is conducted on a mesh 120×60 using a fourth order scheme.

The curve of the elementWise artificial bulk viscosity is approximately ten times higher than its pointWise counterpart. The primary reason for this lies in the elementWise nature of the detector. While a pointWise sensor would consider only a single point as disturbed, an elementWise sensor treats the entire cell as disturbed. As a result, the entire cell receives the same amount of artificial viscosity, leading to a notable increase in the total amount of artificial viscosity present in the simulation. The spread of artificial viscosity, meaning its propagation over a larger area when detection occurs, can explain this significant increase in elementWise artificial viscosity compared to pointWise viscosity. This also accounts for the increase in artificial viscosity observed after the passage of the vortex.

The pointWise artificial viscosity remains relatively constant throughout the simulation. This observation can be explained by the fact that a detector is associated with each quadrature point in the domain, meaning each point detects and injects its own intensity of artificial viscosity, enabling it to truly contour around destabilizing structures.

Kinetic energy and enstrophy

Fig. 4.23 presents the integrated dimensionless kinetic energy and enstrophy over the volume. Greater numerical dissipation is observed when the artificial viscosity is based on elements, which is expected given its much higher intensity.

For kinetic energy, there is a slight initial decrease due to the formation of the shock, and this decrease is less pronounced when the viscosity is pointWise. This already suggests an impact of artificial viscosity on kinetic energy, although it is an aberration.
During the simulation, it appears that when the vortex is within the shock, the element-based curve tends to converge towards the point-based curve. However, as soon as the vortex exits the shock, the gap between the two curves widens again.

Regarding enstrophy, a similar phenomenon occurs: a large initial gap followed by a smaller difference inside the shock. However, as soon as the vortex appears to exceed the shock, the gap between the curves widens again, reaching significantly higher values in terms of enstrophy.

(a) Kinetic energy   (b) Enstrophy

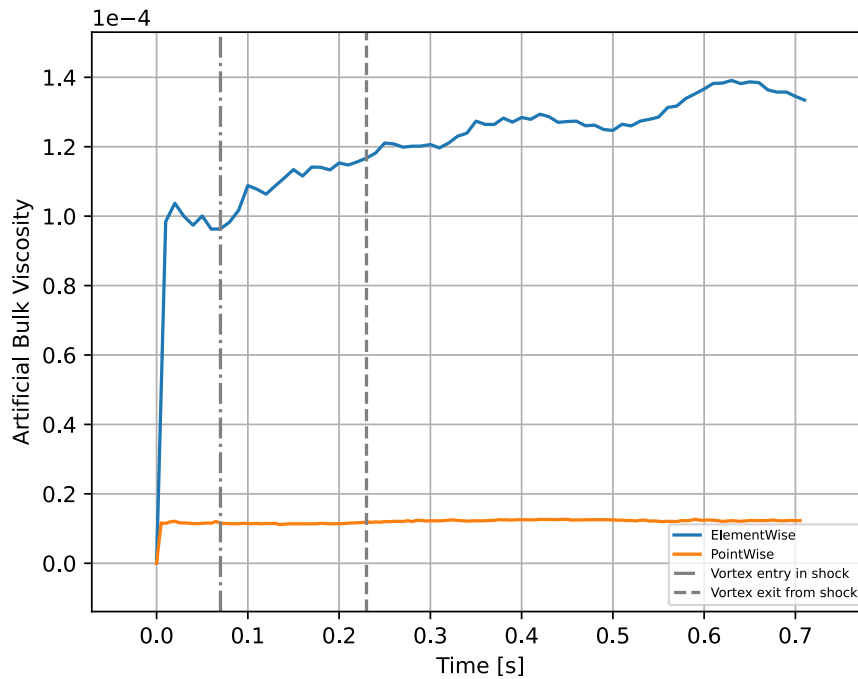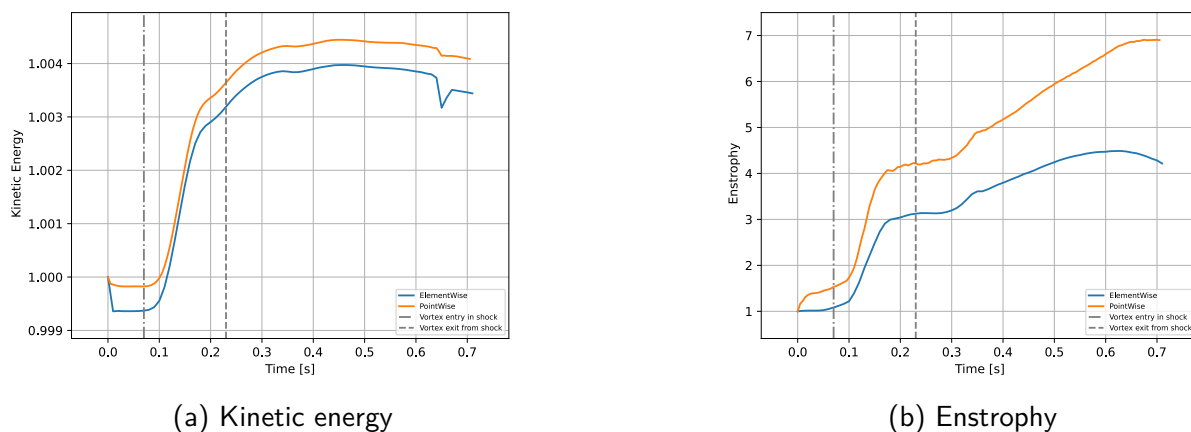Figure 4.23: Dimensionless kinetic energy and enstrophy integrated over the volume, using an elementwise artificial viscosity method and a pointwise artificial viscosity method . Both methods use a physical sensor of the same nature as the artificial viscosity. The simulation is conducted on a mesh 120×60 using a fourth order scheme.

Kinetic energy budget

The graph depicting the kinetic energy budget (Fig. 4.24) highlights two curves with distinct behaviors. At the start of the simulation, the pointWise method displays a trend that deviates more from the null curve, but as the simulation progresses, it tends to approach it. This evolution could be explained by the fact that the pointWise method introduces less numerical dissipation. This hypothesis is supported by the observation that in the elementWise approach, the budget remains constant, while there is an increase in numerical dissipation over time.
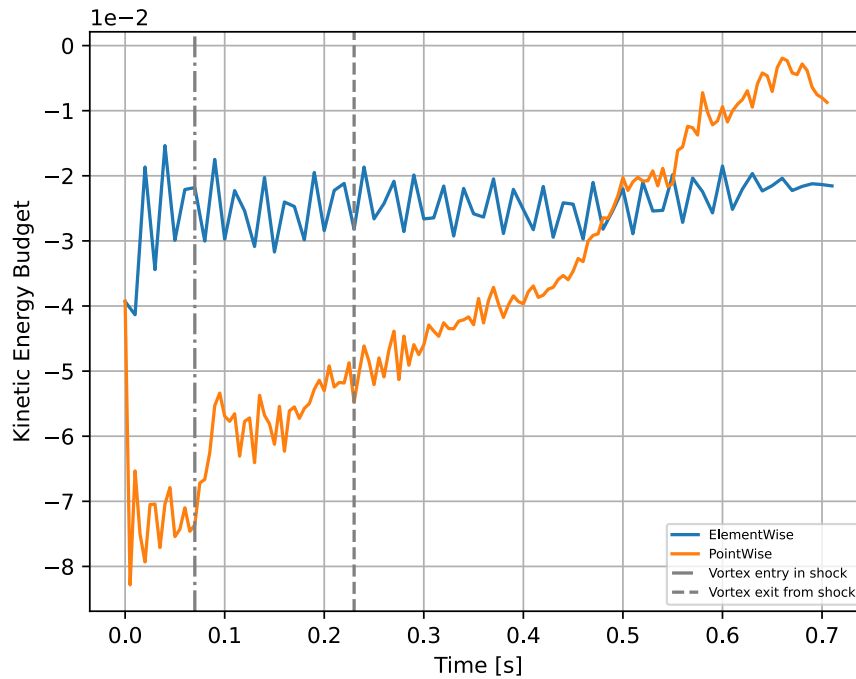
Figure 4.24: Dimensionless kinetic energy budget, using an elementwise artificial viscosity method and a pointwise artificial viscosity method. Both methods use a physical sensor of the same nature as the artificial viscosity. The simulation is conducted on a mesh 120×60 using a fourth order scheme.

3.2.3 Conclusion

Visually, both natures provide good results, better than those of the Laplacian injection with regular sensors. The perceived differences between the two approaches are also very slight visually, but the important remarks are that the elementWise method smooths the shock, allowing to attenuate problematic oscillations, but generates structures with less intensity than the pointWise method.

The quantitative study reveals that the application of Newtonian artificial viscosity with an element by element detection leads to an increase in the overall viscosity injected into the simulation. While this approach is effective in mitigating oscillations near shock zones, it results in greater numerical dissipation, thereby reducing kinetic energy and enstrophy. Furthermore, it fails to perfectly balance the kinetic energy budget, leaving a residual error.

The pointWise approach to artificial viscosity tends to create more small structures, indicating improved accuracy. However, this method also causes significant Gibbs oscillations near the shock. Despite these problematic oscillations, the kinetic energy budget appears to gradually adjust, moving closer to a complete closure.

3.3   Comparative Analysis of Sensor Effects

This section aims to study the impact of Newtonian injection combined with different types of sensors, including both regular and physical sensors. For this purpose, the same grid with dimensions (120×60) of order 4, previously used to investigate the influence of point-like artificial viscosity, will be utilized. The analysis will focus on the effects of using these sensors in conjunction with Newtonian injection on the simulation.

The Hennemann sensor does not facilitate a complete simulation of the phenomenon. Therefore, following the analyses, a section will be dedicated to discussing the reasons behind this limitation.

3.3.1   Visual analysis

Density fields

Fig. 4.25 illustrates the density profiles obtained using PerssonPeraire and the physical sensor. With regard to the use of the PerssonPeraire sensor, it is particularly noteworthy that a new advancement in terms of precision has been achieved. This is evidenced by a higher level of precision in Newtonian injection compared to Laplacian injection.

For Newtonian injection, the figures generated by the different sensors are similar.

(a) PerssonPeraire sensor   (b) Physical sensor

Figure 4.25: Density fields at time $t = 0.7$ [s] obtained through an artificial viscosity method with Newtonian injection for the PerssonPeraire sensor and the physical sensor. The simulation is conducted on a mesh 120×60 using a fourth-order scheme.

Artificial bulk viscosity

Although the variation in density profiles among sensors is nuanced, the distinction in detection and artificial viscosity application is considerably more marked. Fig. 4.26 illustrates the disturbed cells and the injected artificial bulk viscosity, which differ according to the sensor used.

It is noteworthy that using Newtonian injection with the PerssonPeraire sensor results in more extensive detection, including the detection of the vortex. This is surprising because the Persson-Peraire sensor is based on the regularity of the solution, and it is intriguing to see that with Newtonian injection, the solution becomes less regular. This can likely be explained by the absence of the Laplacian in this case, which means there is no smoothing, thereby naturally accentuating

irregularities. A significant outcome is that Newtonian injection intensifies the solution's irregularities, enhancing precision, especially in the compression zones generated by vortex splitting. However, this also makes the simulation more complex and less stable by yielding less smooth solutions. It is also important to note that, while this sensor is elementWise in nature, the addition of viscosity is not elementWise due to the use of other quantities in the equations (Eq. 2.48) taken at the point level rather than the element level.

The physical sensor, in contrast, clearly has better precision in terms of shock capture and hence injection. It has not really changed with respect to the injection method used, likely because it is based on a physical criterion, which is the velocity divergence rather than the solution's regularity.

(a) Troubled cells (PerssonPeraire sensor)

(b) Artificial bulk viscosity (PerssonPeraire sensor)

(c) Troubled cells (Physical sensor)

(d) Artificial bulk viscosity (Physical sensor)

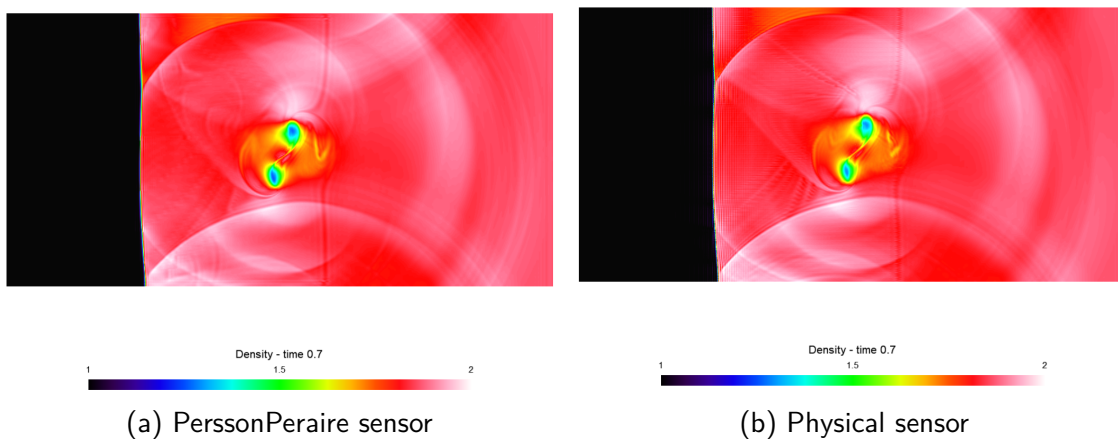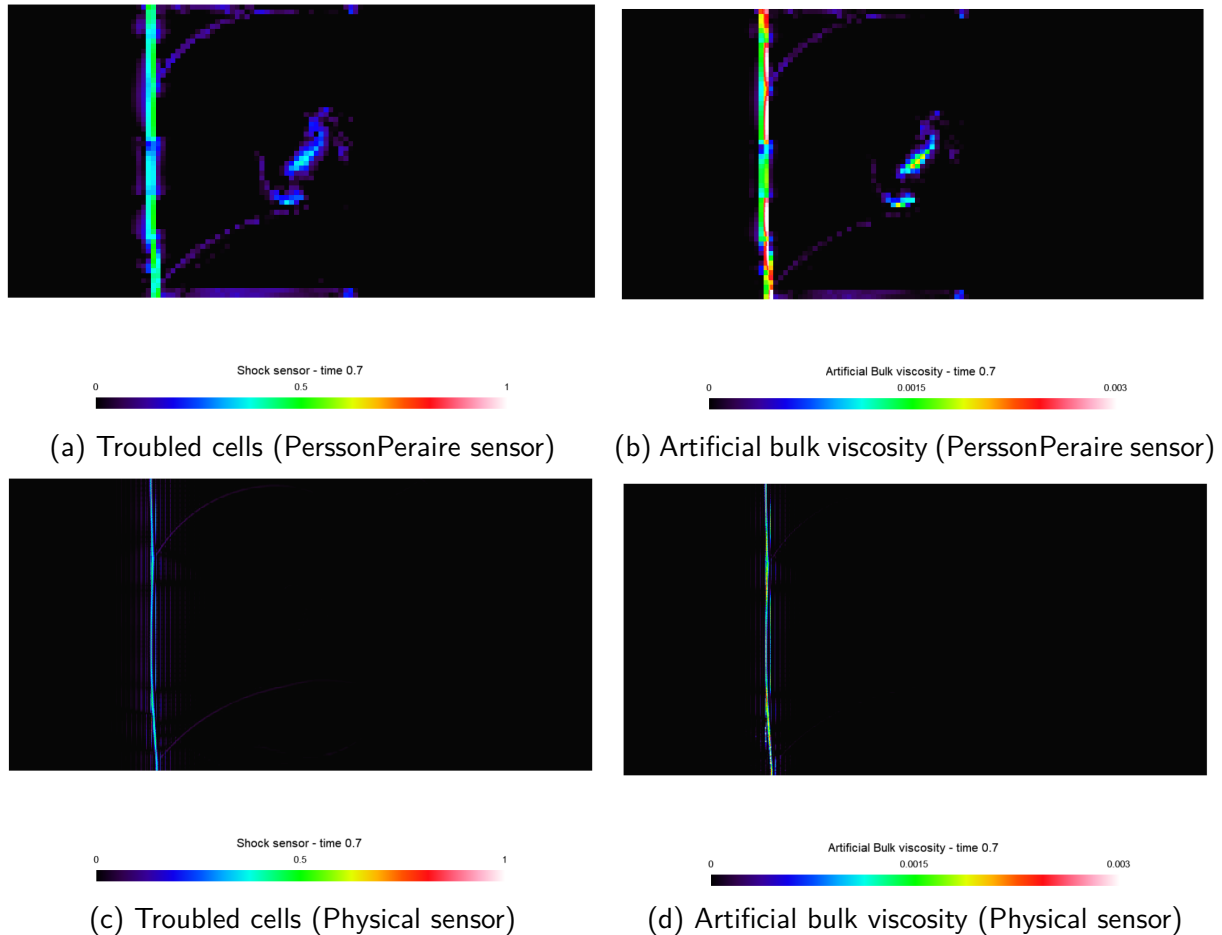Figure 4.26: Regions of troubled cells and artificial viscosity fields at time $t = 0.7$ [s], obtained through an artificial viscosity method with Newtonian injection for the PerssonPeraire sensor and the physical sensor. The simulation is conducted on 120×60 mesh using a fourth-order scheme.

Schlieren

Fig. 4.27 shows that the physical sensor exhibits a higher intensity, more distinctly highlighting areas of density variation, suggesting improved precision. However, it also results in a greater number of oscillations with higher intensities compared to the PerssonPeraire sensor. These outcomes are partly due to the fact that the PerssonPeraire sensor is elementWise and injects a

(a) PerssonPeraire sensor   (b) Physical sensor
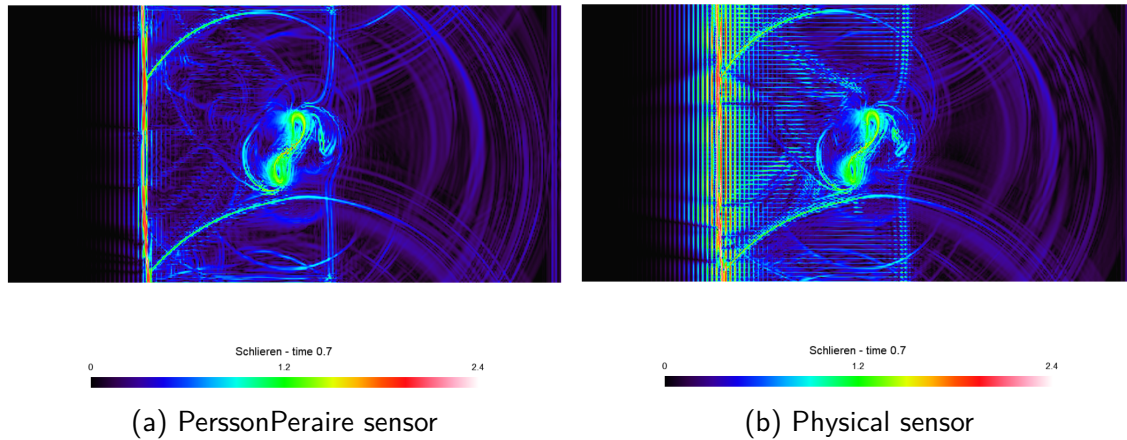
Figure 4.27: Numerical schlieren at time $t = 0.7$ [s], obtained through an artificial viscosity method with Newtonian injection for the PerssonPeraire sensor and the physical sensor. The simulation is conducted on a 120×60 mesh using a fourth-order scheme.

larger amount of viscosity. However, this also reveals that the physical sensor, being pointWise in nature, has a significant drawback in the form of these oscillations.

3.3.2   Quantitative Analysis

Artificial bulk viscosity

Figure 4.28 presents the integrated bulk artificial viscosity β^* over the volume.
The overall behavior of the artificial viscosity is similar to that observed with the Laplacian injection. Regular sensors show an increase in artificial viscosity across the entire volume, and the PerssonPeraire sensor adds even more artificial viscosity.
In the case of the Laplacian injection, this additional increase in artificial viscosity is considered a disadvantage as it reduces precision compared to other sensors. However, with the use of the Newtonian injection, it is noted that this sensor enables the completion of the entire simulation compared to the Hennemann sensor. The latter fails shortly after the vortex crosses the shock, and one possible reason is that it does not inject enough artificial viscosity. In other words, the PerssonPeraire sensor offers better robustness for an artificial viscosity method with the Newtonian Injection than Hennemann sensor.
The other observations regarding the difference between the PerssonPeraire sensor and the physical sensor are the same as with the Laplacian injection. The physical sensor maintains a constant overall artificial viscosity throughout the simulation, as it is based on the divergence of velocity and acts mainly in the shock region. The PerssonPeraire sensor will detect all discontinuities in the simulation, including those not originating from a shock. As a result, the use of this sensor leads to an increase in overall artificial viscosity.

Kinetic energy and enstrophy

Figure 4.29 shows the variations of dimensionless kinetic energy and enstrophy integrated over the entire volume.
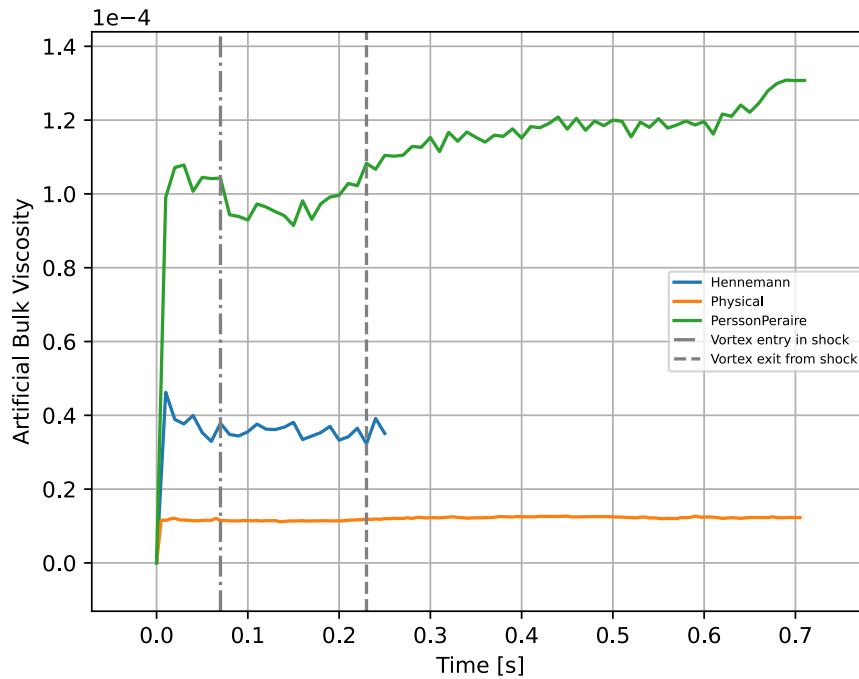
Figure 4.28: Artificial bulk viscosity integrated over the volume obtained through an artificial viscosity method with Newtonian injection for the PerssonPeraire sensor, the Hennemann sensor and the Physical sensor. The simulation is conducted on a 120×60 mesh using a fourth-order scheme.
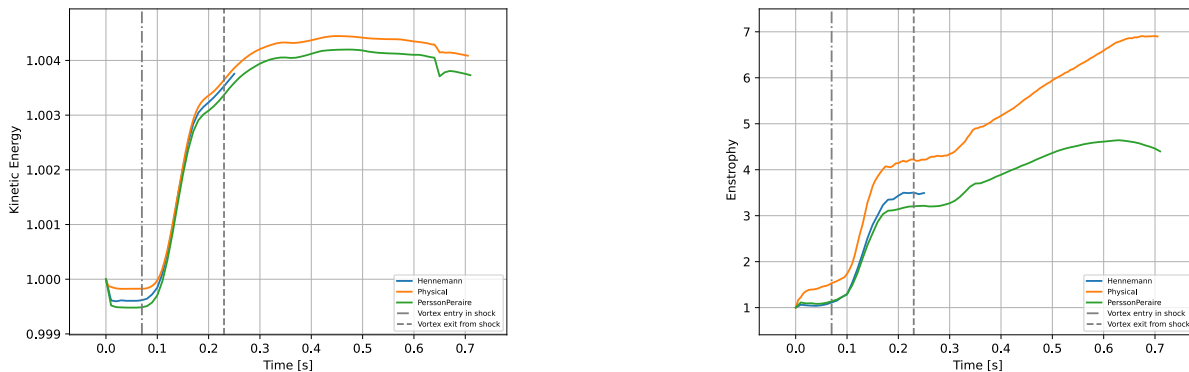
Figure 4.29: Dimensionless kinetic energy and enstrophy integrated over the volume obtained through an artificial viscosity method with Newtonian injection for the PerssonPeraire sensor, the Hennemann sensor and the Physical sensor. The simulation is conducted on a 120×60 mesh using a fourth-order scheme.

Regarding kinetic energy, the overall trends generally follow the same pattern as those observed with Laplacian injection. The initial aberration at the start of the simulation is slightly less influenced by the artificial viscosity generated by the physical sensor. However, greater numerical dissipation is observed with the use of the PerssonPeraire sensor.

In terms of enstrophy, the differences between the sensors are more pronounced, although the behavior is the same (the curve of the physical sensor is higher than that of PerssonPeraire).

Kinetic energy budget

The budget in Figure 4.30 is similar to that encountered for the Laplacian injection in Figure 4.9. This similarity allows us to conclude that the opposing behavior between the physical sensor and the regular sensors is more due to their inherent nature rather than the injection method used. The physical sensor tends to converge towards the null curve, while the regular PerssonPeraire sensor tends to remain constant, maintaining a numerical error due to increasing artificial viscosity.
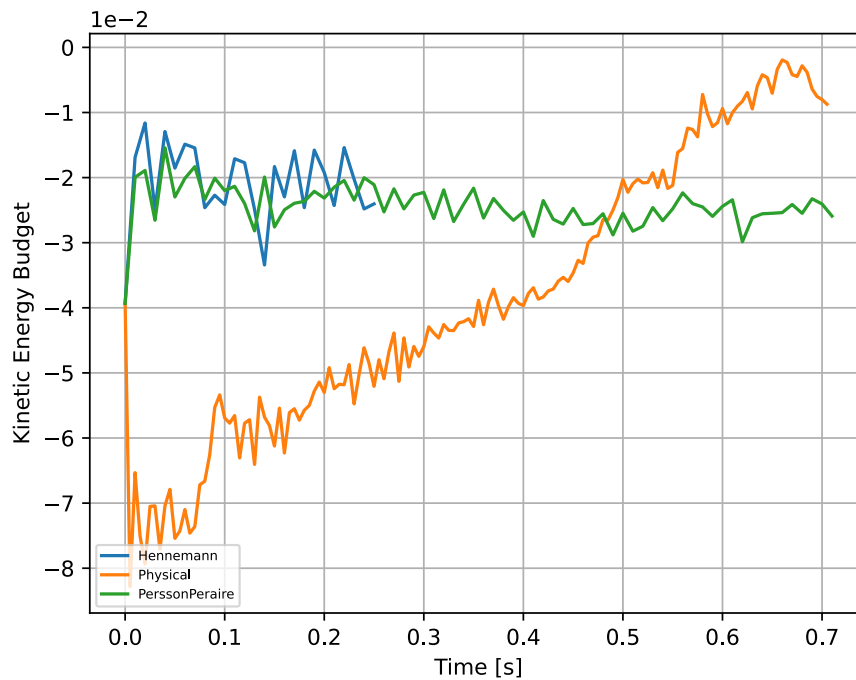
Figure 4.30: Kinetic energy budget obtained through an artificial viscosity method with Newtonian injection for the PerssonPeraire sensor, the Hennemann sensor and the Physical sensor. The simulation is conducted on a 120×60 mesh using a fourth-order scheme.

3.3.3   Simulation failure

It is crucial to recognize that different artificial viscosity sensors can have varying detection thresholds and adjustment methods, making them more or less suitable for different simulation conditions. In this scenario, it appears that the Hennemann sensor may not be capable of generating an adequate amount of artificial viscosity to control instabilities when using the Newtonian injection.

Although temporal graphs do not reveal any obvious problems with this sensor, this section aims to take a closer look at what is happening in terms of artificial viscosities for the different sensors over a relatively short period. The goal is to confirm or refute the hypothesis that the amount of

artificial viscosity generated is insufficient to stabilize this test case in this specific configuration. To achieve this objective, the analysis will focus on a relatively short time interval, specifically at $t = 0.03$ [s].

Figure 4.31 illustrates the values of artificial viscosity injected by the PerssonPeraire and physical sensors. In this representation, the intensity scale for artificial viscosity was specifically chosen to show saturated artificial viscosity at the level of the stationary shock.

(a) PerssonPeraire sensor   (b) Physical sensor

Figure 4.31: Artificial viscosity fields at time t = 0.03 [s], obtained through an artificial viscosity method with Newtonian injection for the PerssonPeraire sensor and the physical sensor. The simulation is conducted on 120×60 mesh using a fourth-order scheme.

Figure 4.32 presents the values of artificial viscosity injected by the Hennemann sensor, using the same saturated scale as mentioned previously. By comparing this figure with that of the PerssonPeraire and physical sensors, it is observable that the artificial viscosity generated by the Hennemann sensor is lower than that of the other two sensors. Indeed, the figure shows no saturation of the scale when examining the artificial viscosity along the shock. This lower value of artificial viscosity allows for the temporary stabilization of the solution. However, when the vortex begins to cross the shock, the discontinuities become so significant that the simulation becomes unstable and can no longer progress.

This observation supports the hypothesis that the Newtonian injection method, as implemented in the configuration defined in this study, fails to produce an adequate amount of artificial viscosity necessary to ensure the stability of the simulation.
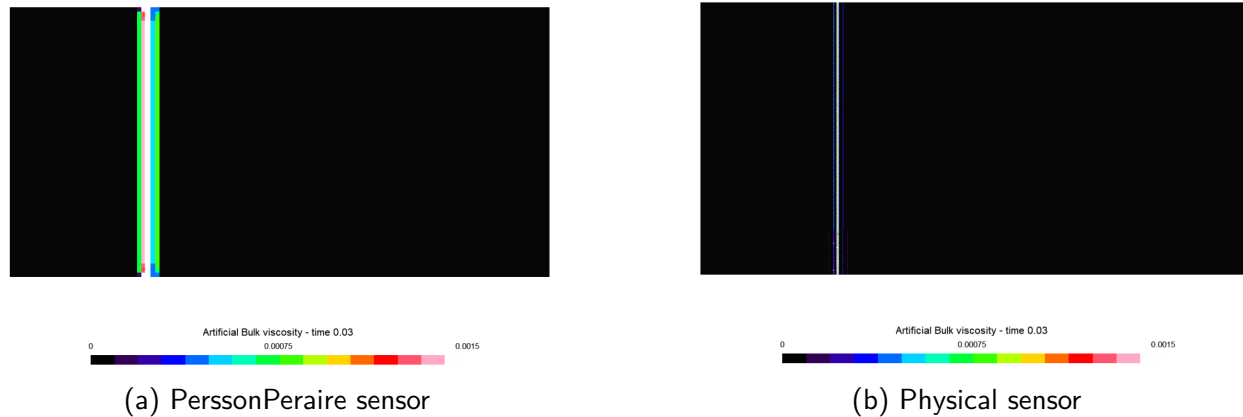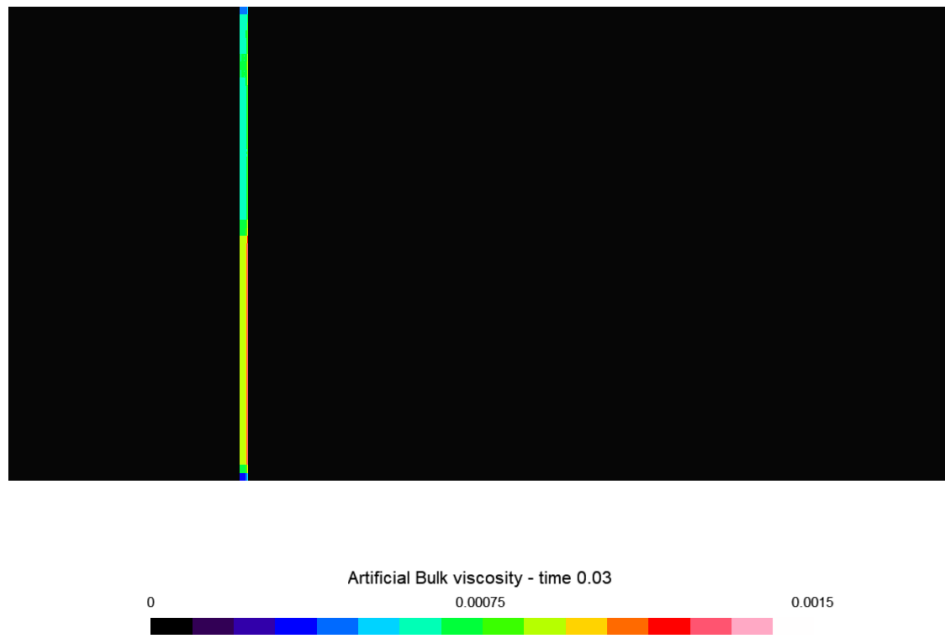
Figure 4.32: Artificial viscosity fields at time t $= 0.03$ [s], obtained through an artificial viscosity method with Newtonian injection for the Hennemann sensor. The simulation is conducted on 120×60 mesh using a fourth-order scheme.

3.3.4　Conclusion

The integration of Newtonian injection into sensors increases accuracy but also leads to more irregularities and oscillations in the solutions, especially pronounced with the physical sensor. Regarding the amount of artificial viscosity, while it cannot be directly compared with the Laplacian method due to the difference in units, the overall trends are similar. The PerssonPeraire sensor proves to be more dissipative than the Hennemann sensor, which in turn is more dissipative than the physical sensor. This hierarchy of dissipation affects quantities such as kinetic energy and enstrophy.

For regular sensors, the budgets do not seem to converge towards closure, likely due to a residual error linked to the incompatibility of these sensors with the Newtonian injection. However, the physical sensor, despite a larger error initially, trends towards the closure of the budget over the course of the simulation.

Regarding the Hennemann sensor, despite its malfunction, the observed overall quantities correspond to the expectations of the Laplacian injection. This malfunction is due to its limited capacity to inject sufficient artificial viscosity at the beginning of the simulation to ensure stability, unlike the PerssonPeraire sensor, which, although more dissipative, is more robust.

4 Evaluating Laplacian and Newtonian Injection Techniques

After separately examining the methods of artificial viscosity through Laplacian and Newtonian injections, it is useful to clearly distinguish their differences. The purpose of this part is to compare them more rigorously, both visually and quantitatively.

The first section is dedicated to a visual analysis of the differences between Laplacian and Newtonian injections based on a physical criterion. This analysis allows for a comparison of the changes that occur simply in relation to the injection method.
The second section will be similar to the first, but this time using a Laplacian injection with a regular criterion, given that this type of criterion is usually associated with Laplacian injection [35, 36].
Finally, the third and last section aims to explore the differences between the types of injections with their respective sensors in a more quantitative way.

This chapter will also include a study on the differences resulting from a change in meshing. The main reason for this analysis in this section is to examine the reaction of the different injection methods to a change in meshing, and to determine whether they exhibit similar behaviors or not. To conduct this analysis, the selected meshes are 96×48 of order 4, 120×60 of order 3, and 120×60 of order 4. The first two were chosen to assess whether, at the same resolution, the simulation is influenced by a higher order of interpolation. The last mesh was selected to examine how the methods vary solely with an increase in order.

4.1 Visual analysis with a physical criteria

Density fields

Fig. 4.33 illustrates the density profiles at time $t = 0.7$ [s] for Laplacian and Newtonian type injections using the same physical sensor, applied to different types of mesh.
The figure illustrates that Newtonian injection, regardless of the type of mesh used, offers better efficiency in highlighting vortex separation and in enhancing contrast. It also demonstrates that to achieve equivalent precision in the density profile with a 96×48 order 4 mesh, representing 115,200 degrees of freedom, using Newtonian injection, it is necessary to employ a finer mesh of 120×60 order 4 for Laplacian injection, which implies the use of 180,000 degrees of freedom.
The figure reveals that with an identical injection method and comparable mesh resolutions (120×60 order 3 versus 96×48 order 4), the density profiles tend to be slightly more precise when a mesh with a larger number of elements is used. This difference, although modest, is particularly noticeable in the visualization of the vortex core.
The figure shows that, although the density profiles are generally similar, the one resulting from the Newtonian injection with a 120×60 order 4 mesh stands out for its greater precision. This increased precision allows for the revelation of structures that are less visible or apparent in the other profiles.

Artificial viscosity

Fig. 4.34 illustrates artificial viscosity at time $t = 0.7$ [s] for each type of injection, using the three previously mentioned meshes. It is important to note that this figure is not intended to

(a) 96×48 - Order 4 (Laplacian injection)

(b) 96×48 - Order 4 (Newtonian injection)

(c) 120×60 - Order 3 (Laplacian injection)

(d) 120×60 - Order 3 (Newtonian injection)

(e) 120×60 - Order 4 (Laplacian injection)

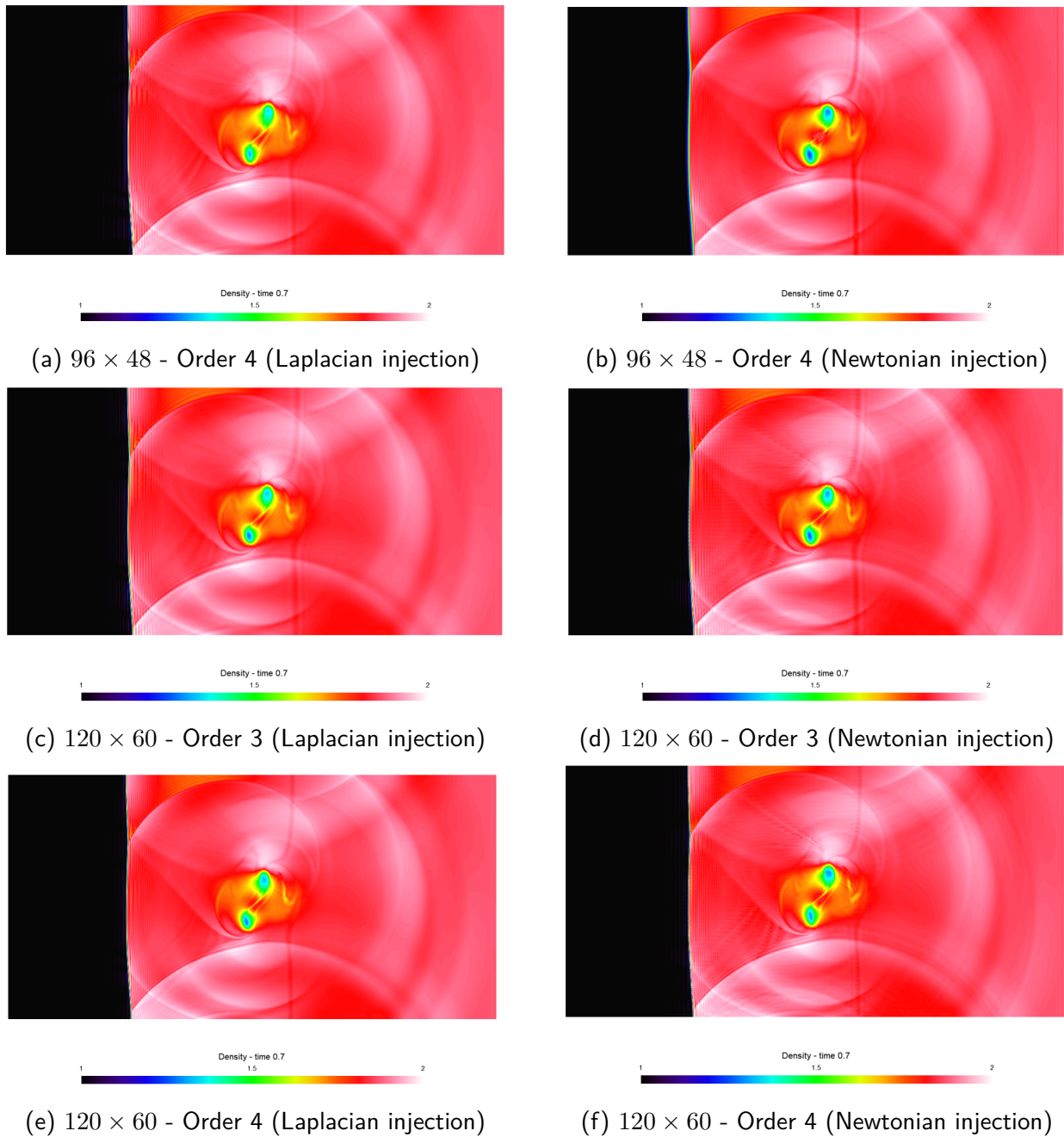(f) 120×60 - Order 4 (Newtonian injection)

Figure 4.33: Density fields at time $t = 0.7$ [s] obtained through an artificial viscosity method with Laplacian and Newtonian injection for the physical sensor. The simulation is conducted on three meshes, 96×48 order 4, 120×60 order 3 and 120×60 order 4.

compare the intensity of artificial viscosity in shock zones. The artificial viscosity differs depending on the injection method used, representing distinct physical quantities. In the case of Newtonian injection, the artificial viscosity corresponds to a form of dynamic viscosity, whereas for Laplacian injection, it equates to a form of kinematic viscosity.

For Laplacian injection, the intensity of artificial viscosity is almost identical, regardless of the mesh used. Although local variations in intensity are present, they do not differ significantly between the different meshes. Despite the fact that the calculation of artificial viscosity incorporates
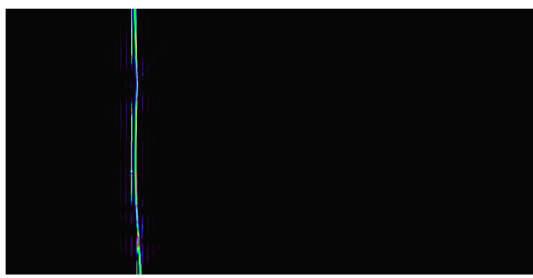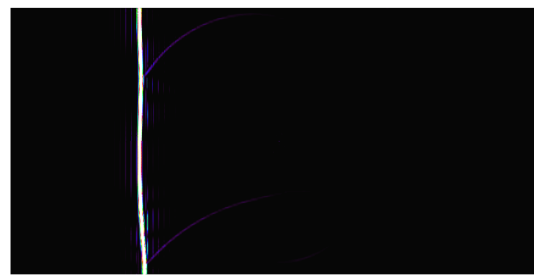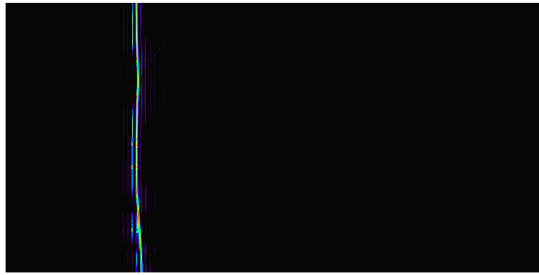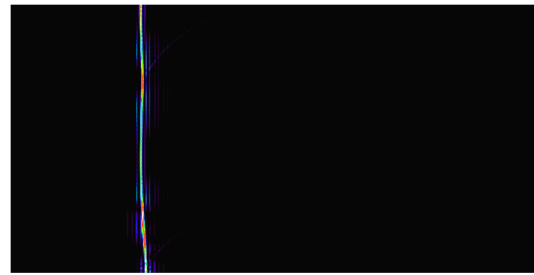
(a) 96×48 - Order 4 (Laplacian injection)

(b) 96×48 - Order 4 (Newtonian injection)

(c) 120×60 - Order 3 (Laplacian injection)

(d) 120×60 - Order 3 (Newtonian injection)
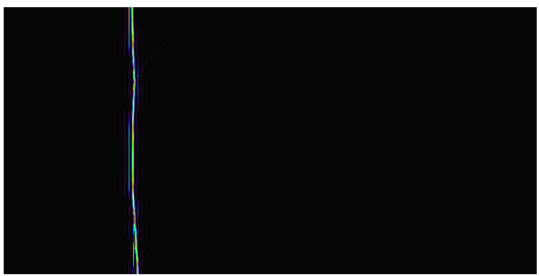
(e) 120×60 - Order 4 (Laplacian injection)
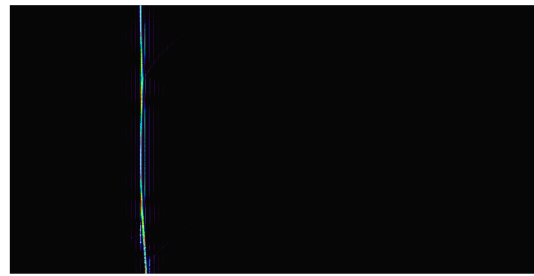
(f) 120×60 - Order 4 (Newtonian injection)

Figure 4.34: Artificial viscosity fields at time $t = 0.7$ [s] obtained through an artificial viscosity method with Laplacian and Newtonian injection for the physical sensor. The simulation is conducted on three meshes, 96×48 order 4, 120×60 order 3 and 120×60 order 4.

a geometric term dependent on the size and order of the mesh, these factors do not seem to have a major impact for the configurations considered in this analysis. It is clear that with a 20×20 order 2 mesh, the difference would be more striking.

Regarding Newtonian injection, it is clearly visible that the finer the mesh, in terms of size or order of precision, the lower the intensity of the artificial viscosity. This is considered beneficial since the thickness of the shock depends on the order of interpolation used; thus, it is logical that the artificial viscosity also depends on it. Nevertheless, this reduction in intensity changes very quickly, and there could be problems using a higher resolution, especially if the order is increased.

Schlieren

Fig. 4.35 displays the numerical schlieren at time $t = 0.7$ [s] for each injection method, using the different meshes. For a comprehensive analysis, the discussion will be conducted in two parts. Firstly, an analysis of the differences observed from the same type of mesh will be performed. Then, a second discussion will focus on the noticeable variations depending on the mesh used for a given injection method, in order to determine if these observations are consistent for both methods of injection.

For the first mesh, 96×48 order 4, a major difference between the two methods of injection is observed in the oscillations near the shock. The Newtonian injection exhibits fewer oscillations than the Laplacian injection, even though the latter uses a direct approach, i.e., without smoothing, unlike the Laplacian.
For the second mesh, the Newtonian injection method generates more structures compared to the 96×48 order 4 mesh. However, for the Laplacian injection method, the difference in terms of structures generated compared to the previous mesh is not particularly pronounced.
Regarding the third mesh, the simulation appears more intense with Newtonian injection, revealing more structures, a better distinction in the separation of vortices accompanied by more contrasted Kelvin-Helmholtz instabilities, and markedly more pronounced oscillations.

Having covered the differences between the meshes, it's now relevant to examine how each method varies with resolution.
For Newtonian injection, it is observed that increasing the order of precision for a reduced number of cells seems to reduce the oscillations near the shock. As the mesh size or its order increases, oscillations become more visible and more intense. This trend could be explained by a gradual reduction in artificial viscosity as shown in the previous figure Fig. 4.34.
It is complex to determine whether, at equal resolution, a coarser mesh with a higher order of precision outperforms its inverse. However, it is noted that with a finer mesh (120×60 order 3), new structures appear, but also the number of oscillations near the shock intensifies.
For the Laplacian injection method, the two meshes of the same resolution seem to show little significant difference between them. A slightly more notable difference is observed in the separation of vortices with a finer mesh, 120×60 order 4. This improvement in precision seems to be mainly attributable to the increase in the mesh's order.

In conclusion, for a given resolution, the Laplacian injection method with artificial viscosity does not seem to bring significant differences compared to its Newtonian equivalent, which generates more microscopic structures with a finer mesh and a lower order of reduction. However, this approach also accentuates the oscillations near the shock. The most precise mesh, 120×60 order 4, reveals more structures in Newtonian injection, demonstrating additional precision, but also leads to an increase in oscillations in terms of intensity, which could pose limits. These oscillations could become problematic with a future increase in order.
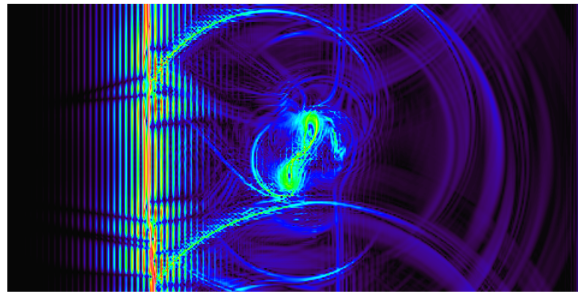
(a) 96×48 - Order 4 (Laplacian injection)
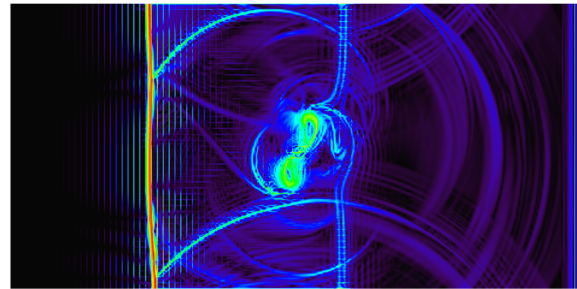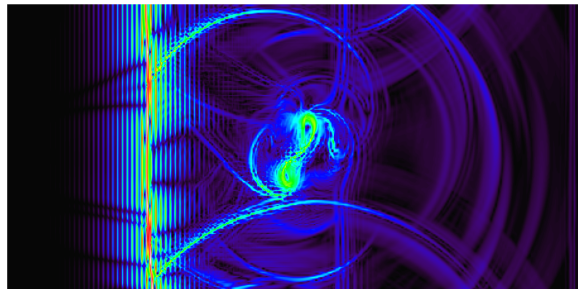
(b) 96×48 - Order 4 (Newtonian injection)

(c) 120×60 - Order 3 (Laplacian injection)
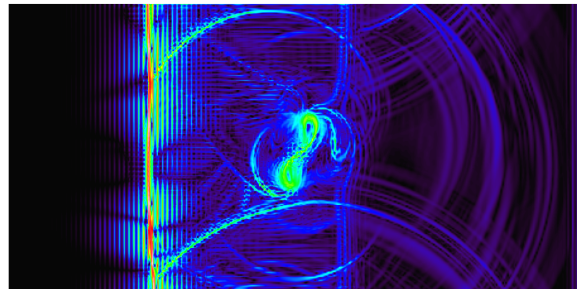
(d) 120×60 - Order 3 (Newtonian injection)
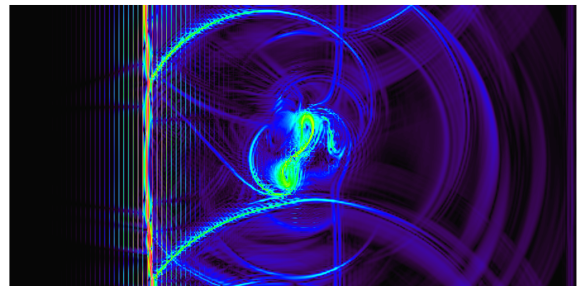
(e) 120×60 - Order 4 (Laplacian injection)
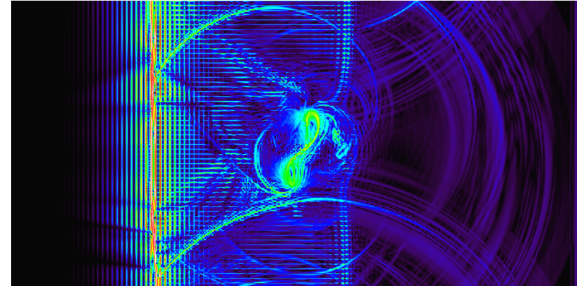
(f) 120×60 - Order 4 (Newtonian injection)

Figure 4.35: Numerical schlieren at time $t = 0.7$ [s] obtained through an artificial viscosity method with Laplacian (left) and Newtonian (right) injection for the physical sensor. The simulation is conducted on three meshes, 96×48 order 4 (top), 120×60 order 3 (middle) and 120×60 order 4 (bottom).

4.2   Visual analysis with a regularity criteria

Density fields

The figure Fig. 4.36 displays density profiles for Laplacian injection with the Hennemann sensor and Newtonian injection with the physical sensor.

Regardless of the mesh resolution, the figure shows that vortex separation is better executed in the context of Newtonian injection, demonstrating increased precision. It also follows that in the figures developed by the Hennemann sensor, arc-shaped structures emanating from the vortex separation are present, caused by the Hennemann sensor. These can appear for other sensors but much less significantly, as can be observed in the figure for a Newtonian injection with a 120×60 order 4 mesh.

For the same resolution (96×48 order 4 and 120×60 order 3), there are no real differences visually for the injection methods, except for a better contrast of the core of the highest vortex in the Newtonian injection method.

Schlieren

The discussion on numerical schlieren will proceed in two parts. First, a comparison of figures for the same mesh to highlight their differences. Then, an examination of the behavior of Laplacian injection with a Hennemann sensor during mesh changes, compared to previous observations. Fig. 4.37 shows the digital schlieren resulting from Laplacian injection with a Hennemann sensor, and a Newtonian injection method using a physical sensor.

When comparing a mesh of dimensions 96×48 order 4, distinct structures appear, particularly arc-like shapes emerging from the vortex separation. The compression regions within the vortex separation are also more distinctly represented with the Newtonian injection method. In both approaches, numerous oscillations are observed near the shock zone. However, these oscillations appear to be relatively similar in terms of intensity and frequency, showing no significant differences between the two methods.

For the second mesh, the same structures appear for both injection methods, with increased visibility in the case of Laplacian injection. The vortex separation is again better defined. Regarding the oscillations near the shock for Laplacian injection, they seem to be finer.

For the last mesh, new structures appear near the shock, absent in the Newtonian injection method. It remains to be determined if these structures are masked by the oscillations induced by the shock or if they are indeed nonexistent in this method. Additionally, a grid-like form between the shock and the vortex separation, while present in all figures, is much more pronounced and striking with the Laplacian injection method. Finally, this injection method reveals other structures, notably Kelvin-Helmholtz instabilities, which are more precisely defined near the vortex separation.

After exploring the differences between the two injection methods, the analysis now focuses on the behavior of the Laplacian injection method with a Hennemann sensor, depending on the mesh used.

The observed results are consistent with expectations: a finer mesh improves the precision of the figure, and this precision also increases with the mesh order. However, an important nuance to note is that a mesh with a lower number of cells but a higher order (e.g., 96×48) allows for

(a) 96×48 - Order 4 (Laplacian injection)

(b) 96×48 - Order 4 (Newtonian injection)

(c) 120×60 - Order 3 (Laplacian injection)

(d) 120×60 - Order 3 (Newtonian injection)

(e) 120×60 - Order 4 (Laplacian injection)

(f) 120×60 - Order 4 (Newtonian injection)
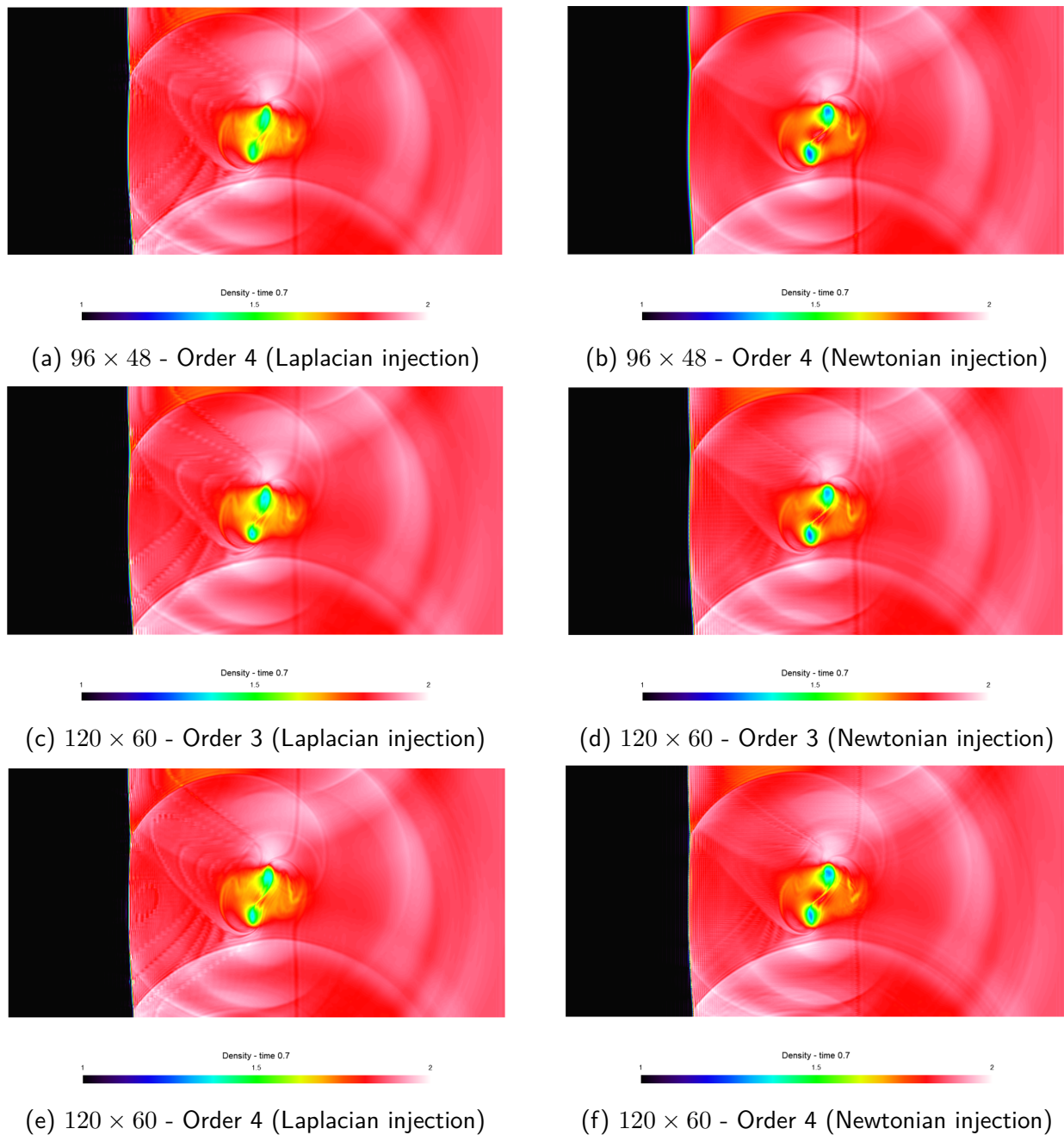
Figure 4.36: Density fields at time $t = 0.7$ [s] obtained through an artificial viscosity method with Laplacian injection for the Hennemann sensor and Newtonian injection for the physical sensor. The simulation is conducted on three meshes, 96×48 order 4, 120×60 order 3 and 120×60 order 4.

better precision in vortex separation. Although this difference is minor, it is worth mentioning.

In conclusion, the schlieren analysis reveals that, regardless of the mesh used, the Newtonian injection method allows for more precise vortex separation compared to the Laplacian injection with a Hennemann sensor. However, additional structures are generated by the Laplacian injection method using this sensor. As observed previously, whether in the comparison of injection methods

with the physical sensor or in the analysis of sensors within the Laplacian injection method, these emerging structures are primarily due to the use of the Hennemann sensor and not the injection method itself.
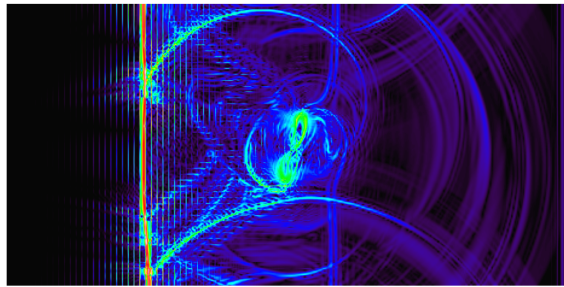
(a) 96×48 - Order 4 (Laplacian injection)
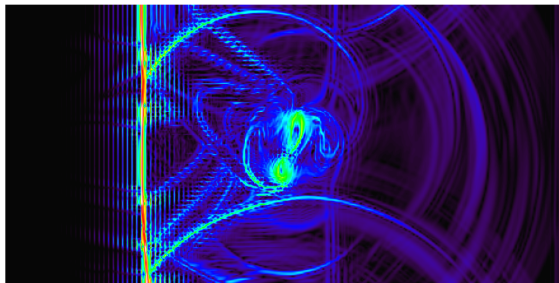
(b) 96×48 - Order 4 (Newtonian injection)

(c) 120×60 - Order 3 (Laplacian injection)
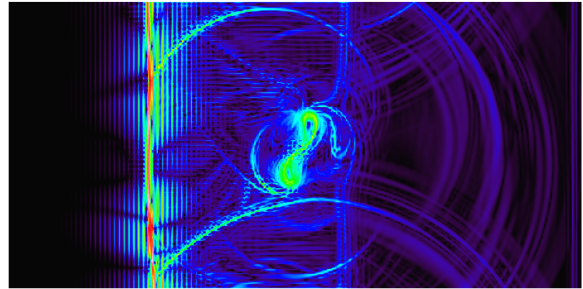
(d) 120×60 - Order 3 (Newtonian injection)
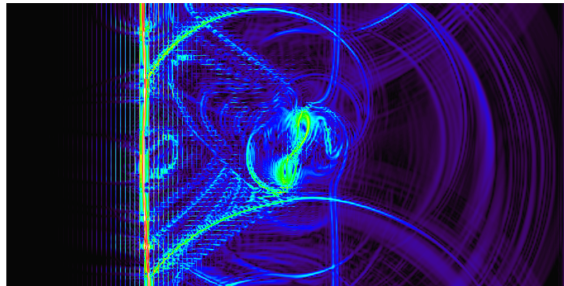
(e) 120×60 - Order 4 (Laplacian injection)
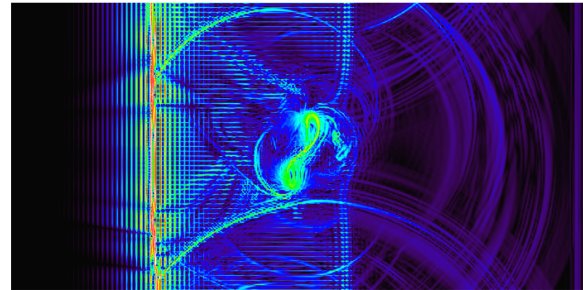
(f) 120×60 - Order 4 (Newtonian injection)

Figure 4.37: Numerical schlieren at time $t = 0.7$ [s] obtained through an artificial viscosity method with Laplacian and Newtonian injection for the physical sensor. The simulation is conducted on three meshes, 96×48 order 4, 120×60 order 3 and 120×60 order 4.
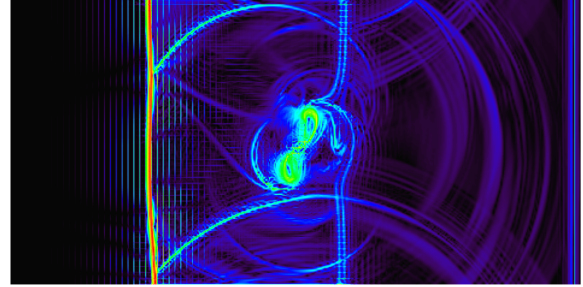
4.3   Quantitative analysis

Artificial viscosity

Fig. 4.38 displays two graphs of artificial viscosity based on the injection method used. The left graph represents the integral of artificial viscosity over the volume for a Laplacian injection, and the right graph is for a Newtonian injection.

(a) Laplacian   (b) Newtonian
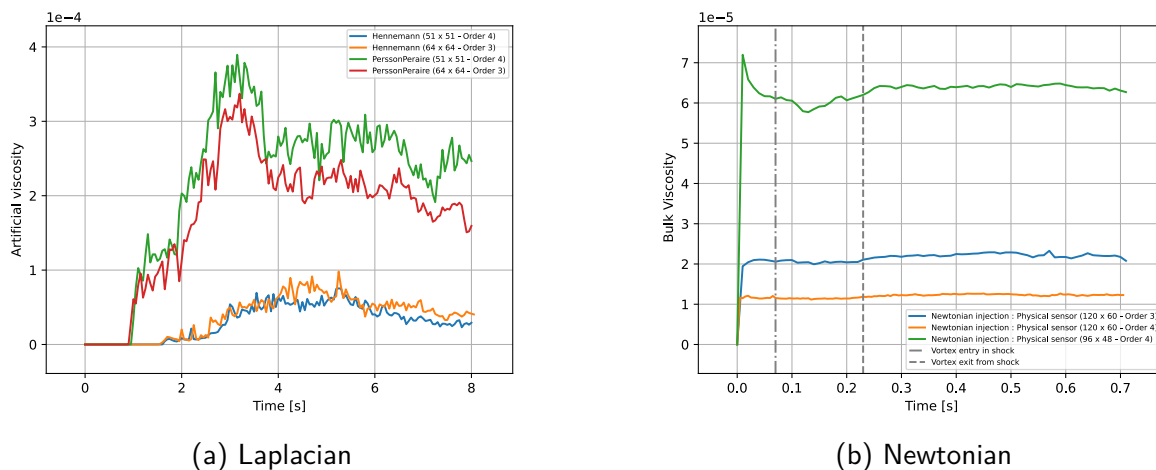
Figure 4.38: Artificial viscosity integrated over the volume obtained through an artificial viscosity method with Laplacian and Newtonian injection. The simulation is conducted on three meshes, 96×48 order 4, 120×60 order 3 and 120×60 order 4.

In each injection method associated with a sensor, the behavior varies depending on the mesh used.

For the Hennemann sensor with Laplacian injection, the primary influence is the mesh resolution (degrees of freedom). The performance curves overlap for a given resolution and improve with a finer resolution.

For the physical sensor with Laplacian injection, the situation is similar. The artificial viscosity is also influenced by a geometric factor, which this sensor integrates into its detection process, thus optimizing the amount of artificial viscosity.

Finally, for the physical sensor with Newtonian injection, different levels of artificial viscosity are observed depending on the refinement of the mesh. The artificial viscosity decreases with an increase in the number of cells or the order of the mesh.

Density profile

The density profile is presented right after the Schlieren analysis to directly examine the impact of oscillations. In this case, as we analyze two Schlierens with two different Laplacian methods, it was preferable to place it in this section.

Fig. 4.39 show the density profiles at $y = 0.4$ for the different meshes and methods discussed.

It is observed that the oscillations vary according to the meshes. Oscillations are more pronounced with a fourth-order mesh than with a third-order mesh.

For the 96×48 mesh of order 4, it is demonstrated that the oscillations are more intense with a physical sensor using a Laplacian method, consistent with the Schlieren observations for this mesh. Moreover, it is noted that the Laplacian methods, and even more so with the Hennemann sensor, show a divergence from the reference, indicating that the Laplacian methods still require adjustments, while the Newtonian injection corresponds better to the expected curve.

For the 120×60 mesh of order 3, the results reveal more pronounced oscillations for the Laplacian method with a physical sensor, and again, a deviation from the reference is observed. However, this gap is now only present with the Hennemann sensor, and no longer with the physical sensor using Laplacian injection. The Newtonian injection also shows oscillations, but they are less significant than those obtained with the Laplacian injection in this case.

For the last mesh 120×60 of order 4, which has already been extensively discussed, it is observed that the Hennemann sensor exhibits significantly more intense oscillations than the others, and despite the increase in order, it still shows a deviation from the reference.

These findings highlight the increased precision achieved with the Newtonian injection method, and to a lesser extent, the use of a pointwise physical sensor.

The oscillations appear more intense when using a fourth-order mesh compared to a third-order one. It is also noted that the most intense oscillations are not those obtained with the Newtonian injection. However, these results should be interpreted cautiously, as they represent a cross-section at a specific moment, and the conclusions drawn could vary significantly with a different section.

Kinetic energy and Enstrophy

The figure Fig. 4.40 illustrates the kinetic energies (on the left) and enstrophy (on the right) for different meshes.

Regarding the kinetic energy for the two meshes of the same resolution, the curve corresponding to the Hennemann sensor appears almost identical, likely due to the same amount of artificial viscosity injection applied. However, for the curves of the physical sensor, regardless of the injection method used, a mesh of 120×60 order 3 seems to generate a slightly higher kinetic energy, while for the mesh of 120×60 order 4, this energy appears to stagnate.

For enstrophy, meshes of the same resolution seem to produce graphs with similar behaviors and intensities. However, with the 120×60 order 4 mesh, an increase in enstrophy is observed. Particularly, in the case of Newtonian injection, this curve shows a significantly more pronounced growth.

Kinetic energy budget

Figure 4.41 displays the kinetic energy budget for the various meshes and methods employed.

The mesh that appears closest to achieving budget closure is the 120x60 at order 4. Initially, this mesh exhibits a more pronounced decrease for Newtonian injection compared to Laplacian, but as the simulation progresses, especially towards the end, it tends to close the budget more effectively than the Laplacian approaches. Despite the oscillations, this mesh seems to be the only one converging.

The other two meshes demonstrate that the simulation is unable to close the budget. The 96×48 mesh with Newtonian injection, however, attempts to converge towards closure, reaching the values of the curves from the second injection method, but it still falls short.
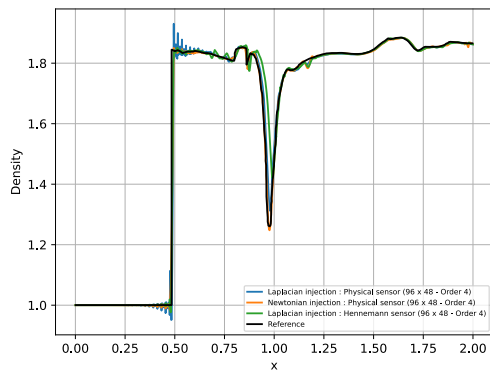
(a) 96×48 - Order 4
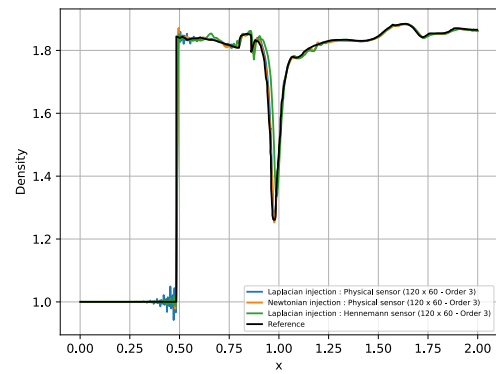
(b) 120×60 - Order 3
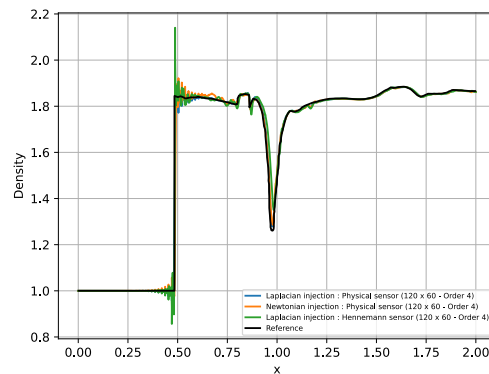
(c) 120×60 - Order 4

Figure 4.39: Density profile along $y = 0.4$ at time $t = 0.7$ [s] obtained through an artificial viscosity method with Laplacian and Newtonian injection. The simulation is conducted on three meshes, 96×48 order 4, 120×60 order 3 and 120×60 order 4.
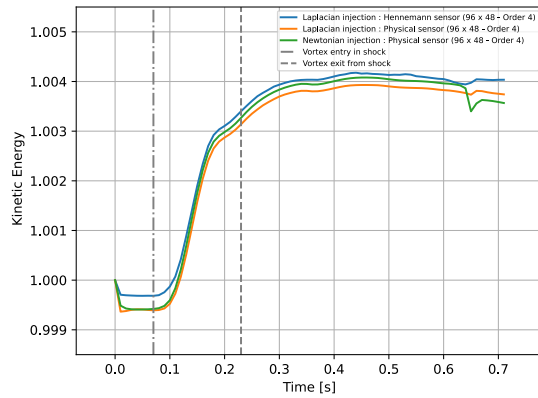
(a) 96×48 - Order 4 (Kinetic energy)
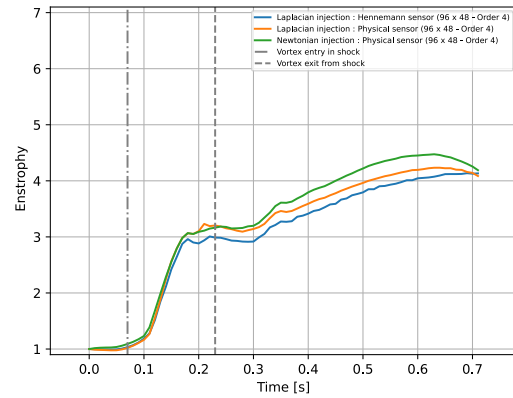
(b) 96×48 - Order 4 (Enstrophy)
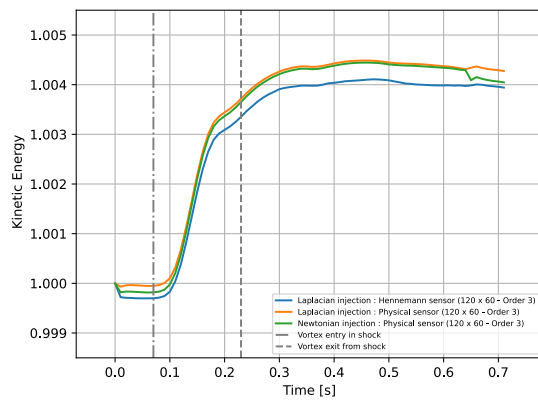
(c) 120×60 - Order 3 (Kinetic energy)
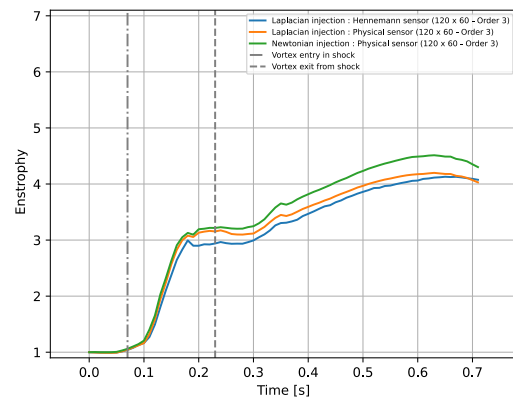
(d) 120×60 - Order 3 (Enstrophy)
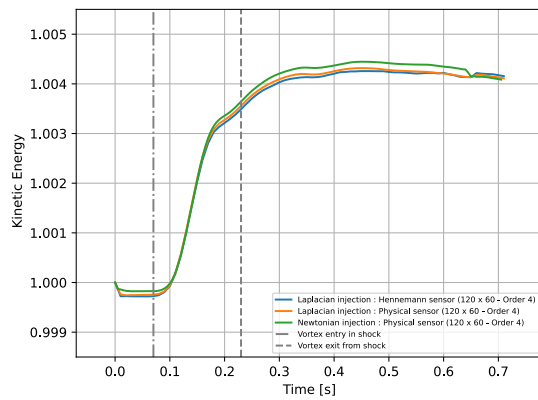
(e) 120×60 - Order 4 (Kinetic energy)
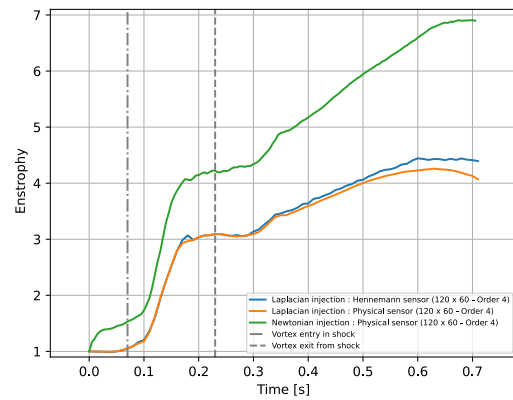
(f) 120×60 - Order 4 (Enstrophy)

Figure 4.40: Dimensionless kinetic energy and enstrophy integrated over the volume obtained through an artificial viscosity method with Laplacian and Newtonian injection. The simulation is conducted on three meshes, 96×48 order 4, 120×60 order 3 and 120×60 order 4.
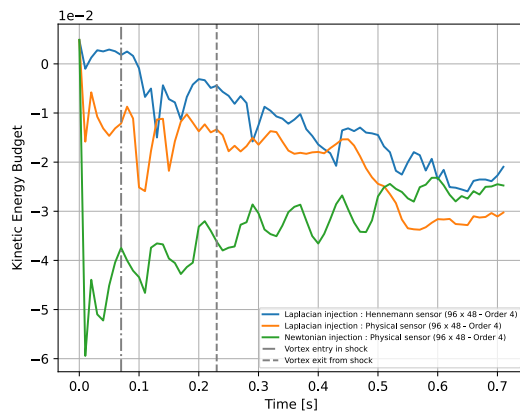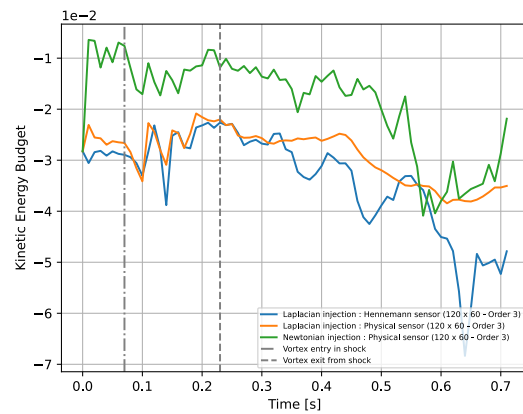
(a) 96×48 - Order 4

(b) 120×60 - Order 3
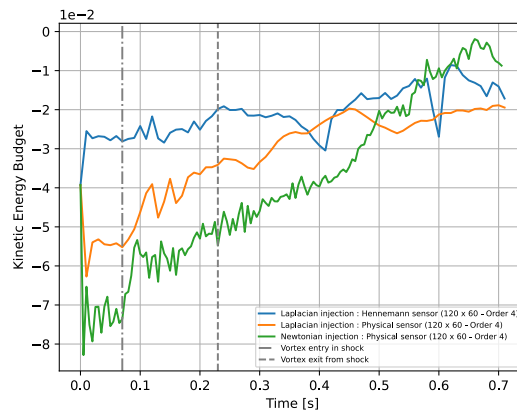
(c) 120×60 - Order 4

Figure 4.41: Kinetic energy budget obtained through an artificial viscosity method with Laplacian and Newtonian injection. The simulation is conducted on three meshes, 96×48 order 4, 120×60 order 3 and 120×60 order 4.

4.4   Conclusion

The Newtonian approach, regardless of the mesh used, shows more accurate results compared to the Laplacian counterpart. This is partly due to the use of the physical shock sensor, which yields better outcomes than the Hennemann sensor, as well as the more realistically conforming injection method employed. The use of a mesh with 115,200 degrees of freedom for a Newtonian injection allows achieving the same (or even better) results as a mesh with 180,000 degrees of freedom for a Laplacian injection.

Both Newtonian and Laplacian injections exhibit increasing oscillations as the mesh becomes finer. However, despite these spurious oscillations, the kinetic energy budget of the methods tends to converge towards its closure.

There is a notable issue with the injection method that could become problematic: the artificial viscosity varies significantly depending on the mesh used. This is not the case with the same sensor but with a Laplacian injection, where the artificial viscosity value remains constant regardless of the mesh used. This mesh-related problem is extremely concerning and could lead to simulation failures with more or less refined meshes.

Conclusion

This master's thesis represents an initial contribution in the field of applying a Newtonian artificial viscosity method focused on physical phenomena. The primary goal of this research was to explore this method and compare it to an approach of artificial viscosity based on the Laplacian.

The study was structured around two major analyses, each focusing on a specific test. The first part, dedicated to Kelvin-Helmholtz instabilities, revealed the significant effect of artificial viscosity in a shock-free simulation, but disrupted by instabilities occurring in the shear layers. The second part, focused on the inviscid strong vortex-shock wave interaction, shed light on the impact of artificial viscosity in a stationary shock context, allowing observation of its reaction and role in system stabilization, particularly in examining how it helps to mitigate or regulate disturbances induced by the shock.

The study demonstrated that, with the user-defined parameters for each test, the application of a Newtonian injection method has the potential to increase the accuracy and realism of simulations. However, this method has not yet reached its full maturity. Additional work is needed to refine this technique and make it completely reliable for regular use.

The Laplacian injection also yields very good outcomes, especially when paired with the Hennemann sensor that requires fewer user-defined parameters. This method is more robust and much more reliable for practical use, yet it nonetheless exhibits a smoothing effect on the solution that could impact the resolution of fine details and the capture of subtle physical phenomena in the simulation.

It is important to approach the results obtained in this study with caution. The application of artificial viscosity is heavily dependent on various user-defined parameters, which can significantly influence the behavior and effectiveness of artificial viscosity methods.

Despite these limitations, this thesis provides an analysis of the initial problem and opens up new avenues for future research. The findings here lay the groundwork for further exploration and improvement, potentially contributing to the advancement of the understanding and application of artificial viscosity methods in fluid dynamics simulations.

Perspectives and futur works

This thesis conducted an in-depth study of two test cases: the Kelvin-Helmholtz instabilities and the inviscid strong vortex-shock interaction. This study highlighted various aspects, such as the effects of regular or physical sensors, the impact of the shear sensor detection threshold, different methods of interpolating artificial viscosity, and the injection techniques used. However, some procedures and choices, like the constants k in the calculation of Newtonian artificial viscosity (Eq. 2.48) or the detection threshold $s_{\mu,0}$ for the shear sensor in the Kelvin-Helmholtz case, were implemented without detailed explanations.

Newtonian artificial viscosity presented stability issues, requiring empirically chosen values to ensure functionality. These parameters, while effective, lack solid theoretical foundations and need further analysis to evaluate their relevance and optimization potential. It is crucial to develop these parameters to make them more universal, like the detection threshold of the PerssonPeraire sensor that varies with the interpolation order. However, even this threshold seems quite low considering the results obtained.

The use of Bernstein polynomials, although promising, did not consistently work in the case of the Inviscid Strong vortex-shock interaction. These polynomials lead to a spread of artificial viscosity, increasing the overall amount of viscosity in the cell. A more refined approach in interpolating artificial viscosity with these polynomials could be explored, aiming to maintain the same amount of viscosity within the cell. Additionally, introducing continuity greater than simple C_0 at interfaces could be beneficial.

Directly applying artificial viscosity at quadrature points showed good results but also revealed a major drawback: Gibbs oscillations near shock areas. Adopting a smoothing method could be an effective solution to this issue. For instance, using a Gaussian filter [44] could significantly improve the management of these difficulties.

For future work not addressed in this thesis, exploring a test case requiring a thermal sensor would be particularly relevant. It would also be interesting to examine the simultaneous use of all physical sensors (shock, shear, and thermal). This would allow for a detailed study of how different artificial viscosities behave and interact with each other, offering deep insights into the complex fluid dynamics and the combined efficiency of these sensors under various simulation conditions.

Continuing with future work, exploring specific test cases to study hypersonic phenomena [2] would be beneficial. These cases could provide crucial insights into fluid behavior at extremely high velocities, a domain presenting unique challenges in terms of fluid dynamics and stability control.

Moreover, incorporating test cases that highlight the importance of using Bernstein polynomials could be very instructive. These cases would allow for specific evaluation of how Bernstein polynomials contribute to the robustness and stabilization in artificial viscosity simulations. By focusing on these areas, it is possible to not only validate and improve existing methods but also to discover new approaches for tackling the complex challenges associated with fluid dynamics in hypersonic and turbulent regimes.

Bibliography

[1]   F. H. Harlow and J. E. Welch. "Numerical Calculation of Time-Dependent Viscous Incompressible Flow of Fluid with Free Surface". In: *Physics of Fluids (1958-1988)* 8 (1965), pp. 2182–2189. DOI: 10.1063/1.1761178.

[2]   Pablo Fernandez, Ngoc Cuong Nguyen, and J. Peraire. "A physics-based shock capturing method for large-eddy simulation". In: *AIAA Aerospace Sciences Meeting* (Jan. 2018), pp. 10.2514/6.2018–0062. DOI: 10.2514/6.2018-0062.

[3]   Scott M. Murman et al. "A Space-Time Discontinuous-Galerkin Approach for Separated Flows". In: *54th AIAA Aerospace Sciences Meeting*. 2016. DOI: 10.2514/6.2016-1059.

[4]   A. Uranga et al. "Implicit Large Eddy Simulation of transition to turbulence at low Reynolds numbers using a Discontinuous Galerkin method". In: *International Journal for Numerical Methods in Engineering* 87.1-5 (2010), pp. 232–261. DOI: 10.1002/nme.3036.

[5]   C.C. de Wiart and K. Hillewaert. "Development and Validation of a Massively Parallel High-Order Solver for DNS and LES of Industrial Flows". In: *IDIHOM: Industrialization of High-Order Methods - A Top-Down Approach*. Vol. 128. Notes on Numerical Fluid Mechanics and Multidisciplinary Design. 2015, pp. 251–292.

[6]   Z. J. Wang et al. "High-order CFD methods: current status and perspective". In: *International Journal for Numerical Methods in Fluids* 73.4 (2013), pp. 269–289. DOI: 10.1002/fld.3767.

[7]   K. Hillewaert et al. "Assessment of high-order DG methods for LES of compressible flows". In: *Center for Turbulence Research Proceedings of the Summer Program*. Vol. 363. 2016.

[8]   Z. J. Wang. "High-order methods for the Euler and Navier–Stokes equations on unstructured grids". In: *Progress in Aerospace Sciences* 43 (2007), pp. 1–41. DOI: 10.1016/j.paerosci.2007.05.001.

[9]   Randall J. LeVeque. *Numerical Methods for Conservation Laws*. Second. Lectures in mathematics: ETH Zürich. Basel, Boston, Berlin: Birkhäuser, 1992. ISBN: 3-7643-2723-5.

[10]   Andrea D. Beck et al. "High-order discontinuous Galerkin spectral element methods for transitional and turbulent flow simulations". In: *International Journal for Numerical Methods in Fluids* 76.8 (2014), pp. 522–548. DOI: 10.1002/fld.3943.

[11]   P. Fernandez et al. "Implicit Large-Eddy Simulation of Compressible Flows Using the Interior Embedded Discontinuous Galerkin Method". In: *54th AIAA Aerospace Sciences Meeting*. American Institute of Aeronautics and Astronautics (AIAA), Jan. 2016. DOI: 10.2514/6.2016-1332.

[12]   P. Fernandez, N. C. Nguyen, and J. Peraire. "The hybridized Discontinuous Galerkin method for Implicit Large-Eddy Simulation of transitional turbulent flows". In: *Journal of Computational Physics* 336 (2017), pp. 308–329. DOI: `10.1016/j.jcp.2017.02.015`.

[13]   A. Frere et al. "Cross-Validation of Numerical and Experimental Studies of Transitional Airfoil Performance". In: *33rd Wind Energy Symposium*. 2015. DOI: `10.2514/6.2015-0499`.

[14]   Gregor J. Gassner and Andrea D. Beck. "On the accuracy of high-order discretizations for underresolved turbulence simulations". In: *Theoretical and Computational Fluid Dynamics* 27.3-4 (2012), pp. 221–237. DOI: `10.1007/s00162-011-0253-7`.

[15]   T. Dzanica, W. Trojak, and F. D. Witherden. "On the anti-aliasing properties of entropy filtering for discontinuous spectral element approximations of under-resolved turbulent flows". In: (2023). DOI: `10.48550/arXiv.2302.13359`.

[16]   Virginie Daru and Christian Tenaud. "Approximations d'ordre élevé pour les écoulements compressibles avec discontinuités". In: (2005).

[17]   Chi-Wang Shu. *Essentially Non-Oscillatory and Weighted Essentially Non-Oscillatory Schemes for Hyperbolic Conservation Laws*. Tech. rep. NASA/CR-97-206253 and ICASE Report 97-65, 1997.

[18]   P. L. Roe. "Approximate Riemann solvers, parameter vectors, and difference schemes". In: *Journal of Computational Physics* 43.2 (Oct. 1981), pp. 357–372. DOI: `10.1016/0021-9991(81)90128-5`.

[19]   Xu-Dong Liu, Stanley Osher, and Tony Chan. "Weighted Essentially Non-oscillatory Schemes". In: *Journal of Computational Physics* 115.1 (1994), pp. 200–212. DOI: `10.1006/jcph.1994.1187`.

[20]   Guang-Shan Jiang and Chi-Wang Shu. "Efficient Implementation of Weighted ENO Schemes". In: *Journal of Computational Physics* 126.1 (June 1996), pp. 202–228. DOI: `10.1006/jcph.1996.0130`.

[21]   D. Kuzmin. "On the design of general-purpose flux limiters for finite element schemes. I. Scalar convection". In: (2006). DOI: `https://doi.org/10.1016/j.jcp.2006.03.034`.

[22]   Hong Luo, Joseph D. Baum, and Rainald Löhner. "A Hermite WENO-based limiter for discontinuous Galerkin method on unstructured grids". In: *Journal of Computational Physics* 225.1 (2007), pp. 686–713. DOI: `10.1016/j.jcp.2006.12.017`.

[23]   F. Bassi and S. Rebay. "A High-Order Accurate Discontinuous Finite Element Method for the Numerical Solution of the Compressible Navier–Stokes Equations". In: *Journal of Computational Physics* 131.2 (1997), pp. 267–279. DOI: `10.1006/jcph.1996.5572`.

[24]   A. Baggag, H. Atkins, and D. Keyes. "Parallel Implementation of the Discontinuous Galerkin Method". In: *Parallel Computational Fluid Dynamics 1999*. 2000, pp. 115–122. DOI: `10.1016/B978-044482851-4/50015-3`.

[25]   Charles Hirsch. *Numerical Computation of Internal and External Flows: The Fundamentals of Computational Fluid Dynamics*. Butterworth-Heinemann, 2007.

[26]   J. Nitsche. "Über ein Variationsprinzip zur Lösung von Dirichlet-Problemen bei Verwendung von Teilräumen, die keinen Randbedingungen unterworfen sind". In: *Abhandlungen Aus Dem Mathematischen Seminar Der Universität Hamburg* 36.1 (1971), pp. 9–15. DOI: 10.1007/bf02995904.

[27]   Koen Hillewaert. "Development of the Discontinuous Galerkin Method for High-Resolution, Large Scale CFD and Acoustics in Industrial Geometries". PhD thesis. Université Catholique de Louvain, 2013.

[28]   Bernardo Cockburn and Chi-Wang Shu. "TVB Runge-Kutta Local Projection Discontinuous Galerkin Finite Element Method for Conservation Laws II: General Framework". In: *Mathematics of Computation* 52.186 (1989), pp. 411–435. DOI: 10.2307/2008474.

[29]   Bernardo Cockburn, San-Yih Lin, and Chi-Wang Shu. "TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: One-dimensional systems". In: *Journal of Computational Physics* 84.1 (1989), pp. 90–113. DOI: 10.1016/0021-9991(89)90183-6.

[30]   Bernardo Cockburn and Chi-Wang Shu. "The Runge–Kutta Discontinuous Galerkin Method for Conservation Laws V: Multidimensional Systems". In: *Journal of Computational Physics* 141.2 (1998), pp. 199–224. DOI: 10.1006/jcph.1998.5892.

[31]   Bernardo Cockburn, Suchung Hou, and Chi-Wang Shu. "The Runge-Kutta Local Projection Discontinuous Galerkin Finite Element Method for Conservation Laws. IV: The Multidimensional Case". In: *Mathematics of Computation* 54.190 (1990), pp. 545–581. DOI: 10.2307/2008501.

[32]   Bernardo Cockburn and Chi-Wang Shu. "The Runge-Kutta local projection P^1-discontinuous-Galerkin finite element method for scalar conservation laws". In: *ESAIM: Mathematical Modelling and Numerical Analysis* 25.3 (1991), pp. 337–361. DOI: 10.1051/m2an/1991250303371.

[33]   Bernardo Cockburn and Chi-Wang Shu. "Runge-Kutta Discontinuous Galerkin Methods for Convection-Dominated Problems". In: *Journal of Scientific Computing* 16.3 (2001), pp. 173–261. DOI: 10.1023/a:1012873910884.

[34]   David Moro, Ngoc Cuong Nguyen, and Jaime Peraire. "Dilation-based shock capturing for high-order methods". In: *International Journal for Numerical Methods in Fluids* (2016). DOI: 10.1002/fld.4223.

[35]   Per-Olof Persson and Jaime Peraire. "Sub-Cell Shock Capturing for Discontinuous Galerkin Methods". In: *44th AIAA Aerospace Sciences Meeting and Exhibit*. 2006. DOI: 10.2514/6.2006-112.

[36]   Sebastian Hennemann et al. "A provably entropy stable subcell shock capturing approach for high order split form DG for the compressible Euler equations". In: *Journal of Computational Physics* 426 (), p. 109935. DOI: 10.1016/j.jcp.2020.109935.

[37]   F. Ducros et al. "Large-Eddy Simulation of the Shock/Turbulence Interaction". In: *Journal of Computational Physics* 152 (1999), pp. 517–549. DOI: 10.1006/jcph.1999.6238.

[38]   Ray Vandenhoeck and Andrea Lani. "Implicit high-order flux reconstruction solver for high-speed compressible flows". In: *Computer Physics Communications* 242 (Sept. 2019), pp. 1–24. DOI: 10.1016/j.cpc.2019.04.015.

[39]   Linbo Zhang. "A second-order upwinding finite difference scheme for the steady Navier-Stokes equations in primitive variables in a driven cavity with a multigrid solver". In: *M2AN. Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique* 24.1 (1990), pp. 133–150.

[40]   Oscar Agertz et al. "Fundamental differences between SPH and grid methods". In: *Monthly Notices of the Royal Astronomical Society* 380.3 (Sept. 21, 2007), pp. 963–978. DOI: `10.1111/j.1365-2966.2007.12183.x`.

[41]   Omer San and Romit Maulik. "Stratified Kelvin–Helmholtz turbulence of compressible shear flows". In: *Nonlinear Processes in Geophysics* 25 (2018), pp. 457–476. DOI: `10.5194/npg-25-457-2018`.

[42]   Jin Seok Park et al. "Comparative Study of Shock-Capturing Methods for High-Order CPR: MLP and Artificial Viscosity". In: *Proceedings of The Eighth International Conference on Computational Fluid Dynamics (ICCFD8)*. July 14-18. The Eighth International Conference on Computational Fluid Dynamics. Chengdu, Sichuan, China, 2014, ICCFD8-2014–0067.

[43]   Paulo Cesar Philippi et al. "Kinetic Projection and Stability in Lattice-Boltzmann Schemes". In: *Proceedings of the CILAMCE 2015 Conference*. Conference Paper. Nov. 2015. DOI: `10.20906/CPS/CILAMCE2015-0398`.

[44]   Andrew W. Cook and William H. Cabot. "Hyperviscosity for shock-turbulence interactions". In: *Journal of Computational Physics* 203.2 (Mar. 2005), pp. 379–385. DOI: `10.1016/j.jcp.2004.09.011`.

[45]   Hojun You, Seonghun Cho, and Chongam Kim. *Department of Mechanical and Aerospace Engineering, Seoul National University, Korea. Jan. 6-7, 2018*. Presented at AIAA SciTech HiOCFD5, Kissimmee, FL, 2018. Aerodynamic Simulation & Design Lab, SNU. 2018.

[46]   Philip E. Johnson and Eric Johnsen. "A recovery-assisted DG code for the compressible Navier-Stokes equations". In: *5th International Workshop on High-Order CFD Methods*. Scientific Computing and Flow Physics Laboratory, Mechanical Engineering Department, University of Michigan, Ann Arbor. Kissimmee, Florida, Jan. 2017.

[47]   Qilin Lu and Z. J. Wang. "CI2 – Inviscid Strong Vortex-Shock Wave Interaction". In: *Presented at 5th International Workshop on High-Order CFD Methods*. University of Kansas. 2017.

[48]   Chongam Kim, Hojun You, and Seonghun Cho. "CI2 – Inviscid Strong Vortex-Shock Wave Interaction". In: *AIAA SciTech HiOCFD5, Kissimmee, FL, 2018*. Aerodynamic Simulation & Design Lab., SNU. Department of Mechanical and Aerospace Engineering, Seoul National University, Korea. Jan. 2018.

[49]   A. N. Kolmogorov. "The local structure of turbulence in incompressible viscous fluid for very large Reynolds numbers". In: *Dokl. Akad. Nauk SSSR* 30 (1941), pp. 301–305.

[50]   A. N. Kolmogorov. "A refined Kolmogorov theory of turbulence". In: *Journal of Fluid Mechanics* 13.1 (1962), pp. 82–85.

[51]   K. R. Sreenivasan. "Kolmogorov's Local Structure Theory of Turbulence". In: *Annual Review of Fluid Mechanics* 23 (1991), pp. 539–600.

Appendices

A Cascade of Kolmogorov

To better understand artificial viscosity, it is helpful to begin by addressing the fundamentals of turbulence in fluid flows. Turbulence is a complex phenomenon that occurs in fluid flows at high Reynolds numbers. The flow can be divided into two distinct components: a regular mean flow and chaotic turbulent flow. When the fluid moves at high speeds, some of the energy from the mean flow is transferred to the turbulent flow. In turbulent flow, this additional energy generates eddies, which are typically of a similar size to the flow itself and exhibit anisotropic characteristics. However, due to their instability, these eddies gradually dissipate while imparting some of their kinetic energy to smaller eddies. This energy transfer process continues in the form of a cascade, creating progressively smaller eddies until reaching a specific scale known as the Kolmogorov scale [49]. The Kolmogorov scale represents a characteristic eddy size where viscous effects become dominant. At this scale, the kinetic energy of the eddies is converted into heat through viscous dissipation. This means that turbulent energy is eventually dissipated as heat through the viscous friction of the fluid, contributing to the overall dissipation of turbulent energy [50, 51].

B Initial Flow Condition : Inviscid Strong-Vortex Interaction

The initial conditions for the test case can be derived simply based on the upstream conditions as well as the parameters for the vortex strength, denoted as M_s, and the shock, denoted as M_v. Consequently, it becomes possible to compute the downstream conditions, represented by the variables (ρ_d, u_d, v_d, p_d), using the equation for the stationary normal shock condition, along with the given upstream conditions

$$\begin{bmatrix} \rho_u \\ u_u \\ v_u \\ p_u \end{bmatrix} = \begin{bmatrix} 1.0 \\ M_s\sqrt{\gamma} \\ 0.0 \\ 1.0 \end{bmatrix}, \tag{4.2}$$

as follows

$$\frac{\rho_u}{\rho_d} = \frac{u_d}{u_u} = \frac{2 + (\gamma - 1)M_s^2}{(\gamma + 1)M_s^2}, \tag{4.3a}$$

$$\frac{p_d}{p_u} = 1 + \frac{2\gamma}{\gamma + 1}(M_s^2 - 1), \tag{4.3b}$$

$$v_d = 0, \tag{4.3c}$$

with $M_s = 1.5$ and $\gamma = 1.4$.

The calculation domain can now be initialized outside the vortex using previously computed upstream and downstream conditions. The current goal is to determine the velocity field inside the vortex by superimposing the upstream velocity conditions (u_u, v_u) with the tangential velocity field (v_θ) provided in the initial conditions. Here, r represents the distance from the vortex center located at $(x_c, y_c) = (0.25, 0.50)$, while the parameters $(a, b) = (0.075, 0.175)$ are used. The value v_m corresponds to the maximum tangential velocity, which is achieved at a distance of $r = a$. $M_v = v_m/\sqrt{\gamma}$ is used as a measure of the vortex's strength, and in this calculation, M_v is set to 0.9. The vortex rotates counterclockwise with an angular velocity defined as follows

$$v_\theta(r) = \begin{cases} v_m \frac{r}{a} & \text{if } r \leq a, \\ v_m \frac{a}{a^2 - b^2}\left(r - \frac{b^2}{r}\right) & \text{if } a \leq r \leq b, \\ 0 & \text{if } r > b. \end{cases} \tag{4.4}$$

As a result,

$$u_{vor}(r) = u_u + x - v_\theta, \tag{4.5a}$$
$$v_{vor}(r) = v_u + y - v_\theta. \tag{4.5b}$$

The temperature field inside the vortex, denoted as $T_{vor}(r)$, can be determined using the following differential equation obtained from the normal momentum equation with the centripetal force,

$$\frac{dT_{vor}(r)}{dr} = \frac{\gamma - 1}{R\gamma} \frac{v_\theta(r)^2}{r}. \tag{4.6}$$

The temperature $T_{vor}(r)$ can be determined by performing an integration of this expression, taking into account the condition $T_{vor}(b) = T_u$ where the temperature at the upstream state of T_u is determined by the ideal gas law $p = \rho R T$ with $R = 1$,

$$T_{vor}(r) = \int_r^b \frac{\gamma - 1}{R\gamma} \frac{v_\theta(r')^2}{r'} \, dr'. \tag{4.7}$$

Finally, by using the isentropic relations, the density and pressure inside the vortex can be calculated. ρ_u, p_u, and T_u represent the upstream conditions as referenced

$$\rho_{vor}(r) = \rho_u \left(\frac{T_{vor}(r)}{T_u}\right)^{\frac{1}{\gamma - 1}}, \tag{4.8a}$$

$$p_{vor}(r) = p_u \left(\frac{T_{vor}(r)}{T_u}\right)^{\frac{\gamma}{\gamma - 1}}. \tag{4.8b}$$

The downstream and within-vortex initial conditions can be deduced solely from the upstream conditions, as shown in this section.