Master of Science Thesis

A study of an artificial viscosity technique for high-order discontinuous Galerkin methods

Marc Cruellas Bordes

February 22, 2019

Aerodynamics

Faculty of Aerospace Engineering

TU Delft

Delft University of Technology

A study of an artificial viscosity technique for high-order discontinuous Galerkin methods

Master of Science Thesis

For obtaining the degree of Master of Science in Aerospace Engineering
at Delft University of Technology

Marc Cruellas Bordes

February 22, 2019

Faculty of Aerospace Engineering · Delft University of Technology

Delft University of Technology

DELFT UNIVERSITY OF TECHNOLOGY
DEPARTMENT OF AERODYNAMICS

The undersigned hereby certify that they have read and recommend to the Faculty of Aerospace Engineering for acceptance the thesis entitled **"A study of an artificial viscosity technique for high-order discontinuous Galerkin methods"** by **Marc Cruellas Bordes** in fulfillment of the requirements for the degree of **Master of Science**.

Dated: February 22, 2019

Supervisors:

Prof.dr. S. Hickel

Prof. T. Magin

Dr.ir. M.I. Gerritsma

Dr.ir. M. Möller

Abstract

Prediction of heat loads during hypersonic re-entry is of great interest in space exploration and more recently in the topic of space debris as well. To date, there is no accurate method to reproduce either experimentally or numerically the physics in re-entry conditions and large safety factors need to be used as a consequence. On the numerical side, the recent developments in a number of areas have helped the creation of new tools to tackle the challenges of re-entry simulation. This work focuses on a high-order discontinuous Galerkin method, which has potential to improve results achieved with state-of-the-art finite volume methods. One important weakness of discontinuous Galerkin methods is the occurrence of Gibbs-type oscillations around discontinuities. A number of methods have been developed to remove these spurious oscillations, and this work chose to focus on artificial viscosity for its simplicity, adaptability, and maturity. The aim was to obtain an accurate heat flux prediction at the surface of a cylinder in a high Mach number flow, a challenging task achieved only a few times with discontinuous Galerkin methods.

The artificial viscosity method used in this work contains three user-defined parameters which specify the magnitude, location and width of the artificial viscosity. A parametric study of these parameters on a steady inviscid wedge test case at Mach 2 showed that tuning their values is a non-trivial task and it appeared that in some cases shock smoothness and shock sharpness need to be traded-off. Furthermore, it was observed that sufficient refinement in mesh size or polynomial order was needed to obtain satisfying results. The Sod problem [80] was considered to test the method in an unsteady inviscid test case. Good results were obtained, although the shock was better resolved than the contact discontinuity. The method was also tested on another unsteady inviscid test case, the Shu-Osher problem [79]. The latter is more challenging due to the shock strength varying in time. Specifying the artificial viscosity parameters was challenging since the set had to account for variation of the shock strength in time. The test case also highlighted that with strong shocks, spurious oscillations can be large enough that artificial viscosity is not sufficient to keep the simulation stable. Therefore, the need for a method that keeps variables within their physical bounds became clear. For both unsteady problems, the choice of inviscid flux scheme was found to have little effect on the solution at high-orders. This observation is consistent with discontinuous Galerkin results from the literature.

Prediction of the heat flux on the surface of a half-cylinder in a hypersonic flow was attempted. However, this was unsuccessful and the Mach number was eventually reduced to 3.25. Strong non-physical oscillations appeared in the heat flux profile even with smooth contour plots and pressure at the surface of the half-cylinder. The key to obtain a smooth heat flux profile was to ensure no artificial viscosity at all is inserted in the boundary layer. Despite a smooth heat flux profile, a symmetric mesh was needed to obtain a symmetric profile. Overall, the heat flux results showed that it is possible to simulate this challenging test case with discontinuous Galerkin and a combination of simple methods, including the Lax-Friedrichs flux and a basic smoothing of the artificial viscosity field.

A new strategy to tune the user-defined parameter specifying the location of the artificial viscosity was implemented and tested on the Shu-Osher and half-cylinder test cases. It was found to be greatly beneficial in terms of stability and user-friendliness, but did not fully eliminate the issues in simulating these challenging test cases. This strategy allowed to run the half-cylinder test case beyond the Lax-Friedrichs flux and a polynomial order of 1. The heat flux profile with SLAU came very close to that with Lax-Friedrichs, whereas the difference was significant between first and second orders.

Acknowledgements

I would like to start by thanking Prof. Thierry Magin for his supervision throughout this project. From the early discussions to define the topic of my thesis to the long discussions throughout the project, he always took the time to address my questions and the challenges I faced. I am very grateful for his availability and for the feedback he offered. Furthermore, I would like to thank Dr. Alessandro Turchi for his guidance and advice, for his pragmatic opinions when needed, as well as for his help installing the code. Speaking of code, I am deeply grateful to Dr. Pierre Schrooyen for his help with Argo, which included welcoming me at Cenaero a number of times, and taking the time to explain how the code and the methods work. I also highly appreciated his involvement in the project and his feedback. As a general comment to Prof. Thierry Magin, Dr. Alessandro Turchi and Dr. Pierre Schrooyen, I would like to thank them for the interest they brought to our weekly meetings, which made me feel part of their team, while at the same time gave me the room to take my own decisions and lead the project in the direction I was most excited to. I would also like to thank Stefan Hickel, first for accepting to supervise an external topic at an external institution, but mostly for his guidance and feedback whether via Skype, email, or in person. I greatly appreciated his advice, particularly on resolving issues preventing simulations to give me satisfying results. Then, although his involvement in this thesis was more limited, I would like to thank Dr. Koen Hilleweart as well. I appreciated his interest in the project, explanations on convergence and HPC concepts, as well as his advice for some of the challenges I faced.

I would also like to mention all STPs at VKI with whom I shared both long, frustrating days and great, exciting times. I am very grateful to all for the great atmosphere they helped create. To all my friends at TU Delft, I thank them for all the time we shared before this project, as well as for their warm welcome at every occasion I had to spend a day or two in Delft. To my friends in Brussels, it felt nice to be back for longer than a weekend after some years moving around and I am very grateful for the good fun times we had. I would also like to thank my family for their support, and Valerie Garcia for always being there through the highs and the lows of this project.

M. Cruellas Bordes
von Karman Institute for Fluid Dynamics, Sint-Genesius-Rode, February 2019

Contents

List of Figures

List of Tables

List of Symbols

Acronyms

AUSM	Advection Upstream Splitting Method
AV	Artificial Viscosity
DAS	Debris Assessment Software
DG	Discontinuous Galerkin
DRAMA	Debris Risk Assessment and Mitigation Analysis
ENO	Essentially Non-Oscillatory
ESA	European Space Agency
FDS	Flux Difference Splitting
FEM	Finite Element Method
FVM	Finite Volume Methods
FVS	Flux Vector Splitting
GMRES	Generalised Minimum Residual
HWENO	Hermite Weighted Non-Oscillatory
ILU	Incomplete LU factorisation
IP	Interior Penalty
LES	Large Eddy Simulations
LFS	Local Fourier Spectral
MPI	Message Passing Interface
NASA	National Aeronautics and Space Administration
Open MP	Open Mulit-Processing
ORSAT	Object Re-rentry Survivial Analysis Tool
RANS	Reynolds Averaged Navier-Stokes
RKDG	Runge-Kutta Discontinuous Galerkin
SCARAB	Spacecraft Atmospheric Re-entry and Aero-thermal Breakup
SLAU	Simple Low dissipation AUSM
SV	Spectral Viscosity
TPS	Thermal Protection System
TV	Total Variation
TVB	Total Variation Bounded
TVD	Total Variation Diminishing
VKI	von Karman Institute for Fluid Dynamics
WENO	Weighted Essentially Non-Oscillatory

Greek symbols

δ_{ij}	Kronecker delta	–
γ	Ratio of specific heats	–
κ	Artificial viscosity parameter defining the interval	–
κ_T	Thermal conductivity	$W/(m \cdot K)$
λ_{max}	Largest wave speed	m/s
μ	Dynamic viscosity	$Pa \cdot s$
Ω	Problem domain	m, m^2, m^3
ρ	Density	kg/m^3
ε	Artificial viscosity	m^2/s

Roman symbols

$\mathbf{u}$	Velocity vector with components u,v,w	m/s
$\mathbf{x}$	Vector of Cartesian coordinates x,y,z	m
c	Speed of sound	m/s
c_p	Specific heat at constant pressure	$m^2/(K \cdot s^2)$

E	Total internal energy	J/kg
H	Total enthalpy	J/kg
h	Size of a mesh element	m
k	Artificial viscosity parameter defining its magnitude	–
M	Mach number	–
P	Static pressure	Pa
p	Polynomial order	–
Pr	Prandtl number	–
q	Heat flux	W/m^2
Re	Reynolds number	–
s_0	Artificial viscosity parameter defining the threshold	–
S_e	Measure of smoothness in the solution calculated by the shock sensor	–
s_e	Scaled measure of smoothness of the solution, equal to $\log_{10} S_e$	–
T	Temperature	K
t	Time	s

Sub- and Superscripts

$\{\cdot\}^n$	Corresponds to $\{\cdot\}$ at time n
$\{\cdot\}_0$	Corresponds to the total quantity of $\{\cdot\}$
$\{\cdot\}_\infty$	Freestream value of $\{\cdot\}$
$\{\cdot\}_e$	Corresponds to $\{\cdot\}$ on a single mesh element
$\{\cdot\}_L$	Corresponds to the left state of $\{\cdot\}$ on an interface
$\{\cdot\}_R$	Corresponds to the right state of $\{\cdot\}$ on an interface
$\{\cdot\}_t$	Derivative of $\{\cdot\}$ with respect to time
$\{\cdot\}_x$	Derivative of $\{\cdot\}$ with respect to coordinate x
$\{\cdot\}_{ed}$	Corresponds to $\{\cdot\}$ at the edge of the boundary layer

1 Introduction

1.1. Space debris and atmospheric re-entry

At the beginning of April 2018, the re-entry of the Chinese space station Tiangong 1 into Earth's atmosphere made headlines. It is not uncommon for space debris to re-enter the atmosphere, but Tiangong 1 was largely covered in the news due to its size and uncontrolled re-entry. Although the probability of the spacecraft causing human casualties was very low, how much of the station would survive the re-entry and where would those surviving pieces land was uncertain. This is partly why there is a growing interest in the problem of space debris. Ideas to remove these objects are being developed and studied, but attention is being paid into improving re-entry predictions as well. There are about 27,000 tons of space debris that re-entered the atmosphere between 1957 and 2002 [41]. Of these, about 100 large objects re-enter our planet's atmosphere each year and it is estimated that 10 to 40% of their mass survive impact on ground or water [1]. Fortunately, there are no known casualties due to space debris to date [1].

Due to the small but non-zero risk that space debris pose to the population, the casualty risk for spacecraft which are due to re-enter the Earth's atmosphere at their end-of-life must remain under 10^{-4} [25]. Estimation methods are necessary to ensure compliance with this requirement, especially for medium and large size spacecraft [25]. A number of tools exist, namely NASA's Debris Assessment Software (DAS) and Object Re-entry Survival Analysis Tool (ORSAT), and ESA's Spacecraft Atmospheric Re-entry and Aero-thermal Breakup (SCARAB) and Debris Risk Assessment and Mitigation Analysis (DRAMA) [51]. For tools such as these to provide more precise results, one needs accurate simulations of the aerodynamic and thermal loads on the spacecraft.

Accurate simulation of aerodynamic and thermal loads is also crucial in space exploration. Spacecraft need to be able to survive the harsh environments occurring during hypersonic re-entry to protect the payload. To cope with the extremely large temperatures characteristic of decent in a planet's atmosphere such as Mars, Jupiter, or Earth, a thermal protection system (TPS) is used. The TPS is a layer of material that is usually either reusable or ablative. The Space Shuttle's heat shield made of ceramic tiles is an example of reusable TPS. Ablative materials, on the other hand, protect the spacecraft from the heat loads by degradation. They were used in missions with high entry velocities such as the Mars Science Laboratory or Apollo. An under-designed TPS would lead to mission failure as the rest of the spacecraft would not be able to cope with the high temperatures. However, over-designing the TPS should be avoided too as it increases the weight of the spacecraft and thus limits the amount of useful payload that can be carried on board. Therefore, accurate simulation tools are needed to optimise the design of the TPS.

These simulations are difficult due to the complex multi-physical phenomena that occur during re-entry. Even today, predicting accurately the heat load the TPS needs to withstand remains a challenging task. On one hand, experiments allow to reproduce different aspects of the flow conditions separately. For example, hypersonic cold wind tunnels can be used to investigate the aerodynamics of the spacecraft and Induced Coupled Plasma wind tunnels can reproduce the high enthalpy conditions to study the material response [75]. On the other hand, numerical codes are based on simplified models that are not able to simulate the full conditions either. Nowadays, the large majority of heat flux predictions are based on second-order finite volume methods which require a carefully designed mesh and a careful choice of several features such as the inviscid flux scheme and the limiting used [15]. However, recent results by Ching et al. [15] suggest high-order

methods could be less sensitive to part of these features. The next section will introduce high-order methods in more detail.

1.2. High-order methods and the discontinuous Galerkin approach

To present high-order methods, the first step is to explain what high-order means. The order r of a method is related to the error by

$$e \leq Ch^r \tag{1.1}$$

where C is a constant, h is the cell size and e is the error. In the aerospace community, there seems to be a consensus that high-order corresponds to $r \geqslant 3$ [88]. In industry, second-order methods are widely used as they benefit from decades of development and have now become robust and efficient. However, interest in high-order methods has increased considerably in the last 20 years because increasing the order (refining in r) can be more efficient than refining in h to achieve a given level of accuracy [88]. Wang et al. [88] also mention that low-order methods are insufficient for a number of applications, such as the computation of vortices at the tip of helicopter's rotors. Despite the potential benefits of and the recent progress in high-order methods, Wang et al. [88, p843] identify four critical areas that need improvement:

1. *Mesh generation:* generating unstructured high-order meshes near curved geometries represented by high-order polynomials currently lacks robustness. The challenge is in cells that can overlap each other near curved boundaries.

2. *Hp-adaptations:* effective methods for hp adaptation (adapting the size h and polynomial order p locally depending on the problem needs) exist but improvements are needed in 3D. It would also be beneficial to develop adaptation methods free of user-specified parameters and that can estimate the error of key flow parameters.

3. *Time integration for Reynolds Averaged Navier-Stokes (RANS) and hybrid RANS/Large Eddy Simulations (LES) approaches:* problems at high Reynolds number suffer from stiff matrices caused by the strong anisotropy of meshes near boundaries. Scalable solvers along with efficient preconditioners need to be designed for massively parallel computers.

4. *Shock capturing:* the two main techniques in shock capturing are artificial viscosity and limiting (including essentially non-oscillatory type of methods). The first requires to set user-specified constants and the second reduces the order of the method. A robust approach free of user-specified parameters, with good convergence properties and that preserves the accuracy away from discontinuities remains to be developed.

In the category of high-order methods, discontinuous Galerkin (DG) methods are a special variation of finite element methods (FEM) combining advantages of FEM and finite volume methods (FVM) [21]. Reed & Hill [73] were the first to introduce the method, to solve the neutron transport equation. It then progressed in two paths: the first focused on hyperbolic problems and resulted among others in the appearance of the popular Runge-Kutta Discontinuous Galerkin (RKDG) method, the second tackled elliptic and parabolic problems and saw the appearance of the interior penalty (IP) method and its many variations [21]. The compressible Navier-Stokes equations started to be solved with DG methods in the late 90s [21]. Since the early 2000s, the method has received considerable interest [21].

Important advantages of DG methods is that they allow to obtain arbitrarily high orders of accuracy and are well suited for unstructured meshes and complex geometries [8, 35]. They have several other advantages. For example, they use compact stencils which lead to sparse block matrices well adapted for parallel computing. Low dispersion and dissipation are another of their benefits [56]. DG methods also have their set of drawbacks. Mainly, they suffer from large memory requirements and high computational cost. Since no continuity is imposed on the cell interfaces, the number of nodes are doubled at the interfaces and the degrees of freedom increase as a consequence [31, 75]. Furthermore, the appearance of Gibbs oscillations around discontinuities in the solution makes the simulation of supersonic and hypersonic flows more challenging. These spurious oscillations are clearly unwanted since they ruin the accuracy of the results around a shock and can become so strong that they crash the simulation. Various treatments have been developed to tackle this issue. However, this remains a key area needing improvement, as mentioned above.

1.3. Shock treatment in high-order methods

To remove spurious oscillations from the solution, shock fitting and shock capturing are the two main strategies that exist. This section will present an overview of both strategies, starting with shock fitting.

1.3.1. Shock fitting

Shock fitting consists of adapting the mesh to be perfectly aligned with the shock and treating the shock as an internal boundary condition. The flow conditions on both sides of this boundary are computed using the Rankine-Hugoniot jump relations [67]. Moretti and his collaborators contributed significantly to the development of shock fitting techniques between the 60s and 80s on structured grids [61, 67]. This development occurred following two main strategies. The first is called boundary shock fitting and is simpler to code but struggles with interacting shocks. The second, floating shock fitting, is similar to the first but can handle shock interactions better [67]. Shock-fitting for unstructured grids was developed later, with the rising use of unstructured grids in CFD about 30 years ago [67].

Theoretically, the shock fitting technique allows to resolve shocks accurately. However, there are a few issues, mainly related to the mesh, that make it difficult to use in practise. Firstly, it is generally not trivial to obtain a perfect alignment of the mesh with the discontinuity. Even if the physical location of the shock is known, this might not correspond to its numerical location. These misalignments can cause the algorithm to fail to converge to the right solution. Secondly, moving shocks create an additional difficulty since the mesh needs to "follow" the discontinuity. A robust meshing algorithm is needed to handle features such as local mesh refinements as the shock moves. Finally, it is difficult to handle complex shock interactions with it. This leads some shock fitting codes to use the alternative technique, shock capturing, to handle the most complex shocks [67]. For these reasons, shock fitting is rarely used nowadays and receives little attention from the scientific community.

1.3.2. Shock capturing

A much more popular approach to shock fitting, although less accurate in theory, is shock capturing. In the context of DG, shock capturing is still an active area of research, and it encompasses a whole range of methods. To avoid spurious oscillations, one option could be to simply lower the polynomial order to zero (and refine the mesh to reclaim some lost accuracy), however this will not be discussed here because one key reason to use high-order methods is precisely to avoid such refinements and low-order accuracy. The remainder of this section will present an overview of the other techniques available.

Limiting

Limiters eliminate spurious oscillations by lowering the order of accuracy near discontinuities. They do so by modifying either the fluxes or slopes of the solution, and are thus given the names of flux and slope limiters, respectively. The latter category is more popular in DG [4]. An often used approach to limiters is the construction of schemes which satisfy the Total Variation Diminishing (TVD) property [6]. This property ensures that the total variation $TV(u) = \sum_i |u_{i+1} - u_i|$ does not increase as the simulation progresses. In 1989, Cockburn & Shu [16] proposed a popular TVD scheme in which the limiter switch is based on a minmod function. However, it is well established that TVD schemes decrease the accuracy to first order at smooth extrema [16]. Therefore, they proposed an alternative scheme which is Total Variation Bounded (TVB). In TVB schemes, the total variation only needs to remain under a given constant rather than being forced to decrease at each time step. However, the scheme has a constant which needs to be defined by the user and can be particularly difficult to tune [12, 94]. Two alternatives by Biswas et al. [9] and Burbeau et al. [12] avoided this issue but were found to be too dissipative [71, 94]. Overall, limiters are popular methods but are typically difficult to generalise to any type of mesh elements and order of accuracy [3, 14, 44]. They are effective in removing spurious oscillations but tend to damage detailed structures in the solution [92].

Reconstruction methods

To avoid decreasing the order of accuracy near shocks, essentially non-oscillatory (ENO) methods appeared in 1987 thanks to the work of Harten and his collaborators [33]. Compared to TVD/TVB methods which use a fixed stencil, the main idea behind ENO is to have an adaptive stencil that uses information in smooth regions to reconstruct high-order polynomials up to discontinuities [78]. Weighted essentially non-oscillatory (WENO) methods appeared later as an improvement to ENO. The key difference is that the newer approach uses a number of potential stencils rather than just one [4]. However, WENO methods have the disadvantage

of using large stencils. HWENO methods, where H stands for Hermite, were developed to address this issue. Examples can be found in the papers of Qiu & Shu [70] and Luo et al. [54].

The work by Qiu & Shu [70] was then extended to unstructured grids by Zhu & Qiu [95]. Luo et al. [54] propose a similar strategy for an unstructured 2D mesh. Zhu et al. [96] and Du & Li [22] presented each another HWENO variation recently, indicating that the development of compact and high-order HWENO methods for 1D and 2D cases is still an active area of research. Applications in three dimensions and curved elements still remain largely unexplored. Xia, Luo, and their collaborators have addressed these two last aspects in a series of papers [52, 55], however their test cases are limited to subsonic and transonic cases.

WENO and HWENO methods were frequently combined with the explicit Runge-Kutta Discontinuous Galerkin method, which suffers from a limitation for marching in time. This issue has been addressed more recently with implicit HWENO schemes [36, 52, 90, 91]. Generally, they have been able to reduce computational time while maintaining the accuracy of their equivalent explicit methods.

Clearly, reconstruction methods also suffer from a generalisation issue. Specific methods need to be designed to address different aspects, and their latest developments remain largely untested at supersonic and hypersonic speeds. Nevertheless, WENO type of methods are being actively improved. The reader is referred to [77] for a more extensive overview of the current developments.

Artificial viscosity

VonNeumann & Richtmeyer [85] proposed in 1950 to add viscous terms to the hydrodynamic equations in order to resolve shocks in numerical simulations. The extra terms spread the shock over a number of cells such that it can be resolved by the numerical method. Their approach, now commonly called artificial viscosity (AV), has been adapted to high-order DG methods successfully. Compared to limiters and reconstruction methods, artificial viscosity is a flexible shock capturing strategy. The user needs to tune manually a few parameters, but variations of the AV methods do not need to be derived for different aspects such as mesh topology, order of accuracy, or problem dimension. One possible approach is the dilation-based (i.e. based on the velocity divergence) artificial viscosity, proposed recently by, for example, Moro et al. [62] and Yu et al. [93]. Another option is to use a Laplacian or physical AV term, as studied by Persson & Peraire in [69]. By introducing a scaling based on the mesh element size h and polynomial order p, they were able to capture shocks at sub-cell resolution. Their method was later improved by Barter & Darmofal [5], who established that by applying artificial viscosity on each mesh element in a discrete fashion, oscillations can appear. They thus proposed to smooth the AV field with an additional partial differential equation. Both smoothing and h/p scaling are now standard features in AV methods.

Hypersonic simulations, especially those for which the heat flux is a quantity of interest, have been little investigated in the context of DG. As mentioned in Ching et al. [15], good results have been obtained in [5, 11, 13, 68]. These are all based on AV except for [68] where a TVB limiter is used and less convincing results are obtained. Arguably, the best results are currently obtained by Ching et al. [15]. Using an artificial viscosity method, they manage to obtain accurate heat flux predictions for two challenging hypersonic problems and, due to the coarseness of the meshes they used, at fewer degrees of freedom than equivalent results obtained with state-of-the-art finite volume methods.

The good progress of AV methods for DG does not mean that little is left to improve. One of their main weakness is the presence of parameters to be tuned manually. Poor tuning leads to poor accuracy around the shock at best and to unstable simulations at worst. This remains a challenge to overcome and can turn a seemingly simple method rather complex to use in practise. In addition, the performance of AV methods on unsteady problems is less clear since the majority of test cases shown are steady-state solutions. It is expected that AV parameters can be especially difficult to tune for unsteady problems where the shock strength varies in time. Furthermore, it remains to be proved that results such as those obtained in Ching et al. [15] can be obtained on fully unstructured meshes. Their plots of the AV field show that it deteriorates where the alignment of the mesh and the shock is the worst. Therefore, it is expected that fully unstructured meshes would result in additional difficulties in obtaining an accurate shock and heat flux. Barter & Darmofal [5] do obtain good results on very fine hybrid meshes (structured in the boundary layer region, unstructured elsewhere). To benefit from the advantages of high-order methods, such results remain to be reproduced on coarser meshes. Finally, the results by Ching et al. [15] on a double cone problem show that there are still cases where AV methods lack robustness. More issues could arise if chemistry is taken into account. The performance of AV methods with both chemical reactions and high Mach number flows remains largely unexplored in DG.

Filtering
A different approach to what has been presented before is to handle discontinuities by stabilising the solution in the frequency domain. Spectral viscosity (VS) was introduced by Tadmor [82]. Gelb & Tadmor [26] presented later an enhanced SV method in which a post-processing step is added to recover accuracy in the solution. Oberai & Wanderer [65] proposed a variation of these SV methods where different viscosities are applied to the coarse and fine scales of the solution, rather than only on the fine scales. For the Burgers' equation, they found that their multiscale viscosity method was more accurate. More recently, Lopez-Morales & Jameson [53] used Local Fourier Spectral filters to stabilise the solution. Their argument is that it is the energy cascade where large scales transfer their energy to small scales which causes high-order methods to become unstable. Indeed, high-order methods have low dissipation which means the smallest scales could not be dissipated before they become aliased. The goal of their method is therefore to filter the small scales that could result in aliasing.

Overall, filtering is mainly applied in spectral methods and seems to be less effective than limiting or artificial viscosity to remove spurious oscillations. Filtering has been applied to DG methods as well but only in a limited number of cases, such as [10, 59, 60]. Therefore, the method has yet to prove that it can be seen as a mature alternative to limiting and artificial viscosity techniques for DG methods.

Posterior solution updating
The main idea in posterior solution updating is to use a finite volume method locally to resolve discontinuities. This recent technique was introduced in Dumbser & Loubère [24] and later extended to unstructured grids in [23]. The key difference between this method and all other strategies discussed is that this method goes back to the previous time step t^{n-1} to recompute a stabilised solution for t^n, whereas other shock capturing methods actually stabilise the solution at time t^{n+1}. The additional computational cost would make this method mainly worth using when going back one time step allows to compute a solution which would crash otherwise. However, with challenging test cases currently being resolved with the AV methods presented above, the case for the posterior solution updating approach might become more and more difficult.

1.4. Thesis objectives and outline

Due to its simplicity, maturity, and adaptability, the focus of the research work will be on artificial viscosity. The project will use the multi-physics CFD platform Argo, based on a high-order discontinuous Galerkin method. The aim is to study how the method performs in Argo, propose an improvement based on the weaknesses identified, and use that to predict the heat flux profile at the wall of a cylinder in a supersonic flow. The first part is based on a number of test cases of increasing difficulty. Although an analytical or reference solution is available for comparison, the results are compared with literature where relevant. This not only allows to assess how the method performs with respect to others, but also allows to ensure that issues appearing in the results are not specific to Argo. This last step is of particular interest with Argo because the computational tool has had only limited testing at these high Mach number regimes. A number of contributions are made in this work. Firstly, the performance of DG with artificial viscosity is investigated on unsteady problems, which have received much less attention in literature. Secondly, the wall heat flux is computed for a cylinder in a high Mach number flow. This has been achieved only in a limited number of cases with DG, with varying success. This work does so in different conditions than used in literature, but is limited to supersonic rather than hypersonic flow. In addition, this work highlights the difficulties in achieving such results, information much lacking in literature. Finally, this work proposes a new method to tune automatically one of the user-defined AV parameters and presents its benefits. The scope of this work is limited to perfect gases, with chemistry, radiation, and material ablation not being considered.

The next chapter will provide the reader with the necessary theoretical background. With the foundations covered, artificial viscosity is applied on a supersonic wedge test case. This simple test case already highlights key trends in the use of the method and certain limits and challenges. This is covered in Chapter 3. Chapter 4 then focuses on unsteady problems, which provide new insights on AV methods. The information gathered in chapters 3 and 4 is applied to a challenging half-cylinder test case. A number of modifications are implemented before a satisfying heat flux profile is obtained. The results are provided and discussed in Chapter 5. Chapter 6 presents the new method to remove one of the user-defined AV parameters and highlights its benefits. Finally, Chapter 7 summarises the important findings in this thesis and provides a number of ideas to explore in future work.

2

Numerical methods and physical modelling

This chapter aims to describe the foundations on which this work is built on. Firstly, the numerical tools used in this work are briefly introduced. Secondly, the equations used to model the flow are presented. Thirdly, the discontinuous Galerkin method is detailed. It was already briefly introduced in terms of its advantages and disadvantages in the previous chapter. Now, the core of the method will be presented. Finally, the artificial viscosity technique used in this work is described.

2.1. Numerical tools

In this section, the two programs used in this work, Argo and Gmsh, are briefly presented. A number of important features will be highlighted due to their relevance for the next chapters.

2.1.1. Argo

Argo is a multi-physics CFD platform initially developed by Hilleweart [35]. The code is now developed at Cenaero, an applied research centre in Belgium. It is based on a discontinuous Galerkin method, to be presented in Section 2.3, and can handle both structured and unstructured meshes. Furthermore, it is designed to run on parallel architectures, using a hybrid of Open Multi-Processing (Open MP) and Message Passing Interface (MPI) libraries. Two modules in the code of particular interest for this work are:

1. *DGFluid:* This module was created to simulate incompressible and compressible flows. It is based on conservative variables, and of relevance to this work, contains a clipping method ensuring the physical variables remain within physical bounds.

2. *DGAblation:* Developed by Schrooyen [75], this module was based on DGFluid and created in order to simulate material ablation problems. A state-of-the-art approach was used to simulate the material ablation and the flow in the same computational domain. It is capable of simulating multi-species compressible flows and chemical reactions, among others. Primitive variables are used for simplicity in the derivation of the Jacobian for implicit time integration [75].

The work initially started on the DGAblation module and later used parts of DGFluid in order to benefit from the clipping. This implied a change from primitive to conservative variables, with the chosen set clearly stated throughout this work to avoid confusion. For further details about Argo, the reader is referred to [35, 75].

2.1.2. Gmsh

The meshes in this work were generated with Gmsh, described in detail in [27]. Unstructured meshes were built using the Delauney algorithm. As mentioned in Section 1.2, generating high-order meshes around curved geometries can be challenging. Gmsh contains high-order tools, which were used in order to obtain valid elements near curved boundaries. The program is also used for post-processing results, being capable of high-order visualisation.

2.2. Compressible flow equations

As mentioned before, simulating re-entry conditions numerically is difficult due to the complex multi-physical phenomena involved. Several typical challenges of hypersonic re-entry flows consist of thin shock layers which can lead to merging boundary and shock layers, an entropy layer interacting with the boundary layer, large boundary layers which can effectively change how the flow outside the boundary layer perceives the size of the body, high temperatures causing the flow to become chemically reacting and to emit thermal radiation, and low density effects which can require to use velocity and temperature slip conditions at the wall or even break down the continuum assumption used in most aerodynamic problems [38]. This work focuses on altitudes where the continuum assumption holds, and a frozen mixture of gas with constant thermodynamic and transport properties is considered. Mainly, the focus is on the Euler equations which are adequate for investigating high Mach number effects, and on the compressible Navier-Stokes which allow to consider viscous and thermal conduction effects as well [38]. Both sets of equations can be written in the general form

$$\frac{\partial \mathbf{U}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{U}) = \nabla \cdot \mathbf{F}_v(\mathbf{U}, \nabla \mathbf{U}) + \mathbf{S}(\mathbf{U}, \nabla \mathbf{U}), \tag{2.1}$$

where $\mathbf{U}$ is the state vector of conservative variables, $\mathbf{F}$ is the inviscid flux, the viscous flux is given by $\mathbf{F}_v$, and $\mathbf{S}$ is the source term. The latter includes chemical production which is neglected in this work.

2.2.1. Euler equations

For the Euler equations, $\mathbf{U} = [\rho, \rho u_i, \rho E]^T$ where ρ is the density, u_i is the velocity in component i and E is the total internal energy. Since the equations model inviscid flow, the viscous flux $\mathbf{F}_v$ is null. The inviscid flux is given by

$$\mathbf{F}(\mathbf{U}) = \begin{bmatrix} \rho u_i \\ \rho u_i u_j + \delta_{ij} P \\ \rho u_i H \end{bmatrix}, \tag{2.2}$$

where δ_{ij} is the Kronecker delta, P is the static pressure, and $H = E + P/\rho$ is the total enthalpy. To compute the pressure from the state vector quantities, Equation (2.3) is used,

$$P = (\gamma - 1)\rho \left(E - \frac{1}{2} u_i u_i \right), \tag{2.3}$$

where γ is the ratio of specific heats and is equal to 1.4 for air [5]. Since constant thermodynamic properties are assumed, γ always takes the value of 1.4 in this work.

As mentioned in Section 2.1.1, this work also uses an alternative state vector which uses so-called primitive variables and is given by $\mathbf{W} = [P, u, T]$, where T is the temperature. Note that by using the relation between conservative and primitive variables $\mathbf{U} = \mathbf{U}(\mathbf{W})$, the system of equations can be solved without loss of conservation. For details of how this is done, the reader is referred to Schrooyen [75].

It is worthwhile mentioning that a different set $\tilde{\mathbf{U}} = [\rho, \rho u_i, \rho H]^T$ has been suggested and used in the literature to better conserve total enthalpy for steady inviscid cases when artificial viscosity is added to the equations [4, 15]. This alternative set of variables was not implemented in this work.

2.2.2. Compressible Navier-Stokes equations

For the Navier-Stokes, only the state vector of conservative variables $\mathbf{U} = [\rho, \rho u_i, \rho E]^T$ is used. The inviscid and viscous fluxes are given by

$$\mathbf{F}(\mathbf{U}) = \begin{bmatrix} \rho u_i \\ \rho u_i u_j + \delta_{ij} P \\ \rho u_i H \end{bmatrix}, \quad \mathbf{F}_v(\mathbf{U}, \nabla \mathbf{U}) = \begin{bmatrix} 0 \\ \tau_{ij} \\ u_j \tau_{ij} + \kappa_T \frac{\partial T}{\partial x_i} \end{bmatrix}, \quad \text{with} \quad \tau_{ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \delta_{ij} \lambda \frac{\partial u_k}{\partial x_k}. \tag{2.4}$$

The new parameters in Equation (2.4) are the dynamic viscosity μ, the bulk viscosity coefficient λ given by $\lambda = -2/3\mu$, the thermal conductivity κ_T and the temperature T which is related to known quantities by the equation of state $p = \rho R T$, where R is the specific gas constant [5]. The heat flux q is given by Fourier's law

$$q = \kappa_T \frac{\partial T}{\partial x_i}, \tag{2.5}$$

and is a critical aspect for the design of thermal protection systems. Since constant transport properties are assumed, κ_T takes a constant value.

2.3. The discontinuous Galerkin method

In order for Equation (2.1) to be solved numerically, the physical domain Ω on which the equations are defined is partitioned into a mesh with non-overlapping elements. In the discontinuous Galerkin method, the functional space $\mathcal{V}$ consisting of polynomial functions of order p is constructed on these elements. The polynomial functions are regular within each element but can be discontinuous across elements.

Using a simple 1D scalar version of Equation (2.1) as example, the solution $\tilde{u}$ is thus approximated by a linear combination of shape functions ϕ

$$\tilde{u} \approx u = \sum_i U_i \phi_i, \tag{2.6}$$

where U_i are the degrees of freedom, or solution moments. In Argo, the shape functions are Lagrange interpolants based on equidistant points in a parametric space. A mapping is thus used to translate physical coordinates into corresponding parametric ones.

As in FEM, each term is multiplied by test functions v and integrated over the size of each element, Ω_e, to obtain the weak form of the equation

$$\sum_{\Omega_e} \int_{\Omega_e} v \frac{\partial u}{\partial t} d\Omega_e + \sum_{\Omega_e} \int_{\Omega_e} v \frac{\partial F(u)}{\partial x} d\Omega_e = \sum_{\Omega_e} \int_{\Omega_e} v \frac{\partial}{\partial x} \left(D \frac{\partial u}{\partial x} \right) d\Omega_e \tag{2.7}$$

with

$$D \frac{\partial u}{\partial x} \approx F_v. \tag{2.8}$$

2.3.1. Spatial discretisation of the inviscid term

The second, inviscid, term is discretised by integrating by parts and approximating F with a numerical flux $\hat{F}$ on the boundary $\partial\Omega_e$, which results in [21]:

$$\sum_{\Omega_e} \int_{\Omega_e} v \frac{\partial F(u)}{\partial x} d\Omega_e = -\sum_{\Omega_e} \int_{\Omega_e} F(u) \frac{\partial v}{\partial x} d\Omega_e + \sum_{\partial\Omega_e} \int_{\partial\Omega_e} [v] \hat{F}(u^+, u^-, \mathbf{n}) dS \tag{2.9}$$

with $[\bullet]$ being the jump operator such that

$$[v] = v^- n^- + v^+ n^+, \tag{2.10}$$

with the convention that $+$ is for quantities approaching the interface in the direction of its normal and $-$ is associated with direction opposite to the normal.

It is clear that if the shape functions are piecewise constants, the first term on the right-hand side disappears and the method becomes effectively a first-order FVM. The spatial discretisation of the inviscid flux in DG methods can thus be seen as a high-order adaptation of FVM. Inviscid fluxes $\hat{F}$ are determined by solving a Riemann problem at each element interface [21]. The first exact Riemann solver is attributed to Godunov [29]. Nowadays, the scheme is considered expensive and there are many more efficient solvers that solve the Riemann problem approximately [83]. Two main categories of approximate Riemann solvers exist [49]:

1. *Flux Difference Splitting (FDS).* The flux difference is split in two parts, one associated to the downstream travelling wave and the other with the upstream travelling one, as follows: $\mathbf{F}(\mathbf{U}_R) - \mathbf{F}(\mathbf{U}_L) = (\Delta\mathbf{F})^+(\mathbf{U}_L, \mathbf{U}_R) + (\Delta\mathbf{F})^-(\mathbf{U}_L, \mathbf{U}_R)$. Popular schemes in this category include the solvers of Roe [74], of Osher & Solomon [66], and the Harten-Lax-Van Leer (HLL) splitting [34].

2. *Flux Vector Splitting (FVS).* The flux vector is split in a sum of two vectors, again each associated with the downstream or upstream travelling wave: $\mathbf{F}(\mathbf{U}) = \mathbf{F}^+ + \mathbf{F}^-$. The Steger-Warming splitting [81] and the Van Leer splitting [84] are two well-know schemes.

Hybrid flux splittings have appeared with the intention to combine the benefits of both categories. Specifically, the goal was to take the robustness of the FVS, the ability of selecting only entropy-satisfying approximate solutions of FVS, and the ability of capturing exactly stationary contact discontinuities of FDS [49]. The most popular family of hybrid schemes is the advection upstream splitting method (AUSM), for which more information can be found in [47–50]. Comparisons of different flux schemes can be found in [18, 46, 72]. One important trend to note is that for DG several studies have indicated that the influence of the chosen flux scheme on the solution decreases as the order of the method is increased [14, 31, 69]. Various schemes are implemented in Argo and the effect of the chosen scheme will be tested. Those used in this work are the Lax-Friedrichs, AUSM+up, SLAU, and Roe schemes.

Lax-Friedrichs
A very simple FVS scheme regularly used in this work is the Lax-Friedrichs flux [45], given by

$$\hat{\mathbf{F}} = \frac{\mathbf{F}_L + \mathbf{F}_R}{2} - \frac{1}{2}|\lambda|(\mathbf{U}_R - \mathbf{U}_L) \tag{2.11}$$

where R and L correspond to the right and left states and λ is the maximum eigenvalue given by $\lambda = \bar{\mathbf{u}}\cdot\mathbf{n} \pm c$, in which $\bar{\mathbf{u}}$ is the average velocity at the interface, $\mathbf{n}$ is the normal vector and c is the speed of sound. The Lax-Friedrichs scheme is very simple to implement but is excessively dissipative [83]. This drawback is actually of interest here because it makes the scheme very stable. This is the main reason why much of this work was based on the Lax-Friedrichs Riemann solver. Another drawback of this scheme is that it does not have any mechanism to account for large disparities in convective and acoustic scales [75]. The scheme AUSM+up tackles this issue, making it more accurate at low Mach number.

AUSM+up
The AUSM$^+$up scheme was developed by Liou [48] and is based on a splitting of the pressure and convective terms, given by

$$\hat{\mathbf{F}} = \dot{m}\Phi + \mathbf{P}, \quad \text{with} \quad \dot{m} = \rho u,\ \Phi = (1, u, H)^T \text{ and } \mathbf{P} = (0, P, 0)^T. \tag{2.12}$$

The algorithm starts by defining

$$M_{L,R} = \frac{u_{L,R}}{c_{1/2}} \tag{2.13}$$

where $u_{L,R} = \mathbf{u_{L,R}} \cdot \mathbf{n}$ and

$$c_{1/2} = \min(\tilde{c}_L, \tilde{c}_R), \text{ with } \tilde{c} = \frac{(c^*)^2}{\max(c^*, |u|)}. \tag{2.14}$$

The critical speed of sound c^* is given by

$$c^* = \sqrt{\frac{2(\gamma-1)}{\gamma+1}H}. \tag{2.15}$$

With these, the interface Mach number $M_{1/2}$ is computed by

$$M_{1/2} = \mathscr{M}^+(M_L) + \mathscr{M}^-(M_R) - \frac{\kappa_p}{f_a}\max(0, 1-\sigma\bar{M}^2)\frac{P_R - P_L}{\bar{\rho}c_{1/2}^2} \tag{2.16}$$

which uses split Mach numbers $\mathscr{M}$ being polynomial functions given by

$$\mathscr{M}^{\pm}(M) = \begin{cases} 0.5(M \pm |M|) & \text{if } |M| \geq 1, \\ \pm 0.25(M \pm 1)^2 \pm \beta(M^2-1)^2 & \text{otherwise.} \end{cases} \tag{2.17}$$

The mean Mach $\bar{M}$ and density $\bar{\rho}$ at the interface are

$$\bar{M} = \frac{u_L^2 + u_R^2}{2c_{1/2}^2} \quad \text{and} \quad \bar{\rho} = \frac{\rho_L + \rho_R}{2}. \tag{2.18}$$

The interface pressure is defined by Equation (2.19), for which $\mathscr{P}^+$ and $\mathscr{P}^-$ are computed with Equation (2.20).

$$P_{1/2} = \mathscr{P}^+(M_L)P_L + \mathscr{P}^-(M_R)P_R - 2\kappa_u(\bar{\rho} f_a c_{1/2})(u_R - u_L)\mathscr{P}^+(M_L)\mathscr{P}^-(M_R) \tag{2.19}$$

$$\mathscr{P}^{\pm}(M) = \begin{cases} 0.5(1 \pm \text{sign}(M)) & \text{if } |M| \geq 1, \\ \pm 0.25(M \pm 1)^2(2 \mp M) \pm \alpha M(M^2-1)^2 & \text{otherwise.} \end{cases} \tag{2.20}$$

Defining

$$\dot{m}_{1/2} = c_{1/2} M_{1/2} \begin{cases} \rho_L & \text{if } M_{1/2} > 0, \\ \rho_R & \text{otherwise,} \end{cases} \tag{2.21}$$

the complete flux can be written by

$$\hat{\mathbf{F}} = \dot{m}_{1/2} \begin{cases} \Phi_L + \mathbf{P}_{1/2} & \text{if } \dot{m}_{1/2} > 0, \\ \Phi_R + \mathbf{P}_{1/2} & \text{otherwise.} \end{cases} \tag{2.22}$$

A number of parameters were introduced in this scheme and take the values $\kappa_p = 0.25$, $\kappa_u = 0.75$, $\sigma = 1$, $\beta = 1/8$ and $\alpha = -3/4 + 15/16 f_a^2$ where $f_a = M_0(2 - M_0)$ and $M_0^2 = \min(1, \max(\bar{M}^2, M_{co}^2))$ which includes the cut-off Mach number M_{co} in case the actual Mach number in the domain tends to zero.

SLAU

The Simple Low Dissipation AUSM (SLAU) scheme from Shima & Kitamura [76] performs similarly to AUSM+up but has the benefit of not containing user-defined parameters. Since its implementation is based on similar ideas as the AUSM$^+$up scheme, the reader is referred to [39] for the algorithm.

Roe

The Riemann solver of Roe [74] is one of the most popular, and has been the scheme of choice in state-of-the-art heat flux predictions using AV and DG methods [5, 15, 83]. To explain Roe's method, the notations from [74, 83] are used and the following simplified system is considered:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{U})}{\partial x} = 0. \tag{2.23}$$

The method seeks an exact solution to an approximate problem. The system given in Equation (2.23) is linearised and the Jacobian is replaced by a constant matrix $\tilde{A}$, which results in

$$\frac{\partial \mathbf{U}}{\partial t} + \tilde{A}\frac{\partial \mathbf{U}}{\partial x} = 0. \tag{2.24}$$

To construct $\tilde{A}$, a new parameter vector $\mathbf{Q}$ is designed such that

$$\mathbf{U} = \mathbf{U}(\mathbf{Q}) \text{ and } \mathbf{F} = \mathbf{F}(\mathbf{Q}) \tag{2.25}$$

and for the problem at hand is given by

$$\mathbf{Q} = \rho^{1/2}(1, u, v, w, H)^T. \tag{2.26}$$

Solving

$$\mathbf{U}_L - \mathbf{U}_R = \tilde{B}(\mathbf{Q}_L - \mathbf{Q}_R) \tag{2.27}$$

$$\mathbf{F}_L - \mathbf{F}_R = \tilde{C}(\mathbf{Q}_L - \mathbf{Q}_R) \tag{2.28}$$

for $\tilde{B}$ and $\tilde{C}$, matrix $\tilde{A}$ is obtained through $\tilde{A} = \tilde{C}\tilde{B}^{-1}$. The eigenvalues of $\tilde{A}$ are given by

$$\tilde{\lambda}_1 = \tilde{u} - \tilde{a}, \quad \tilde{\lambda}_2 = \tilde{\lambda}_3 = \tilde{\lambda}_4 = \tilde{u}, \quad \tilde{\lambda}_5 = \tilde{u} + \tilde{a} \tag{2.29}$$

and the right eigenvectors are

$$\mathbf{e}_1 = \begin{pmatrix} 1 \\ \tilde{u} - \tilde{a} \\ \tilde{v} \\ \tilde{w} \\ \tilde{H} - \tilde{u}\tilde{a} \end{pmatrix}, \quad \mathbf{e}_2 = \begin{pmatrix} 0 \\ 0 \\ \tilde{v} \\ 0 \\ \tilde{v}^2 \end{pmatrix}, \quad \mathbf{e}_3 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ \tilde{w} \\ \tilde{w}^2 \end{pmatrix}, \quad \mathbf{e}_4 = \begin{pmatrix} 1 \\ \tilde{u} \\ \tilde{v} \\ \tilde{w} \\ \frac{1}{2}(\tilde{u}^2 + \tilde{v}^2 + \tilde{w}^2) \end{pmatrix}, \quad \mathbf{e}_5 = \begin{pmatrix} 1 \\ \tilde{u} + \tilde{a} \\ \tilde{v} \\ \tilde{w} \\ \tilde{H} + \tilde{u}\tilde{a} \end{pmatrix} \tag{2.30}$$

where

$$\tilde{u} = \frac{\sqrt{\rho_L}u_L + \sqrt{\rho_R}u_R}{\sqrt{\rho_L} + \sqrt{\rho_R}} \tag{2.31}$$

$$\tilde{v} = \frac{\sqrt{\rho_L}v_L + \sqrt{\rho_R}v_R}{\sqrt{\rho_L} + \sqrt{\rho_R}} \tag{2.32}$$

$$\tilde{w} = \frac{\sqrt{\rho_L}w_L + \sqrt{\rho_R}w_R}{\sqrt{\rho_L} + \sqrt{\rho_R}} \tag{2.33}$$

$$\tilde{H} = \frac{\sqrt{\rho_L}H_L + \sqrt{\rho_R}H_R}{\sqrt{\rho_L} + \sqrt{\rho_R}} \tag{2.34}$$

$$\tilde{a} = \left((\gamma - 1)\left(\tilde{H} - \frac{1}{2}(\tilde{u}^2 + \tilde{v}^2 + \tilde{w}^2)\right)\right)^{\frac{1}{2}}. \tag{2.35}$$

Projecting the jump $\mathbf{U}_L - \mathbf{U}_R$ on the right eigenvector leads to

$$\mathbf{U}_L - \mathbf{U}_R = \sum_{i=1}^{5} \tilde{\alpha}_i \mathbf{e}_i \tag{2.36}$$

which can be solved for the wave strengths $\tilde{\alpha}_i$. With the wave strengths, eigenvalues and eigenvectors, the numerical flux at the interface can now be obtained. It is given by any of Equation (2.37) or Equation (2.38):

$$\hat{\mathbf{F}} = \mathbf{F}_L + \sum_{\tilde{\lambda}_i \leq 0} \tilde{\alpha}_i \tilde{\lambda}_i \mathbf{e}_i \tag{2.37}$$

$$\hat{\mathbf{F}} = \mathbf{F}_R - \sum_{\tilde{\lambda}_i \geq 0} \tilde{\alpha}_i \tilde{\lambda}_i \mathbf{e}_i. \tag{2.38}$$

Having now covered the spatial discretisation of the convective term with the Riemann solvers used in this work, the spatial discretisation of the diffusive term is treated next.

2.3.2. Spatial discretisation of the diffusive term

Many techniques exist to discretise the diffusive term in DG methods. Only the Interior Penalty (IP) method will be described because it is the only discretisation available in the code, which was chosen for its compactness and simplicity in implementing the interface term [35]. For a more detailed overview of viscous flux schemes, the reader is referred to [2].

The IP method is based on the boundary penalty approach from Nitsche [63]. Values at the boundary are imposed in a weak manner with the use of a penalty term that minimises the average interpolation error between the value of the solution and the value of the boundary condition. This leads to a lower global error due to smoother variation of the solution [35]. This is illustrated in Figure 2.1, where the green and red lines correspond to weak and strong imposition, respectively.

Following this approach, the last term in Equation (2.7), the diffusive term, is computed as

$$\begin{aligned} \sum_{\Omega_e} \int_{\Omega_e} v \frac{\partial}{\partial x}\left(D\frac{\partial u}{\partial x}\right) d\Omega_e = &\sum_{\Omega_e} \int_{\Omega_e} \frac{\partial v}{\partial x}\left(D\frac{\partial u}{\partial x}\right) d\Omega_e \\ &- \sum_{\partial\Omega_e} \int_{\partial\Omega_e} \left\langle D(u)\frac{\partial u}{\partial x}\right\rangle [v] dS - \theta \sum_{\partial\Omega_e} \int_{\partial\Omega_e} \left\langle D(u)\frac{\partial v}{\partial x}\right\rangle [u] dS + \alpha \sum_{\partial\Omega_e} \int_{\partial\Omega_e} [v][u] dS \end{aligned} \tag{2.39}$$

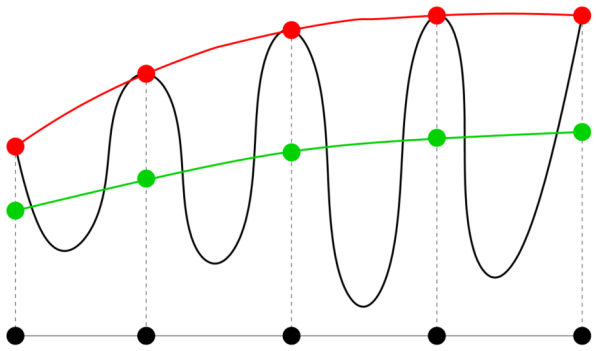


Figure 2.1: Illustration of strong (red) and weak (green) imposition of values on the boundary, from [35, p22].

where $\langle \bullet \rangle$ is the average operator defined as

$$\langle \bullet \rangle = \frac{1}{2}(\bullet^- + \bullet^+). \tag{2.40}$$

On the right hand side, the first two parameters are the result of the discretisation by parts. The third and forth term on the right hand side are the symmetric and penalty terms, respectively, and ensure the method is stable [75]. The penalty parameter α needs to be chosen within a bounded interval, for which the limits can be found in the work of Hilleweart [35]. The parameter θ takes any of the following three options:

1. $\theta = 1$: *symmetric interior penalty (SIPDG) method.* To choose for optimum convergence properties. This is the method used in this work unless mentioned otherwise,
2. $\theta = -1$: *non-symmetric interior penalty (NIPDG) method.* To choose for better stability and α only needs to be strictly positive,
3. $\theta = 0$: *incomplete interior penalty (IIPDG) method.* To choose for simplicity, rarely used in practise.

A key drawback of the Interior Penalty method is that it is inconsistent at $p = 0$. Indeed, all the derivatives on the right-hand side become zero, cancelling all terms except the penalty one. This limitation will prove to make the simulation of certain test cases more challenging. High-order simulations of supersonic and hypersonic flows are typically obtained with order sequencing. This means that a simulation at $p = 1$ takes a converged $p = 0$ solution as initial condition, a simulation at $p = 2$ takes a converged $p = 1$ solution as initial condition, and so on. At $p = 1$, spurious oscillations around shocks can already make simulations difficult. Starting at $p = 0$ is thus highly beneficial as no spurious oscillations appear at that order, allowing to easily obtain a first solution from which higher-order simulations can be done. In this work, it was chosen to accept this limitation rather than implement a different discretisation for the diffusive term. A popular method which is consistent at $p = 0$ is the second scheme of Bassi and Rebay (BR2) [7] and could be implemented and used in future work.

2.3.3. Boundary conditions

The boundary conditions are weakly imposed, meaning that boundaries are treated as an interface between elements at the limit of the domain and ghost elements, on which the desired conditions are specified. For the convective part, the Riemann problem is solved at the interface with the state in ghost elements corresponding to the desired boundary conditions. To impose a Dirichlet boundary condition for the diffusive part, $\frac{\partial u}{\partial x}$ is set to zero thus the only interface terms remaining are the symmetry and penalty terms. The value of u^-, where the minus refers to the ghost value, is specified in both. For a Neumann boundary condition, both states either side of the interface (i.e. the boundary) are set to be equal thus the jump is zero and only the first interface term remains. The value of $D(u)\frac{\partial u}{\partial x}$ is then specified [75]. Table 2.1 summarises the boundary conditions implementation used in this work.

Table 2.1: Boundary conditions implementation in Argo, from [75]

Type	Convective	Diffusive
Adiabatic wall	$\mathbf{u}^-$ specified and for the rest $\mathbf{U}^+ = \mathbf{U}^-$	$F_{v,\text{mass}} \cdot \mathbf{n} = F_{v,\text{energy}} \cdot \mathbf{n} = 0$
Isothermal wall	$\mathbf{u}^-$, T^- specified and for the rest $\mathbf{U}^+ = \mathbf{U}^-$	$F_{v,\text{mass}} \cdot \mathbf{n} = 0$
Freestream	$\mathbf{u}^-$, T^-, P^- specified	$F_v \cdot \mathbf{n} = 0$

2.3.4. Time discretisation

The explicit Runge-Kutta scheme was integrated in DG by Cockburn, Shu, and collaborators, leading to the popular Rung-Kutta Discontinuous Galerkin (RKDG) method for which the development is treated in a series of papers [16–20]. The method has been extensively used and analysed with limiters and reconstruction methods. The main drawback of explicit methods is the constrain on the time-step which can be significant in certain problems. This issue can be overcome by the use of implicit schemes.

The implicit time marching scheme used in this work is the second order multistep Backward Differentiation Formula (BDF2). Since the scheme is not self-starting, the solution after the very first time step, $\mathbf{U}^1$, is computed with a simple BDF1 given by Equation (2.41) with $n = 0$.

$$\mathbf{U}^{n+1} = \mathbf{U}^n + \Delta t \mathscr{R}(\mathbf{U}^{n+1}), \tag{2.41}$$

where

$$\mathscr{R}(\mathbf{U}) = -\nabla \cdot \mathbf{F}(\mathbf{U}) + \nabla \cdot \mathbf{F}_v(\mathbf{U}, \nabla \mathbf{U}) + \mathbf{S}(\mathbf{U}, \nabla \mathbf{U}). \tag{2.42}$$

For $n > 0$, the solutions at the current and previous time steps, $\mathbf{U}^n$ and $\mathbf{U}^{n-1}$, respectively, can be used to compute $\mathbf{U}^{n+1}$, the solution at the next time step, using

$$\mathbf{U}^{n+1} = \frac{4}{3}\mathbf{U}^n - \frac{1}{3}\mathbf{U}^{n-1} - \frac{2}{3}\Delta t \mathscr{R}(\mathbf{U}^{n+1}). \tag{2.43}$$

The next section explains how this scheme is solved.

2.3.5. Solvers for non-linear and linear systems

The resulting non-linear system from Equations (2.41) and (2.43) is solved with a Newton-Raphson (NR) scheme, which works as follows. Writing the non-linear system as

$$H(\mathbf{U}) = 0, \tag{2.44}$$

its linearisation can be obtained by Taylor expansion around $\mathbf{U}^{n+1}$ and neglecting higher order terms, giving

$$H(\mathbf{U}^{n+1}) \approx H(\mathbf{U}^n) + J(\mathbf{U}^{n+1} - \mathbf{U}^n), \tag{2.45}$$

where $J = \partial H / \partial \mathbf{U}$ is the Jacobian evaluated at time n. Using $\Delta \mathbf{U} = \mathbf{U}^{n+1} - \mathbf{U}^n$, this linearised system can now be written in the form

$$J \Delta \mathbf{U} = -H(\mathbf{U}^n), \tag{2.46}$$

and solved for $\Delta \mathbf{U}$. Gaussian elimination is available in Argo to solve the system. However, it is not used in this work because the system is generally large and iterative methods are more efficient. Those used in this work are now briefly explained. The reader is referred to the book from Golub & Van Loan [30] for a more extensive treatment of iterative methods.

Just as Gaussian elimination, iterative methods aim to solve the system

$$A\mathbf{x} = \mathbf{f}, \tag{2.47}$$

where in this case $A = J$, $\mathbf{x} = \Delta\mathbf{U}$, and $\mathbf{f} = -H(\mathbf{U}^n)$. In an iterative method, the system is solved by improving an approximate $\mathbf{x}^k$ at iteration k towards the exact solution $\mathbf{x}^*$. As such, the residual $\mathbf{r}^k$ is then defined as

$$\mathbf{r}^k = \mathbf{f} - A\mathbf{x}^k \tag{2.48}$$

and indicates the quality of the estimate at iteration k. Basic iterative methods split matrix A into $A = M - N$. The system can then be written as $M\mathbf{x} = N\mathbf{x} + \mathbf{f}$. This splitting allows the construction of an iterative scheme given by

$$\begin{aligned}
\mathbf{x}^{k+1} &= M^{-1}N\mathbf{x}^k + M^{-1}\mathbf{f} && (2.49)\\
&= M^{-1}(M - A)\mathbf{x}^k + M^{-1}\mathbf{f} && (2.50)\\
&= \mathbf{x}^k + M^{-1}(\mathbf{f} - A\mathbf{x}^k) && (2.51)\\
&= \mathbf{x}^k + M^{-1}\mathbf{r}^k. && (2.52)
\end{aligned}$$

It is clear that M should be chosen such that it can easily be inverted. In the Jacobi method used in this work, M is simply the diagonal of A. This leads to very fast iterations and the parallelisation is straightforward. However, the lack of off-diagonal elements can make the method slow to converge.

The other iterative method used in this work is called Generalised Minimum Residual (GMRES), and belongs to a different category under the name of Krylov Subspace methods. GMRES is based on the Krylov subspace $\mathcal{K}(A, \mathbf{f}_0, k) = \text{span}\{\mathbf{f}_0, A\mathbf{f}_0, ..., A^{k-1}\mathbf{f}_0\}$. At the kth iteration, $||\mathbf{f} - A\mathbf{x}_k||_2$ is minimised with $\mathbf{x}_k$ belonging to the space $\mathbf{x}_0 + \mathcal{K}(A, \mathbf{f}_0, k)$ [30]. A drawback of GMRES is that the algorithm becomes more computationally and memory intensive at each iteration k. Argo has a restart option to deal with this issue, however the need to use it did not appear in this work. The convergence threshold for the GMRES algorithm was set at 10^{-4} and was generally reached in less than 100 iterations, at which the algorithm would be forced to stop. GMRES can also be made more efficient with the use of preconditioners. The basic idea of preconditioning is that matrix A is multiplied by another matrix P such that their product is as close as possible from the identity matrix. If matrix A is multiplied by P from the right, the system becomes:

$$\begin{aligned}
AP\mathbf{y} &= \mathbf{f} && (2.53)\\
\mathbf{x} &= P\mathbf{y}. && (2.54)
\end{aligned}$$

Two options for P are used in this work. Bloc Jacobi preconditioners use the diagonal blocs of A. The advantages and drawbacks are as for the basic iterative method. The second option is Bloc Incomplete LU, which is based on an incomplete LU factorisation of the blocs. The LU factorisation consists of decomposing A into a lower (L) and upper (U) triangular matrices. This results in dense L and U even if A is sparse. In Argo, the incomplete factorisation allows to control the levels of fill-in, with level 0 meaning that no additional entries occur. In theory, a higher level will result in increased memory and computational cost but in better convergence. In practise, this work found that levels 0 and 1 were the most efficient for the problems considered.

2.4. Artificial viscosity

Before getting into the artificial viscosity method itself, shock detectors (also called sensors or indicators), need to be treated. Shock detectors are part of most shock capturing techniques and have for purpose to identify the location of discontinuities in the computational domain. They work by computing a quantity which measures the smoothness of the solution in each element of the mesh. If this quantity is larger than a threshold on a given element, that element is flagged as "troubled" and artificial viscosity (or another shock capturing technique) will be applied on it. Many sensors have been proposed in the literature, and many have been found to work reliably. The indicators of Krivodonova et al. [43], Harten [32], and Persson & Peraire [69] are among popular options. The main weakness of sensors is that the threshold is user-defined. In practice,

its value needs to be defined by trial and error and will be case-dependent. A few solutions to this issue have been proposed, the most generic being the one from Vuik & Ryan [86] at the moment.

The artificial viscosity technique used in this work is largely based on the method proposed by Persson & Peraire [69] and the improvement proposed by Barter & Darmofal [5]. Other options are available but those chosen for this work are known to work well and have been considered as the most robust by Burgess and Mavripilis [13], among the methods they considered. The detector by Persson and Peraire works on the principle that expansions of different order will be similar when the solution is smooth but different when it contains discontinuities. In each element, a component of the solution of order p can be written as:

$$u = \sum_{i=1}^{N(p)} U_i \phi_i \tag{2.55}$$

where $N(p)$ corresponds to the expansion's number of terms. The solution can also be expanded in the same way but with terms up to order $p-1$ only:

$$\hat{u} = \sum_{i=1}^{N(p-1)} U_i \phi_i \tag{2.56}$$

The shock sensor is based on these two expansions. In regions where the solution is smooth, the decay of the expansion coefficients will be quick. However, discontinuities will lower the decay rate. Therefore, discontinuities can be detected by comparing the two expansions of different order. The sensor S_e is given by:

$$S_e = \frac{(u-\hat{u}, u-\hat{u})_e}{(u,u)_e} \tag{2.57}$$

where $(\cdot,\cdot)_e$ is the inner product in $L_2(\Omega_e)$ [69]. A threshold needs to be specified in order to define above which value of S_e is artificial viscosity (or another stabilisation technique) applied. Overall, the sensor has been found to be reliable by Persson and Peraire as well as Lv et al. [57], and has been used by others in their work [5, 40]. Worth noting is that the detector does not work for simulations at $p = 0$ but does not need to since spurious oscillations do not occur at that order.

Artificial viscosity is then added to the equations in the form of a Laplacian term:

$$\nabla \cdot (\varepsilon \nabla \mathbf{U}). \tag{2.58}$$

The amount of artificial viscosity added ε is scaled by a user-defined magnitude k, the element size h, the polynomial order p and by the maximum wave speed λ_{max}, such that

$$\varepsilon_0 = k \frac{h}{p} \lambda_{max}. \tag{2.59}$$

This value is then inserted into Equation (2.60) to obtain a smoother variation depending on the sensor values.

$$\varepsilon_e = \begin{cases} 0 & \text{if } s_e < s_0 - \kappa \\ \frac{\varepsilon_0}{2}\left(1 + \sin\frac{\pi(s_e - s_0)}{2\kappa}\right) & \text{if } s_0 - \kappa < s_e < s_0 + \kappa \\ \varepsilon_0 & \text{if } s_e > s_0 + \kappa \end{cases} \tag{2.60}$$

where $s_e = \log_{10} S_e$, s_0 is the sensor threshold and theoretically scales with $1/p^4$, and κ is the interval on which the artificial viscosity increases smoothly from zero to its maximum value. The method as such results in a discrete artificial viscosity field, in which each mesh element e has a specific amount of AV ε_e attributed to it. Barter & Darmofal [5] found that this could lead to oscillations and proposed to smooth the AV field with an additional PDE. Smoothing the AV field has now become a standard procedure in AV methods. In Argo, a simpler implementation than that of Barter and Darmofal is used. At each interface, the maximum ε_e value is chosen to construct a linear polynomial interpolation for ε, which leads to a smooth AV field.

Also common in current artificial viscosity methods in DG is the h/p scaling introduced by Persson & Peraire [69]. By scaling the artificial viscosity with the length resolution of a polynomial, subcell shock capturing can be achieved for sufficiently high p. Just as for the detector, AV methods have parameters that need to be defined by the user. In this case, the user needs to specify three parameters before each simulation,

namely the threshold s_0, the interval κ, and the magnitude k. In practise, a few time steps or iterations are run, at which point the user can already obtain an idea of what values would be appropriate for the parameters. It is worth mentioning explicitly that the cutting point above which elements are flagged as troubled is not at the threshold but at $s_0 - \kappa$. Therefore, there are two parameters which define together the point above which artificial viscosity is applied. In addition, the user usually has the choice between adding more artificial viscosity on a smaller region around the discontinuity, or adding less on a larger area. The three parameters are thus very much interlinked.

3

Application to an inviscid and steady supersonic wedge test case

The artificial viscosity method described in the previous chapter is applied to a steady wedge test case solved with the Euler equations. This problem is chosen because it contains a simple oblique shock and has an analytical solution. The latter allows to verify that the method performs as expected, i.e. that it removes oscillations locally without affecting accuracy elsewhere. A parametric study is done to observe the effect of each AV parameter at a low supersonic Mach number. The Mach number is then increased to 4 at which point challenges start to arise.

3.1. Problem set-up

As its name implies, the wedge test case simulates a supersonic flow over a wedge. An oblique shock appears in order to turn the flow parallel to the ramp. The wedge forms an angle of about 11.5°, and the full problem geometry is given in Figure 3.1. Freestream conditions are imposed on the left and top boundaries, the flow is forced to be tangent to the bottom wall and nothing needs to be imposed on the right boundary since the flow is supersonic.

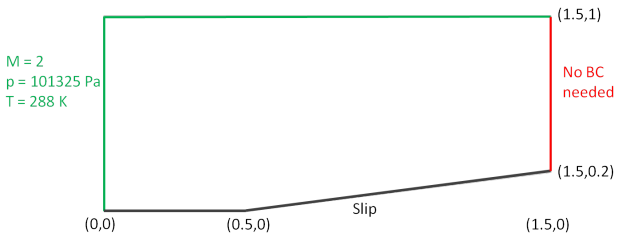


Figure 3.1: Wedge test case set-up. Coordinates are in brackets (figure not to scale).

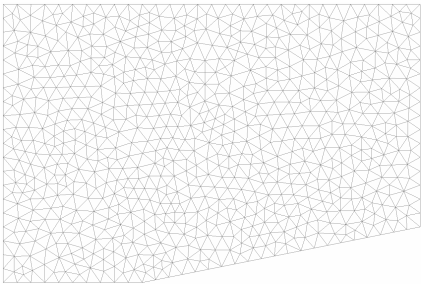


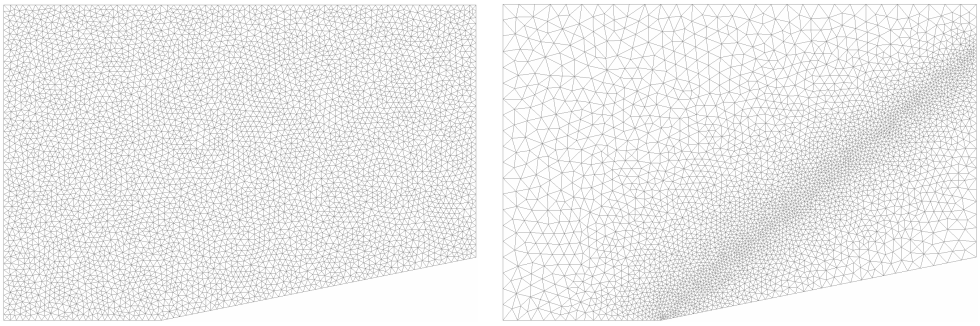
Figure 3.2: Baseline mesh with 1467 elements for the wedge test case

Using the oblique shock relations (see [37, p608] for their derivation), the post-shock conditions can be computed analytically. The results are given in Table 3.1, and the shock angle with respect to the horizontal is 40.8°.

Table 3.1: Pre- and post-shock conditions for the wedge test case at Mach 2

Parameter	Pre-shock	Post-shock
Mach number [-]	2	1.5915
Static pressure [Pa]	101325	184798
Static temperature [K]	288	344.1

The problem is solved with an implicit BDF2 scheme, and a GMRES algorithm with Jacobi preconditioning is used to solve the linear system. Shock detection uses the density as variable. The main mesh is relatively coarse at 1467 triangular elements, as shown in Figure 3.2. Two other meshes with different refinements are also used for comparison. One is the fine mesh with about 9000 elements shown in Figure 3.3(a) and the other has a local refinement around the shock, allowing smaller elements in the critical region while keeping the overall number of elements at about 7000. The latter mesh is shown in Figure 3.3(b). All $p = 1$ solutions in this chapter were obtained from a converged $p = 0$ solution as initial condition.



(a) Isotropic mesh with 9283 elements (b) Locally refined mesh with 7252 elements

Figure 3.3: Fine meshes for the wedge test case

3.2. Parametric study at Mach 2

The magnitude k, threshold s_0, and interval κ are modified one by one to test the sensitivity on the solution. The baseline set-up uses the Lax-Friedrichs scheme for the flux, a polynomial order of 1, no smoothing of the AV field, and the coarse mesh. The Lax-Friedrichs scheme is chosen for its stability, and smoothing is not applied to visualise more clearly the effect of changing AV parameters. Primitive variables are used for all simulations in this chapter. It was decided to clip all results slightly above and below the theoretical values given in Table 3.1 in order to highlight oscillations, which appear as white areas in the images. The first results to be shown were computed with no artificial viscosity to highlight the issue of spurious oscillations and the need to remove them.

No artificial viscosity

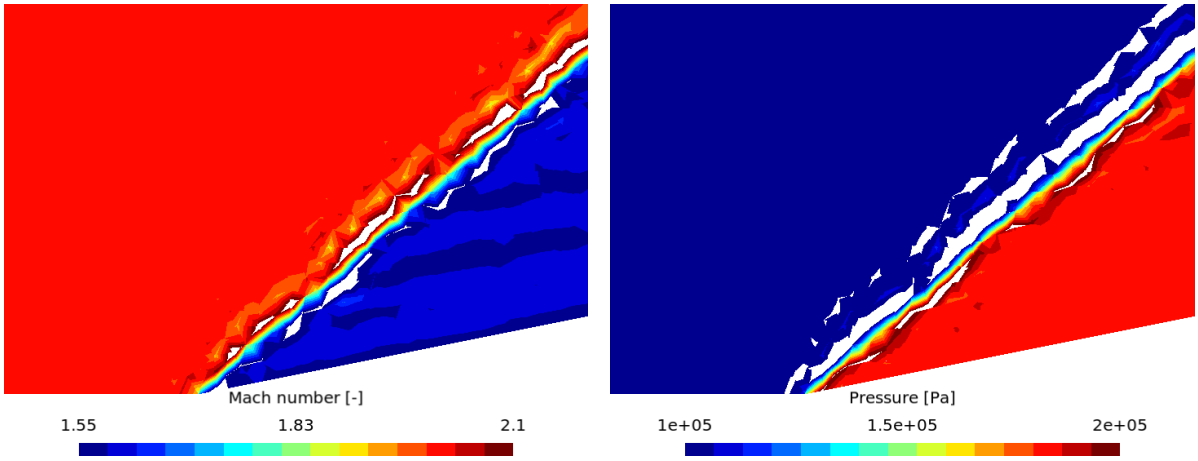
Results for the Mach number and pressure are shown in Figures 3.4(a) and 3.4(b) respectively, when no artificial viscosity is added to the equations. As expected, solving the equations without artificial viscosity results in spurious oscillations before and after the shock. They are stronger for the pressure than for the Mach number. They also affect a wider region in the case of the pressure. Furthermore, these oscillations vary along the shock length, as does the shock thickness.

Baseline parameters: $k = 0.05, \quad s_0 = -2.0, \quad \kappa = 1.0$

Figures 3.5(a) and 3.5(b) show the artificial viscosity field (i.e. the magnitude ϵ across the domain) and the sensor values s_e on each element, respectively. It is clear that AV is added only on a thin region of approximately one element, which is generally a desired feature as long as oscillations are effectively removed. However, the amount of AV added varies along the length of the shock. This is not surprising considering it was just seen that oscillations vary along the length of the shock when no artificial viscosity was added. Also, it can be clearly seen how each mesh element has a specific artificial viscosity value since no smoothing is applied, resulting in a discrete AV field.

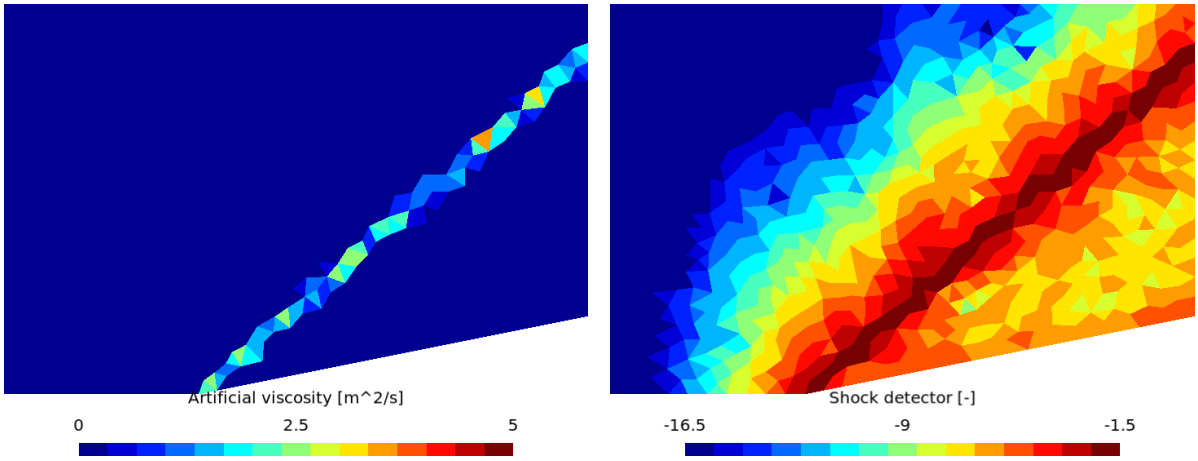
For a perfectly smooth solution, the sensor returns a value of -16. In this case, the sensor value is close to -1.5 at the shock location, but it is clear from Figure 3.5(b) that there is a large transition region between the two extremes. Also, the flow after the shock does not recover the pre-shock smoothness.

A clear improvement can be seen for the Mach number field in Figure 3.5(c) with respect to the results with no artificial viscosity, but the pressure field in Figure 3.5(d) suggests that a better set of parameters can be found. Furthermore, it can be seen that, as expected, the shock is not homogeneous along its length, with varying thickness. The regions just before and after the shock further show the non-homogeneity of the solution in the direction parallel to the shock.

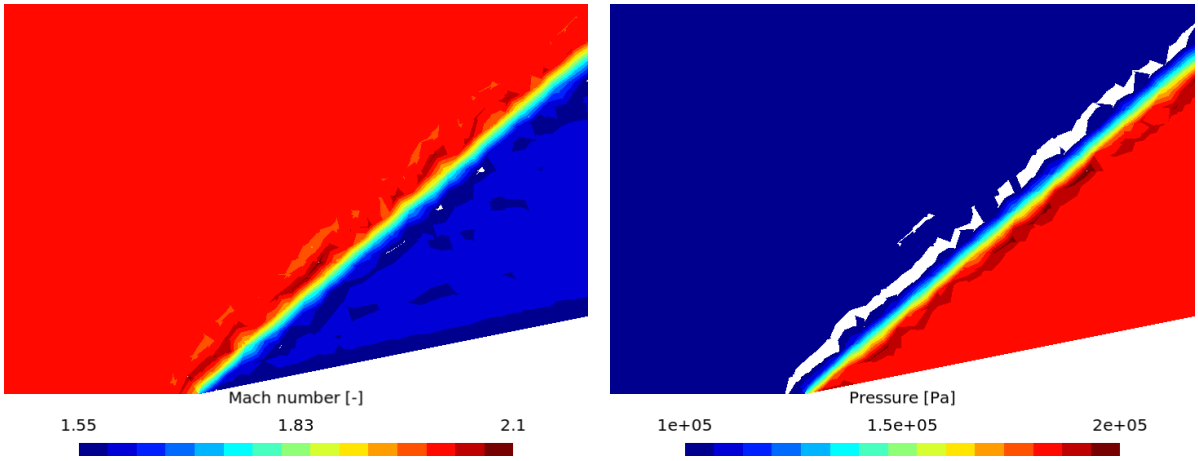


(a) Mach number (b) Pressure

Figure 3.4: Solution to the wedge test case at $p = 1$ with Lax-Friedrichs and without artificial viscosity



(a) Artificial viscosity (b) Shock detector s_e



(c) Mach number (d) Pressure

Figure 3.5: Solution to the wedge test case at $p = 1$ with Lax-Friedrichs and baseline artificial viscosity parameters $k = 0.05$, $s_0 = -2.0$, and $\kappa = 1.0$

If white areas in the results show the error caused by spurious oscillations, they do not show the magnitude of the oscillations. For that reason, Figure 3.6 shows a 3D plot of the Mach number distribution where the vertical axis corresponds to the Mach number values. One can clearly see how the spurious oscillations vary in strength along the length of the shock. For the pressure, drawing the 3D plot to scale resulted in poor visualisation. It was thus preferred to show a cut of the pressure field at $y = 0.3$, in Figure 3.7. The overshoot and undershoot are clearly visible.

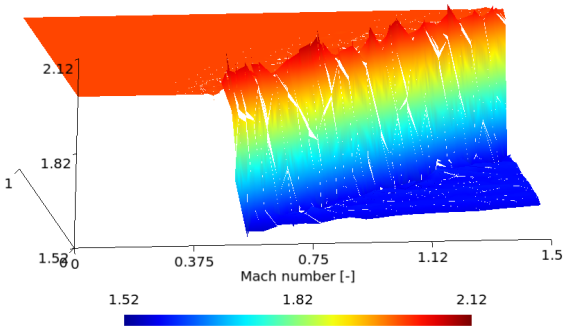


Figure 3.6: 3D view of the Mach number distribution, vertical axis corresponds to Mach number values

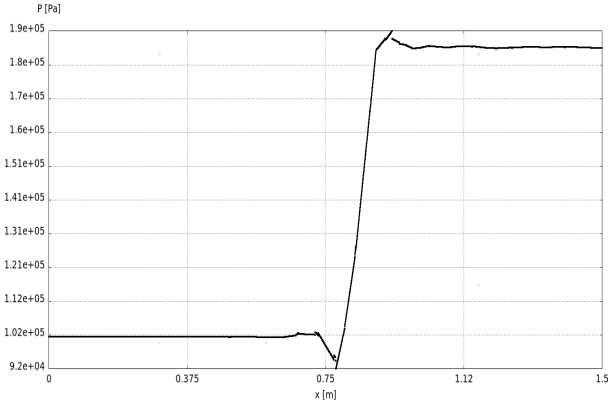


Figure 3.7: Cut of the pressure field from at $y = 0.3$

Overall, spurious oscillations remain despite the use of artificial viscosity. Partly, this is due to suboptimal choices for the parameters k, s_0, and κ. The next sections vary each of the three parameters to understand their impact on the solution.

Changing the interval κ

By changing the interval κ, the size of the region on which AV is applied increases or decreases. The artificial viscosity profile as given by Equation (2.60) is shown in Figure 3.8 for the different interval values. The larger value of $\kappa = 2$ will result in AV being applied to mesh elements having a relatively low sensor value (down to -4) but also implies that the maximum amount of AV $k = 0.05$ will only be applied on elements receiving a sensor value above 0. Figure 3.9(a) shows that the maximum value is -1.5, thus the largest AV magnitude will clearly never be applied. In the case of the smallest interval $\kappa = 0.5$, $s_e > -2.5$ is needed for AV to be applied on that element, but the maximum AV magnitude is applied already for sensor values slightly above -2. For $s_e = -1.5$ as example, it is clear in Figure 3.8 that more artificial viscosity is applied in the $\kappa = 0.5$ case. A quick look at Figures 3.10(a) and 3.10(b) confirms that the largest magnitude is indeed applied for the low $\kappa = 0.5$ value. In short, the largest interval adds less artificial viscosity but on more elements, whereas the shortest interval adds more artificial viscosity on less elements.

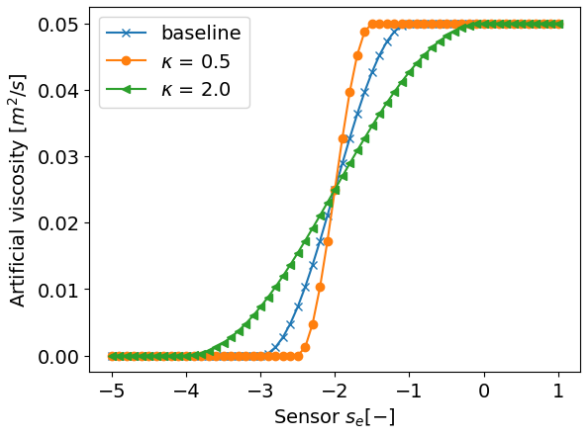
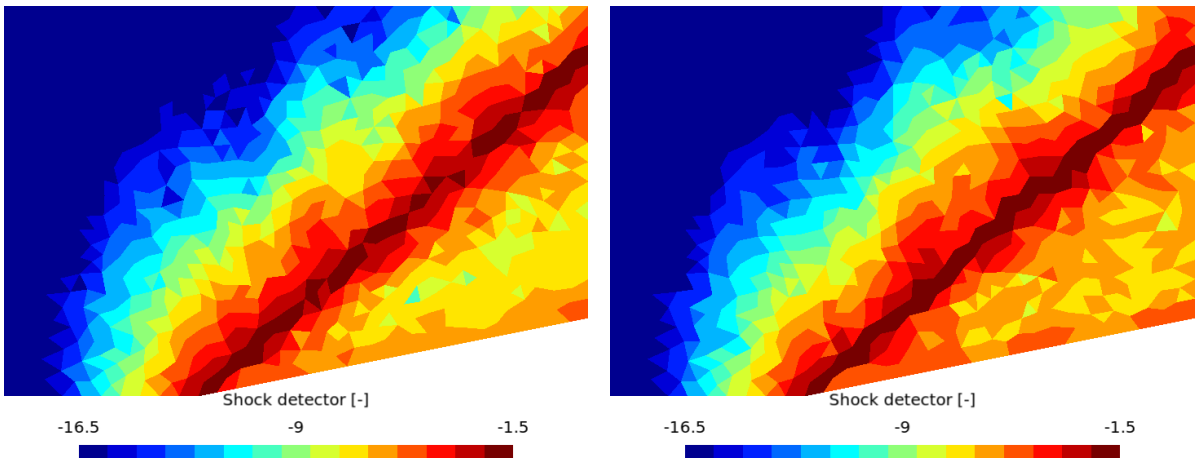


Figure 3.8: Artificial viscosity profile assuming $h = p = \lambda_{max} = 1$ for different interval κ values

Figures 3.9(a) and 3.9(b) show the sensor distribution for both intervals. One can see how, due to the differences in how the artificial viscosity is added, the smoothness of the solution is different and thus the sensor fields are not identical.

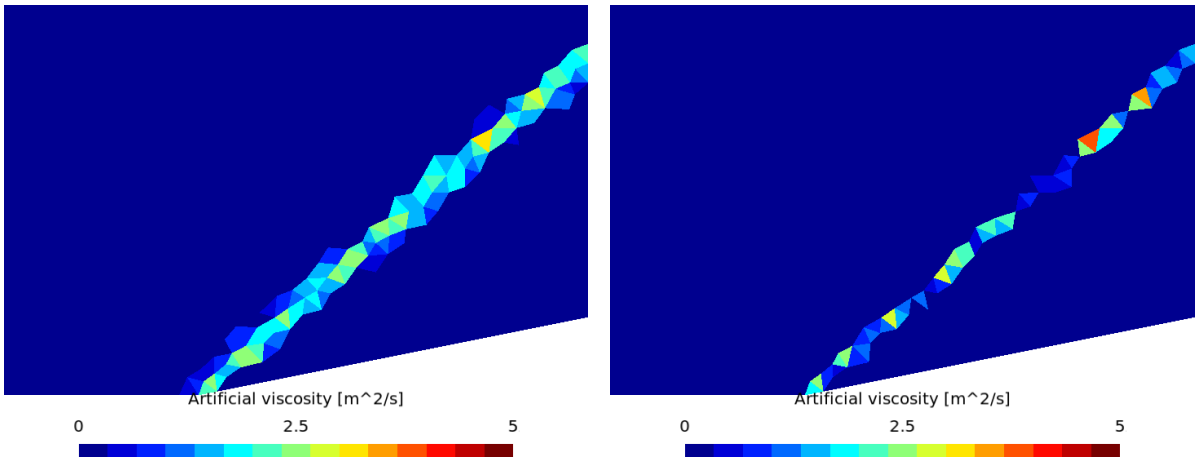


(a) Shock detector s_e for $\kappa = 2$

(b) Shock detector s_e for $\kappa = 0.5$

Figure 3.9: Shock detector for changing interval κ at $p = 1$ with Lax-Friedrichs, $k = 0.05$, $s_0 = -2.0$

Plots of the artificial viscosity fields for both intervals in Figures 3.10(a) and 3.10(b) show clearly what had already been discussed above. The largest interval sees lower magnitudes on more elements, whereas for the smallest interval the opposite is true.

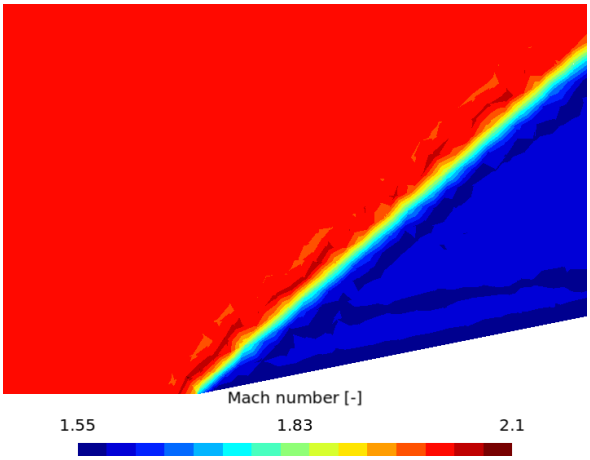


(a) Artificial viscosity for $\kappa = 2$

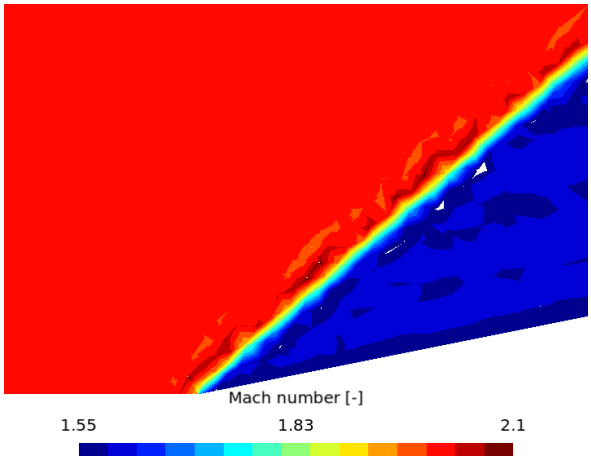
(b) Artificial viscosity for $\kappa = 0.5$

Figure 3.10: Artificial viscosity for changing interval κ at $p = 1$ with Lax-Friedrichs, $k = 0.05$, and $s_0 = -2.0$

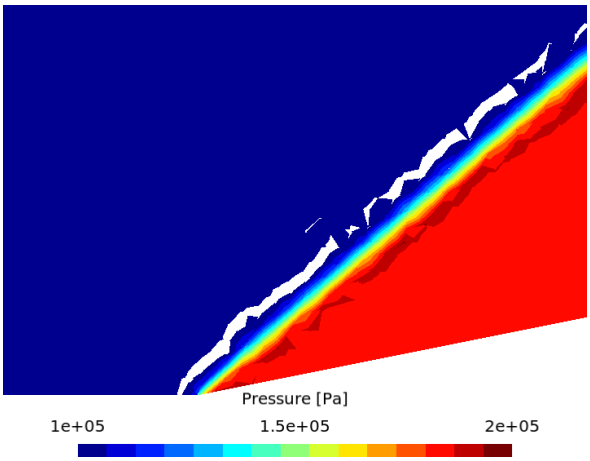
The results for the Mach number (Figures 3.11(a) and 3.11(b)) and the pressure (Figures 3.11(c) and 3.11(d)) show that an improvement with respect to the baseline is achieved for $\kappa = 2$, but that $\kappa = 0.5$ deteriorates the results. In this case, dissipating on a larger area thus leads to better results, even though the maximum amount of artificial viscosity is slightly lower. It is worth noting that if the artificial viscosity region becomes too thin, the shock will not be stabilised independently of how much artificial viscosity is added. Indeed, the shock needs to be spread by the artificial viscosity over a sufficiently large region in order to be resolved by the DG method.



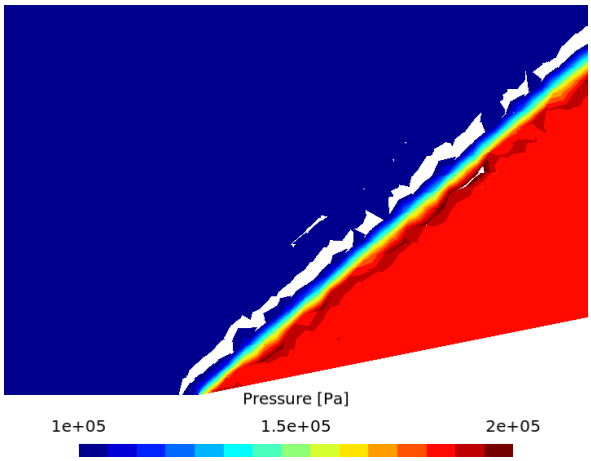
(a) Mach number for $\kappa = 2$



(b) Mach number for $\kappa = 0.5$



(c) Pressure for $\kappa = 2$



(d) Pressure for $\kappa = 0.5$

Figure 3.11: Solution to the wedge test case for changing interval κ at $p = 1$ with Lax-Friedrichs, $k = 0.05$, and $s_0 = -2.0$

Changing the threshold s_0

Only the effect of a lower threshold will be investigated here because increasing its value would clearly be the wrong choice. The combination of the baseline threshold and interval is already somewhat inappropriate since it leads to the maximum AV amount only being applied to elements having a sensor value of at least -1. However, the largest sensor value is -1.5. Therefore, the maximum AV amount is never added. Increasing the threshold would only make matters worse. On the other hand, lowering the threshold will result in a larger region where AV is applied. The new AV profile for a different threshold is shown in Figure 3.12. Figures 3.13(a) and 3.13(b) show what the AV magnitude and sensor fields look for a threshold $s_0 = -3$. As opposed to the baseline case where the largest AV magnitude is never applied, in this case it is for sensor values approximately larger than -2. Therefore, a number of elements will get the maximum AV magnitude, as can be seen in Figure 3.13(a). Note that white elements correspond to values higher than 5, which have been clipped. The larger region where AV is applied results in the oscillations being the most dissipated of all cases seen so far, which then translates into lower sensor values. This is clear in Figure 3.13(b).

Since the region where AV is applied is large, it is not a surprise to see in Figures 3.13(c) and 3.13(d) that the shock thickness is larger as well. Some oscillations remain visible in the pressure field and to a lesser extent in the Mach number field as well. The AV acting on a larger area, oscillations have been better suppressed. However, this has come at the cost of obtaining a wider shock. In many cases, a wider shock will not be an issue if it means the pre- and post-shock regions are more accurate. Nevertheless, it will become necessary to trade-off shock sharpness with solution smoothness if a sharper shock is desired.

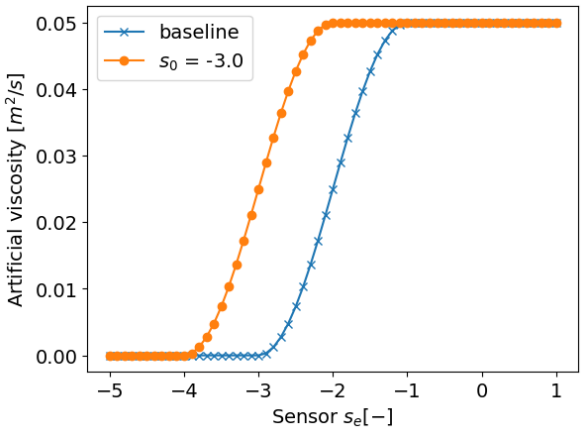
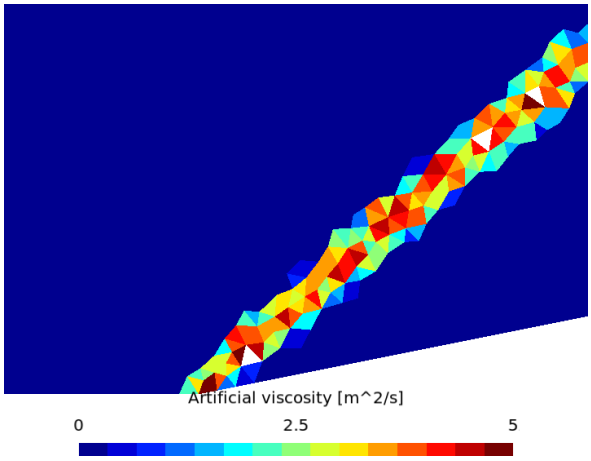
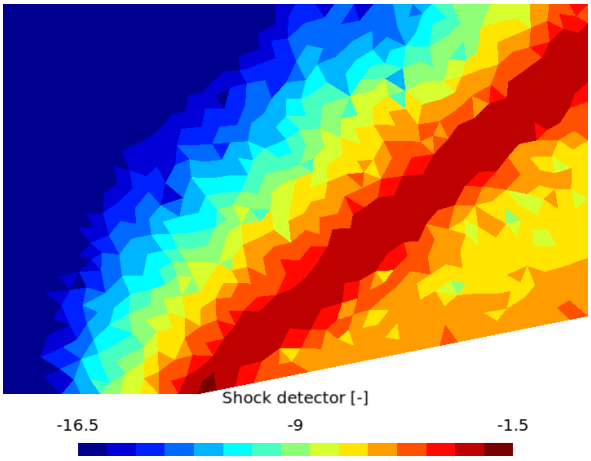


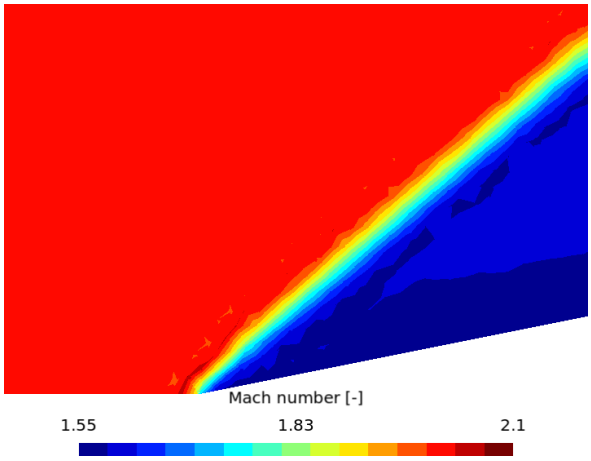
Figure 3.12: Artificial viscosity profile assuming $h = p = \lambda_{max} = 1$ for different threshold s_0 value



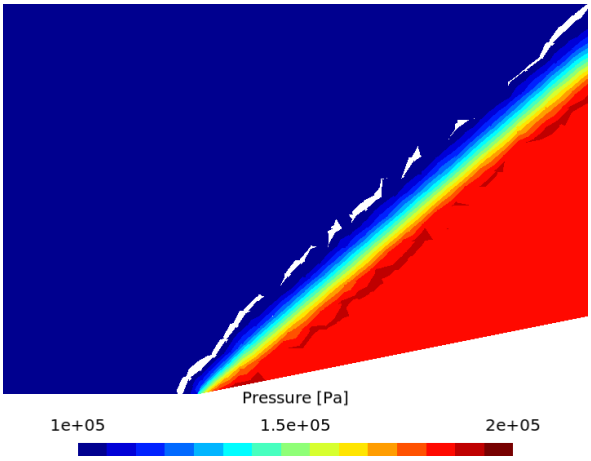
(a) Artificial viscosity for $s_0 = -3$



(b) Shock detector s_e for $s_0 = -3$



(c) Mach number for $s_0 = -3$



(d) Pressure for $s_0 = -3$

Figure 3.13: Artificial viscosity and shock detector for changing threshold s_0 at $p = 1$ with Lax-Friedrichs, $k = 0.05$, and $\kappa = 1.0$

Changing the magnitude k

Increasing the AV magnitude to $k = 0.1$ (Figure 3.15(c)) or decreasing it to $k = 0.01$ (Figure 3.15(d)), as shown in Figure 3.14, affects the solution as expected.

For the larger magnitude, Figure 3.15(a) suggests that less spurious oscillations will appear since the shock

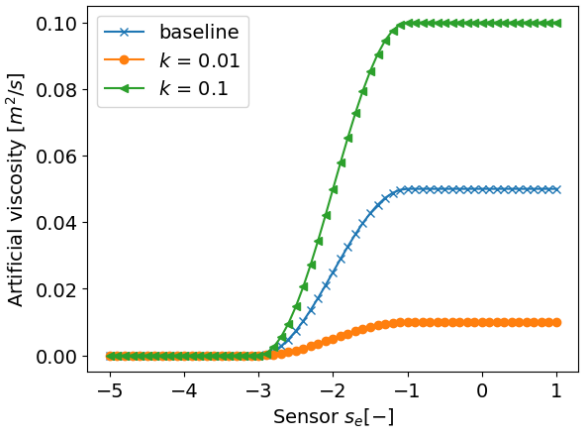
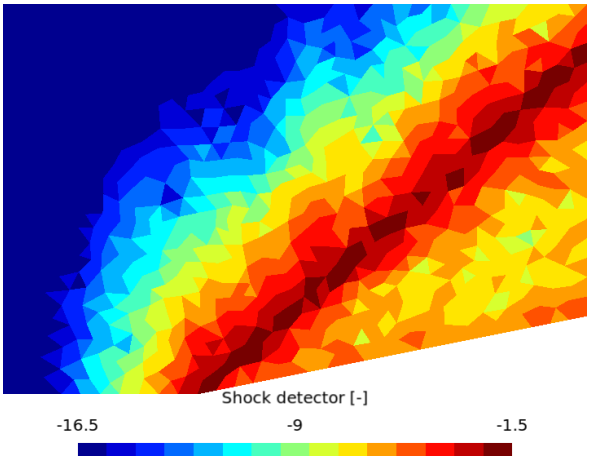
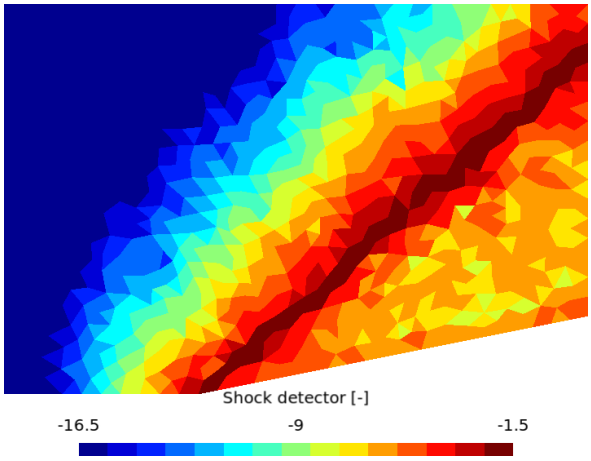


Figure 3.14: Artificial viscosity profile assuming $h = p = \lambda_{max} = 1$ for different magnitudes k

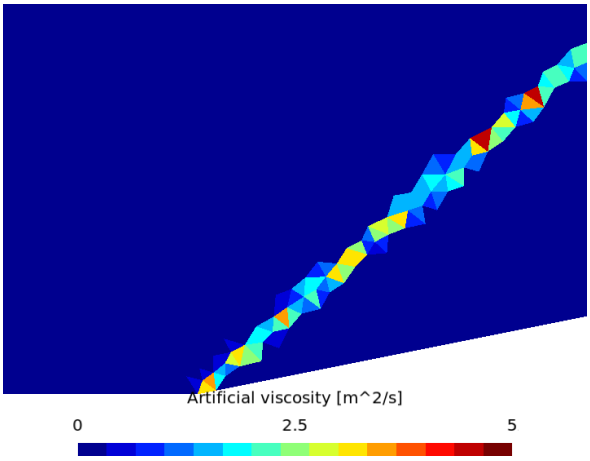
sensor returns lower values than in the baseline case. The opposite is true for the lower magnitude, with Figure 3.15(b) showing larger sensor values.



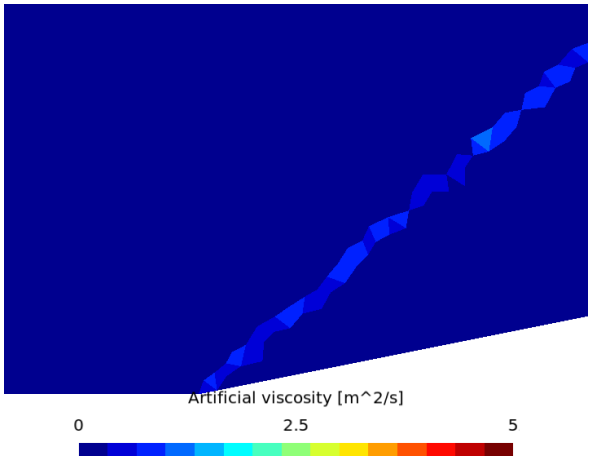
(a) Shock detector s_e for $k = 0.1$



(b) Shock detector s_e for $k = 0.01$



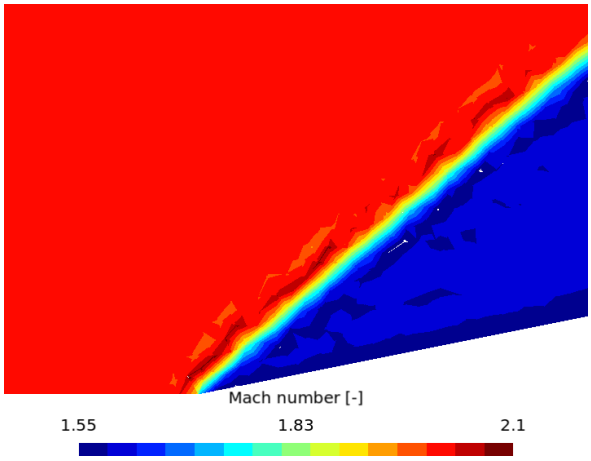
(c) Artificial viscosity for $k = 0.1$



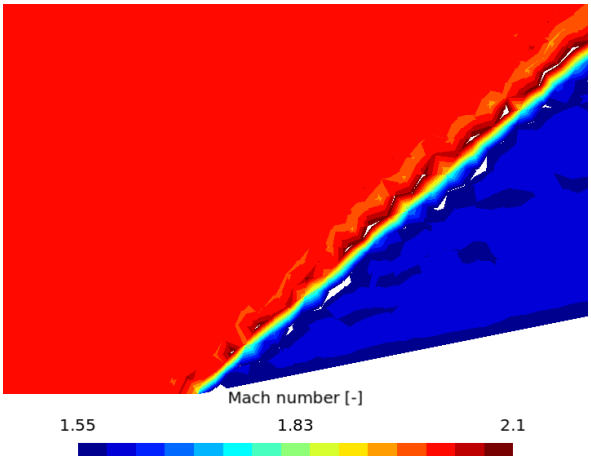
(d) Artificial viscosity for $k = 0.01$

Figure 3.15: Artificial viscosity and shock detector for changing magnitude k at $p = 1$ with Lax-Friedrichs, $\kappa = 1.0$, and $s_0 = -2.0$

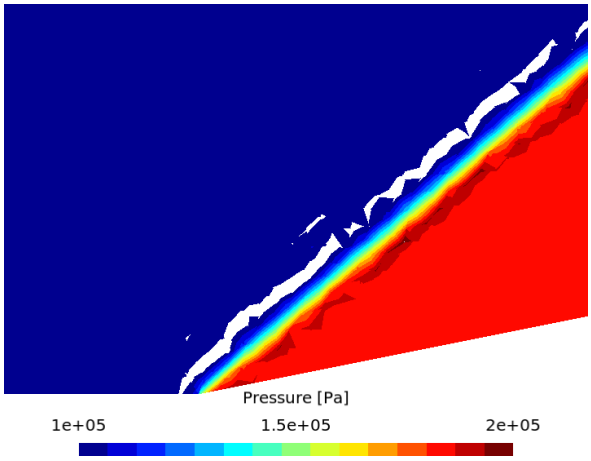
Figures 3.16(a) and 3.16(b) show the Mach number field and confirm what the sensor values suggested. Indeed, the larger k value leads to less spurious oscillations whereas the lower k value clearly results in insufficient artificial viscosity. Although not as clear, the same observations can be made for the pressure distribution, shown in Figures 3.16(c) and 3.16(d). As already seen before, the pressure field contains clipped values even when the Mach number does not.



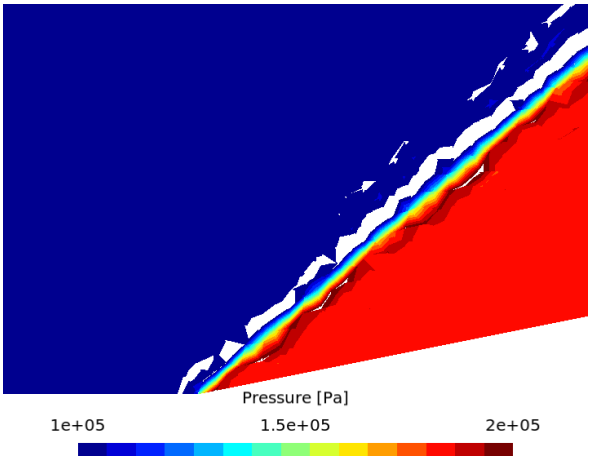
(a) Mach number for $k = 0.1$



(b) Mach number for $k = 0.01$



(c) Pressure for $k = 0.1$



(d) Pressure for $k = 0.01$

Figure 3.16: Solution to the wedge test case for changing magnitude k at $p = 1$ with Lax-Friedrichs, $\kappa = 1.0$, and $s_0 = -2.0$

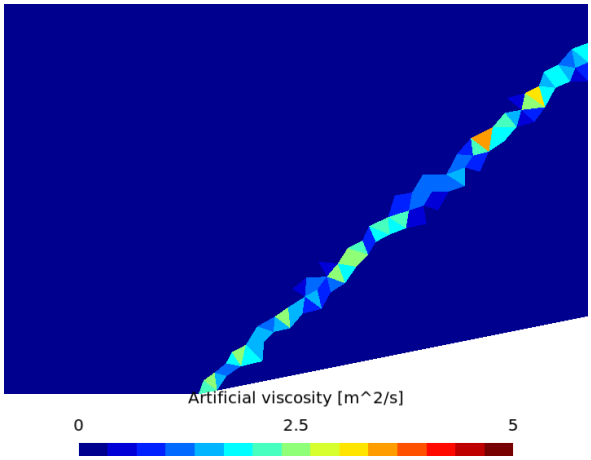
A feature clearly visible in both the Mach and pressure distributions is that the shock thickness appears to be larger for the highest k. This makes sense because although artificial viscosity is added on a region with thickness of one element in both cases, the larger magnitude will dissipate the shock more.

Increasing the magnitude further does not necessarily keep improving the results. In fact, using $k = 0.4$ gives very similar results. Increasing further the value would eventually lead to numerical stability issues. At this point, it starts becoming apparent that the difficulty with artificial viscosity methods is to find the combination of parameters that results in a sharp shock and removes spurious oscillations. This task is made more challenging by the fact that there can be a trade-off to make between the two. Barter & Darmofal use $s_0 = -(4 + 4.25\log_{10}(p))$ and $\kappa = 0.5$ in [5]. In this section, $p = 1$ thus the parameters would take the values $s0 = -4$ and $\kappa = 0.5$. For the problem at hand, this would result in a very wide shock, with the magnitude of the artificial viscosity changing rapidly from zero to its maximum.

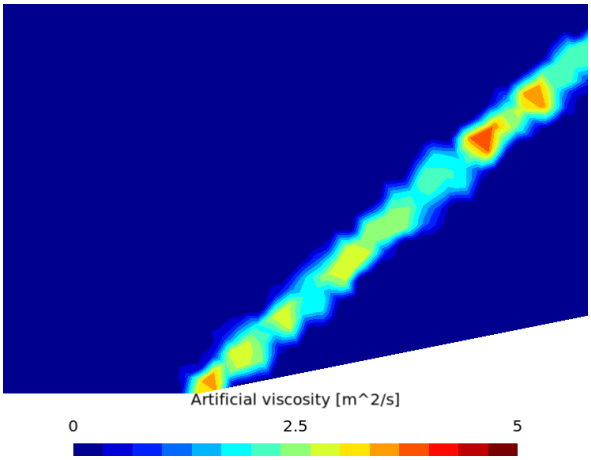
Using a continuous artificial viscosity field

The artificial viscosity was thus far applied in a discrete way to visualise the impact of each AV parameter individually. A specific value was attributed to each mesh element, defined by the detector output. Since smoothing the AV field has become a common procedure in artificial viscosity methods, the effect is investigated here. Again, $p = 1$ is used and $k = 0.05$, $s_0 = -2.0$, and $\kappa = 1.0$. Figures 3.17(a) and 3.17(b) show the discrete and continuous AV fields, respectively. Clearly, smoothing the AV field increases its footprint.

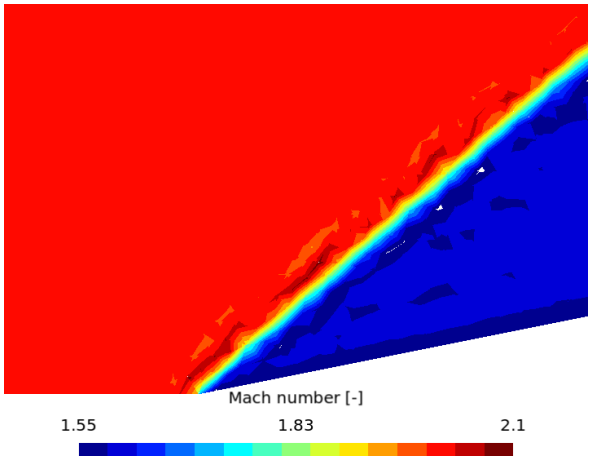
Looking at Figures 3.17(c) and 3.17(d), showing the Mach number distribution for discrete and smooth AV fields, one can see that smoothing the AV field has reduced spurious oscillations. The pressure distributions in Figures 3.17(e) and 3.17(f) confirm this observation. However, it is difficult to tell whether the improved results are due to the smoothing or due to a wider AV footprint. Nevertheless, the need for smoothing the AV field becomes clear when using larger k values, and was done for all results shown in the following sections.



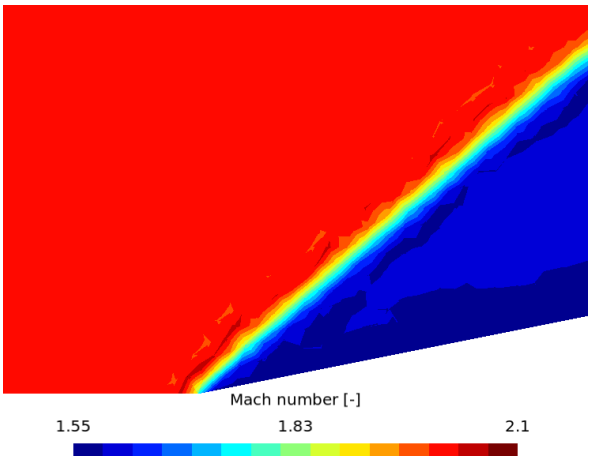
(a) Mach number for discrete AV field



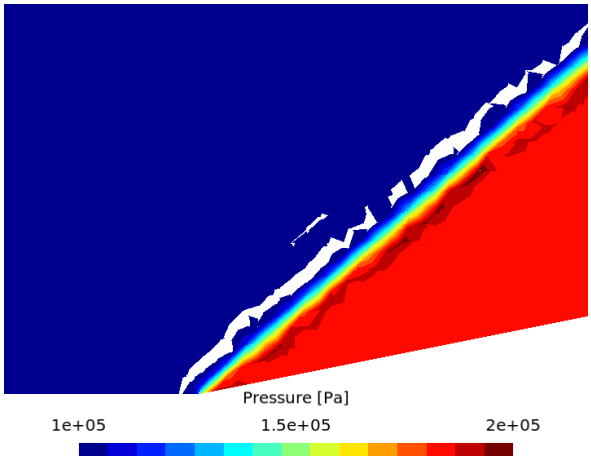
(b) Mach number for smooth AV field



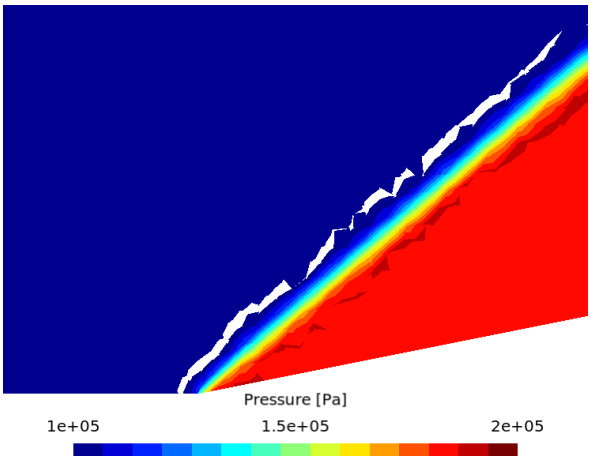
(c) Mach number for discrete AV field



(d) Mach number for smooth AV field



(e) Pressure for discrete AV field



(f) Pressure for smooth AV field

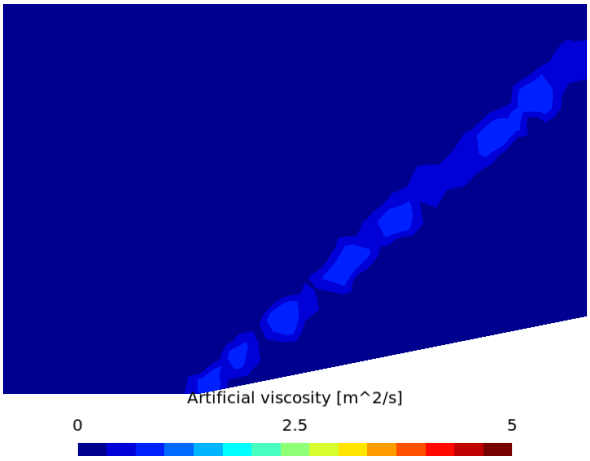
Figure 3.17: Solution to the wedge test case for discrete and smooth AV field at $p = 1$ with Lax-Friedrichs, $k = 0.05$, $s_0 = -2.0$, and $\kappa = 1.0$

Changing the polynomial order

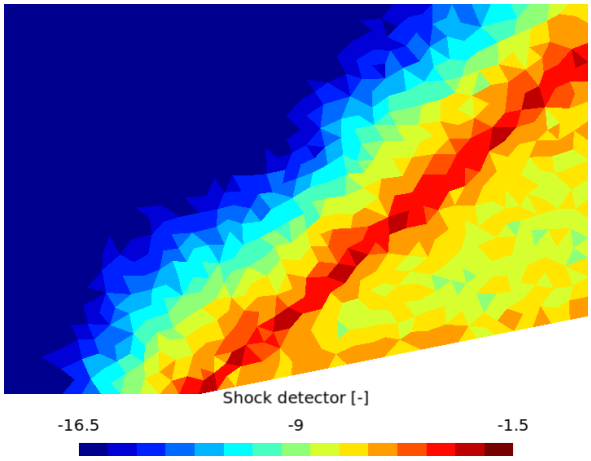
Tuning the parameters is a non-trivial task, even with a smooth artificial viscosity field. Therefore, increasing the polynomial order can be the simplest step to improve the results. This is investigated in this section.

Using a $p = 1$ solution as initial condition, the polynomial order was increased to $p = 3$. As a consequence, the artificial viscosity parameters were modified to $k = 0.05$, $s_0 = -4$, and $\kappa = 3$. With such set of parameters, the resulting artificial viscosity and sensor fields are shown in Figures 3.18(a) and 3.18(b), respectively. One striking difference is that the magnitude of artificial viscosity is much lower than at $p = 1$. This is normal however, since ε_e is computed by Equations (2.59) and (2.60), the former scaling with $1/p$. The shock sensor looks similar to previous results. Nevertheless, there is one difference worth mentioning. Although the values returned by the sensor are within the same range, the highest values are now concentrated in a thinner region, while more elements are assigned a -16 value before the shock. The reason is related to Equation (2.57), which explains how the sensor works. For a $p = 1$ solution, the two expansions can only be of order 1 and 0. At $p = 3$, the two expansions are then calculated at $p = 3$ and $p = 2$. In non-smooth regions, the comparisons will naturally give different results.

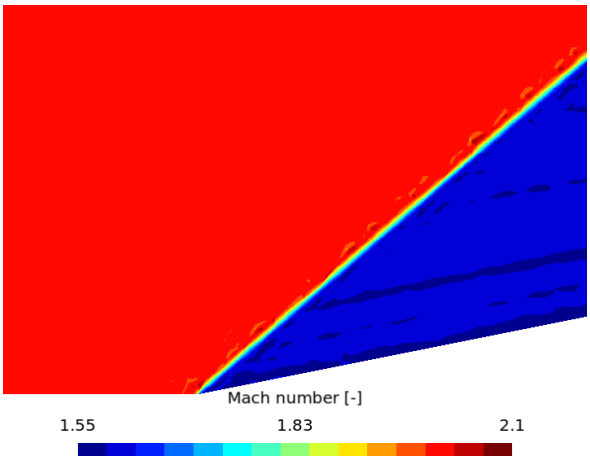
The results are significantly better than any of the $p = 1$ solutions for the Mach number and the pressure, as can be seen in Figures 3.18(c) and 3.18(d). The shock is sharply resolved and while some oscillations still remain in the pressure distribution, they are minor in comparison to those for $p = 1$. Darker lines parallel to the wall can be seen in the post-shock region of the Mach number distribution. These are caused by slight differences in shock strength along its length, thus leading to a slightly different post-shock Mach number value, which is then convected downstream. This feature is present in all results but is hidden in many cases by the size of colour intervals.



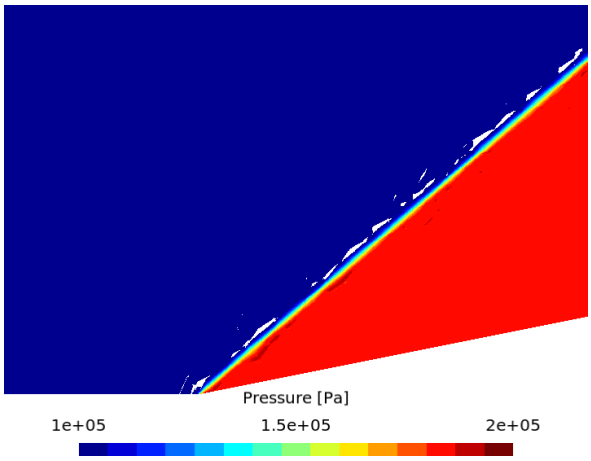
(a) Artficial viscosity



(b) Shock detector



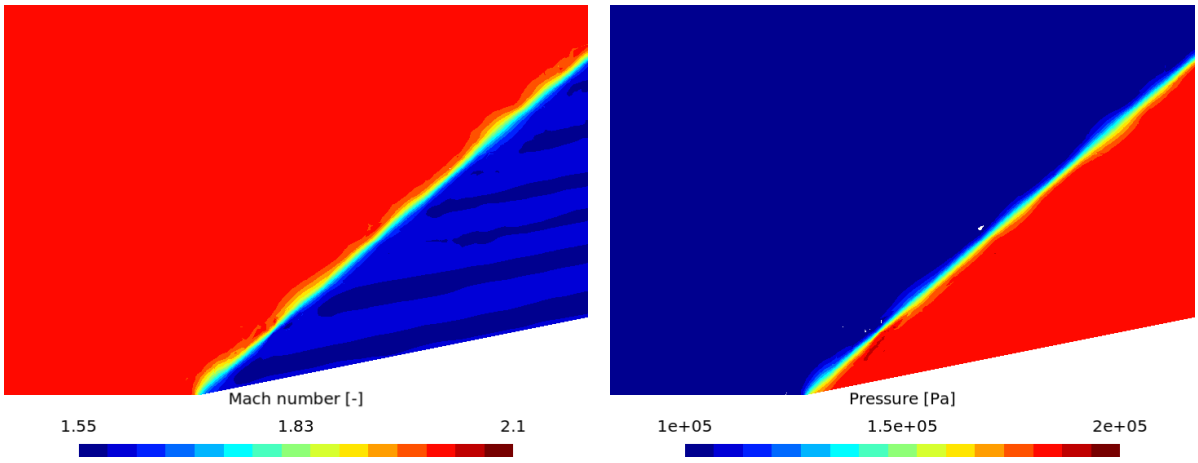
(c) Mach number



(d) Pressure

Figure 3.18: Solution to the wedge test case at $p = 3$ with Lax-Friedrichs, $k = 0.05$, $s_0 = -4.0$, and $\kappa = 3.0$

As mentioned before, a compromise may be needed between removing all oscillations and obtaining a sharp shock. To illustrate this, a different set of parameters was used. The previous set obtained a sharp shock while leaving some oscillations. The new set, given by $k = 0.9$, $s_0 = -3.0$, $\kappa = 1.0$, is more aggressive in removing oscillations but results in a larger and varying shock thickness. This can be clearly seen in Figures 3.19(a) and 3.19(b).



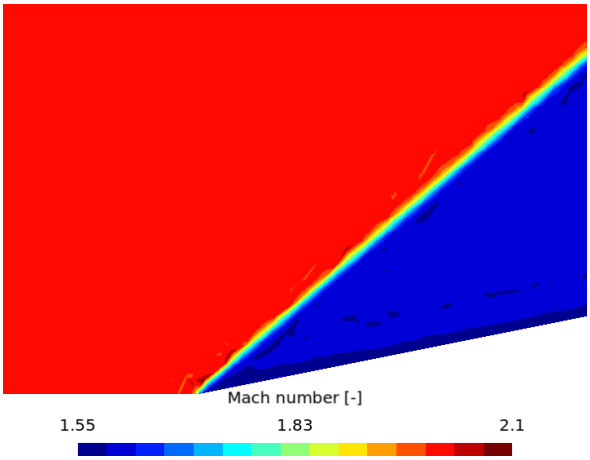
(a) Mach number (b) Pressure

Figure 3.19: Solution to the wedge test case at $p = 3$ with Lax-Friedrichs, $k = 0.9$, $s_0 = -3.0$, and $\kappa = 1.0$

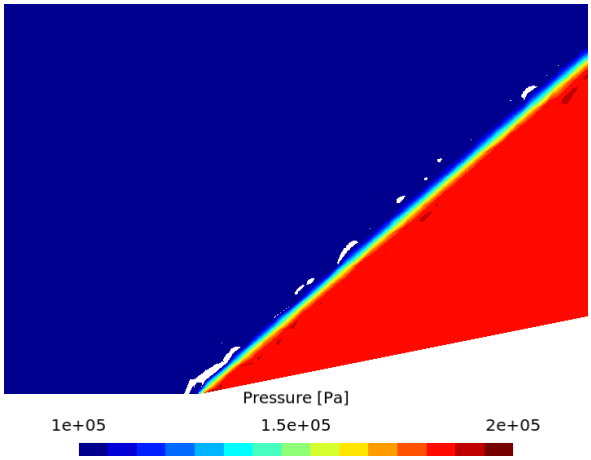
Changing the mesh refinement

Apart from increasing the polynomial order, refining the mesh should also lead to improved results. As mentioned before, three mesh refinements are considered and are shown in Figures 3.2, 3.3(a) and 3.3(b). The Mach number and pressure are plotted for each of the three mesh refinements. Figures 3.20(a) and 3.20(b) are for the coarse mesh, the fine mesh results are shown in Figures 3.20(c) and 3.20(d) and Figures 3.20(e) and 3.20(f) correspond to the mesh refined locally around the shock. In this case, the results are shown for $p = 2$ and the AV parameters used were $k = 0.05$, $s_0 = -3.0$ and $\kappa = 2.0$, with smoothing of the AV field applied. The $p = 2$ solutions were reached with order sequencing.

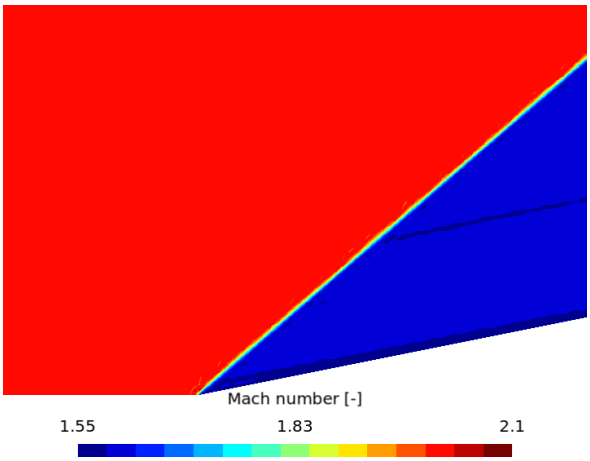
As expected, the results have improved with increasing refinement around the shock. Naturally, the smaller mesh elements have allowed to capture a thinner shock and reduced spurious oscillations around it. Refining the mesh locally has shown the clear advantage of an efficient element size distribution, allowing to better resolve the shock with fewer overall number of elements than the fine mesh. Nevertheless, it is worthwhile keeping in mind that the point of using a high-order method is to reduce the need to refine the mesh by refining in p instead.



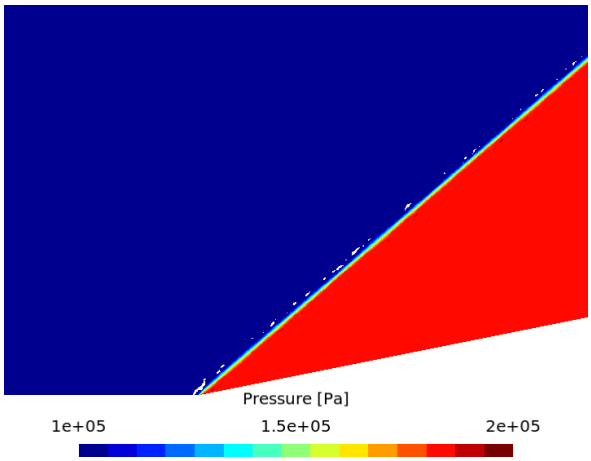
(a) Mach number for baseline mesh



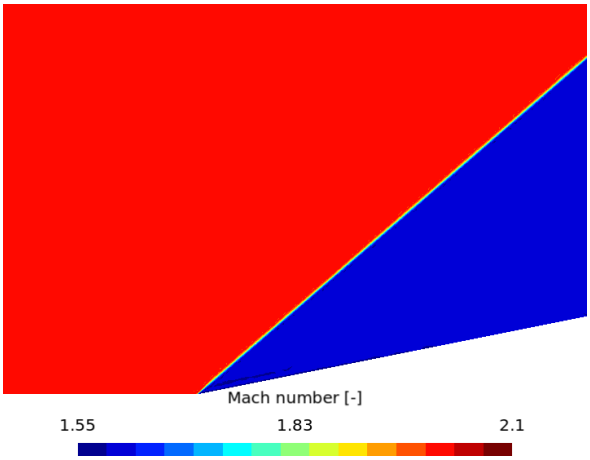
(b) Pressure for baseline mesh



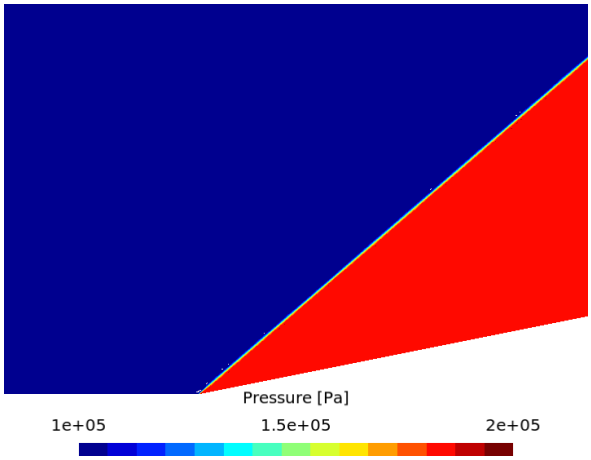
(c) Mach number for isotropic fine mesh



(d) Pressure for isotropic fine mesh



(e) Mach number for locally refined mesh



(f) Pressure for locally refined mesh

Figure 3.20: Solution to the wedge test case for different mesh refinements at $p = 2$ with Lax-Friedrichs, $k = 0.05$, $s_0 = -3.0$ and $\kappa = 2.0$

3.3. Mach 4 challenges

At a Mach number of 4, it became much more difficult to tune AV parameters. Many sets can fail to stabilise the simulation. The main difficulty was that in order to keep the simulation numerically stable, the AV needed to be applied on a region so large that it covers almost all the post-shock area. This results in the post-shock conditions being obtained in a very small portion of the post-shock area. The analytical solution to this problem is given in Table 3.2, and the shock angle with respect to the horizontal is now 23.4°.

Table 3.2: Pre- and post-shock conditions for the wedge test case at Mach 4

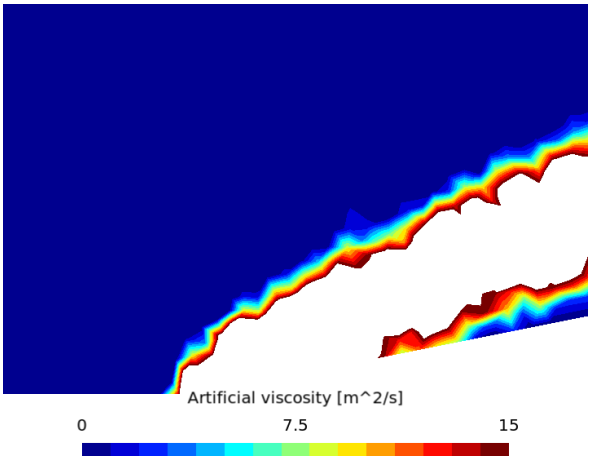
Parameter	Pre-shock	Post-shock
Mach number [-]	4	3.1931
Static pressure [Pa]	101325	282449
Static temperature [K]	288	398

The problem is solved as a steady computation, but as before a GMRES algorithm with Jacobi preconditioning is used to solve the linear system. The results are shown at $p = 1$ only because higher orders were not achieved. The baseline and isotropic fine meshes used at $M = 2$ and shown in Figures 3.2 and 3.3(a) are reused here. The first results are shown for the baseline mesh.

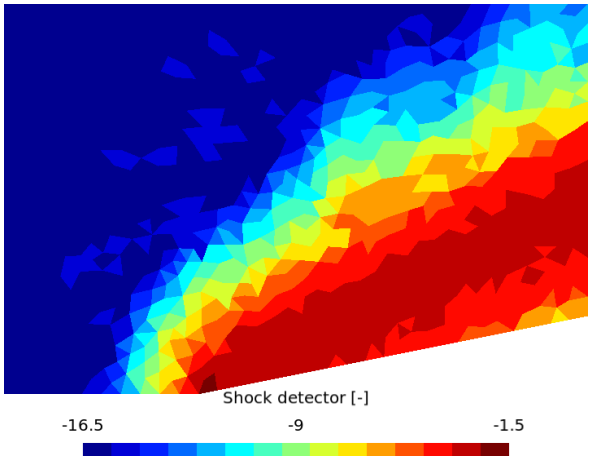
Baseline mesh

The artificial viscosity field, shown in Figure 3.21(a) illustrates precisely the issue mentioned while introducing the $M = 4$ case. Spurious oscillations growing in the vicinity of the shock tend to become large enough that they crash the simulation. To avoid this, it was necessary to apply AV on a very large region and in large amounts, as highlighted by the large clipped region. One can see how large AV footprints could become an issue if quantities at the wall were of interest. For this case, the AV parameters used were $k = 0.25$, $s_0 = -3.0$, and $\kappa = 2.0$. The shock sensor values are shown in Figure 3.21(b), which in magnitude are similar to the Mach 2 case. However, more mesh elements are assigned the highest values, meaning that even for a high threshold the AV field would be relatively large.

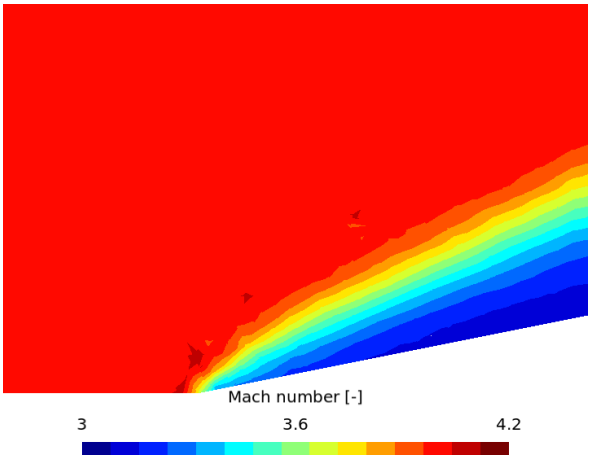
The Mach number distribution is shown in Figure 3.21(c). The shock is thick as expected, but it can be seen how its thickness varies, especially reducing around the beginning of the ramp. Due to the AV applied, spurious oscillations are minimal. Figure 3.21(d) shows the pressure distribution. The shock is in this case of constant thickness, and even though a white region appears suggesting strong spurious oscillations, in fact they are minor as well. The white areas appear mainly due to a stricter clipping applied on the lower pressure bound.



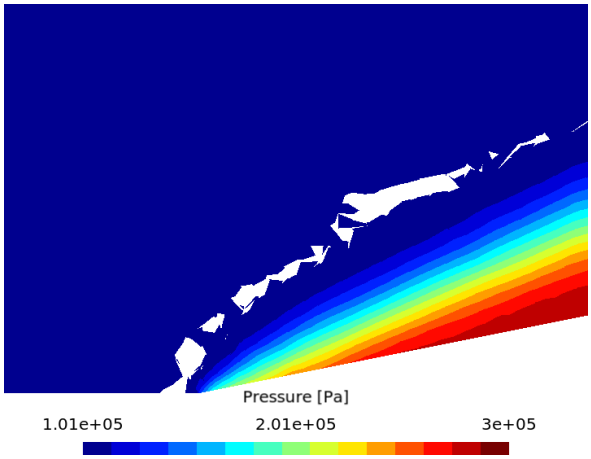
(a) Artificial viscosity



(b) Shock detector s_e



(c) Mach number



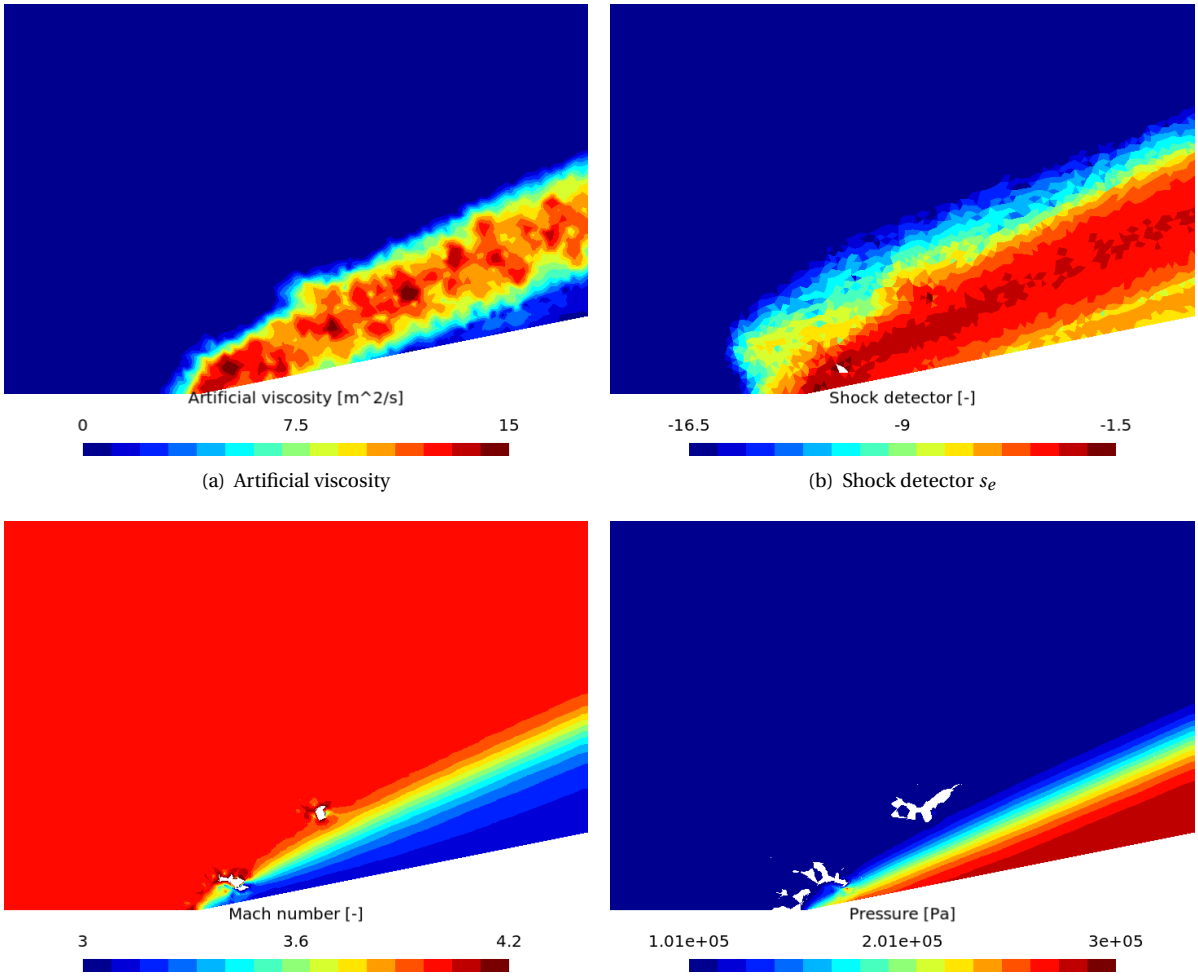
(d) Pressure

Figure 3.21: Solution to the wedge test case at Mach 4 on the baseline mesh at $p = 1$ with Lax-Friedrichs, $k = 0.25$, $s_0 = -3.0$, and $\kappa = 2.0$

Isotropic fine mesh

It was seen at $M = 2$ that the shock thickness can be reduced without creating spurious oscillations by refining the mesh. This section thus presents the results on the fine mesh. The artificial viscosity field shown in Figure 3.22(a) is consistent with previous results. The magnitude is lower overall, which is expected because the finer mesh should lead to a smoother solution and because the computation of ε scales with h. The sensor values are shown in Figure 3.22(b), which show the similar trend observed when the mesh was refined at $M = 2$. A much larger pre-shock region is attributed the -16 value, indicating that spurious oscillations are more concentrated near the shock.

The Mach number distribution, which is shown in Figure 3.22(c), contains two spots where the error peaks. These spots are seen in the pressure field as well, shown in Figure 3.22(d). In both cases, the shock is thinner though still much larger than in the $M = 2$ solutions. Burgess and Mavripilis [13] face the same issue, although at Mach 7 and wedge inclination of 15°. They need to use an extreme refinement around the shock to tackle this issue, as can be seen in Figures 3.23 and 3.24. The white spots in Figures 3.22(c) and 3.22(d) are what prevent to use this solution as initial condition for a higher-order simulation. The error is amplified further and causes the simulation to become unstable. These spots where the error peaks, in particular near the point where the ramp starts, occur if the coarse mesh is used for higher p as well.



(a) Artificial viscosity

(b) Shock detector s_e

(c) Mach number

(d) Pressure

Figure 3.22: Solution to the wedge test case at Mach 4 on the isotropic fine mesh at $p = 1$ with Lax-Friedrichs, $k = 0.25$, $s_0 = -3.0$, and $\kappa = 2.0$

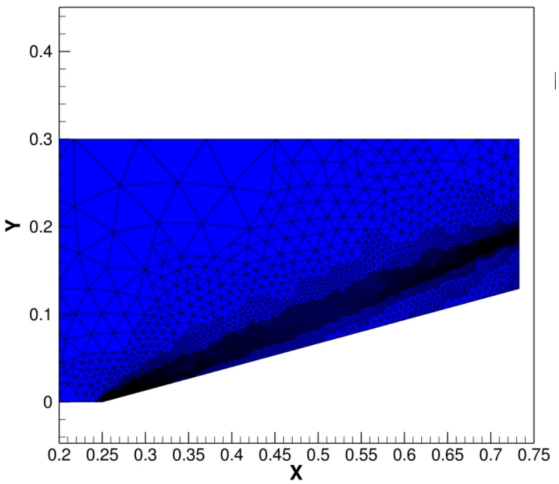


Figure 3.23: Fine mesh used by Burgess & Mavripilis [13]

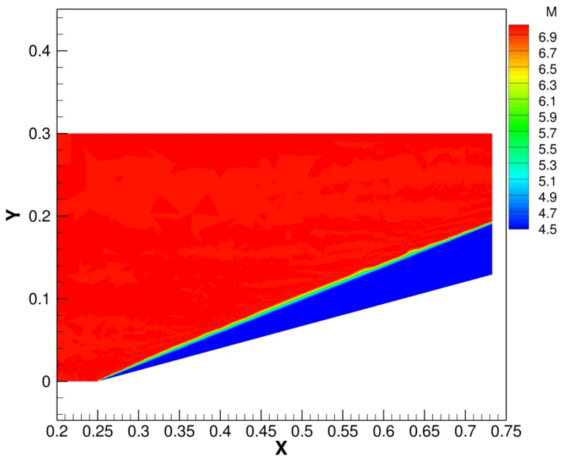


Figure 3.24: Mach number obtained by Burgess & Mavripilis [13] on the fine mesh

3.4. Summary

To end this chapter, the key findings are summarised in this short paragraph. At $M = 2$, finding an appropriate set of AV parameters was found to be challenging for $p = 1$ and the coarse mesh. The threshold s_0 and interval κ need to be tuned in a coordinated fashion such that ideally the maximum AV magnitude is applied where the sensor values peak and such that the shock is sufficiently dissipated to be captured. The AV magnitude is then tuned to provide sufficient dissipation. Nevertheless, the cases on the coarse mesh at $p = 1$ showed that a trade-off needs to be made between removing spurious oscillations and keeping a shock sharp. Increasing the order p or decreasing the element size h improved the results significantly, both in sharpening the shock and removing oscillations. Furthermore, smoothing the AV field was found to be beneficial although in this case the better results may have been due to the resulting wider AV field. The results at Mach 4 highlighted a difficulty in keeping the simulation stable. On both the fine and coarse meshes, the shock width was large, leaving only a very small portion of the domain with the actual post-shock conditions. This could become an issue should quantities at the wall be of interest, since they would be polluted by the artificial viscosity. Spots where the error peaks prevented simulating the problem at orders higher than $p = 1$.

4

Application to inviscid, unsteady problems

Unsteady problems present new challenges for shock capturing, in particular if the shock strength varies in time. In such case, the sensor values at the shock are expected to also vary in time. Therefore, it can be challenging to find an appropriate threshold. Furthermore, unsteady problems do not allow the use of order sequencing, where the high-order solution for an order p is started from the solution at order $p-1$. In other words, it is not possible to start a high-order simulation from an already good low-order solution. In addition to these challenges, the unsteady problems analysed in this chapter are relevant for steady hypersonic simulations which can also exhibit strong variations while the transient converges to its steady-state. Finally, it is interesting to explore unsteady problems because they have received less attention with the latest artificial viscosity methods.

The first test case considered in this chapter has a moving shock of constant strength and the performance of the AV method is assessed for different mesh and order refinements, as well as various inviscid flux schemes. The second test case contains a moving shock of varying strength, making the problem much more challenging to simulate. Both are used extensively in the literature about shock detectors.

4.1. The Sod problem

The Sod shock tube problem [80] is a specific implementation of the Riemann problem and tests the performance of a method in treating three types of waves in motion in one dimension, for the Euler equations. The domain $x \in [0,1]$ is split in two equal parts and each has a different set of initial conditions, given by:

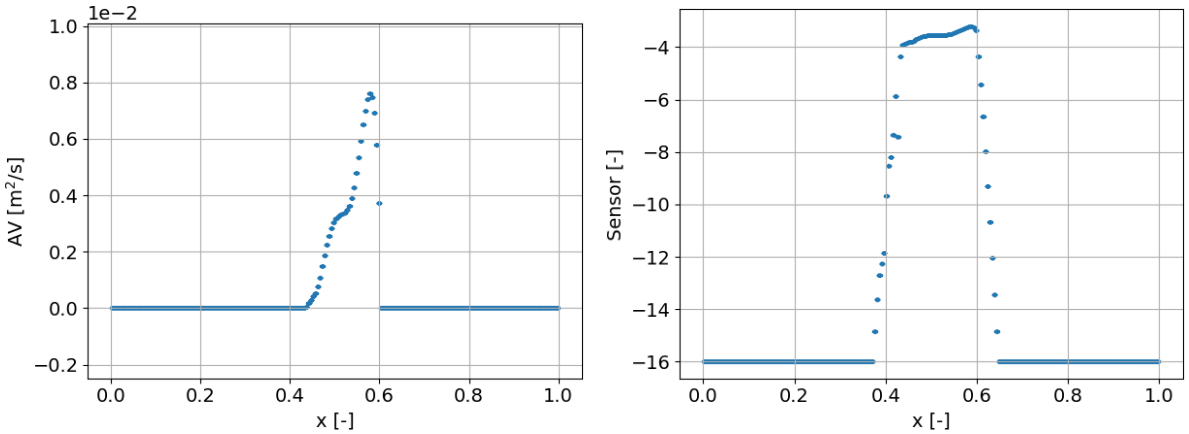
$$\begin{cases} \rho_L = 1, & \rho_R = 0.125 \\ P_L = 1, & P_R = 0.1 \\ u_L = 0, & u_R = 0 \end{cases} \tag{4.1}$$

where the subscript L stands for left side and R for right side. At time $t = 0$, four regions form, separated by three waves: a shock moving to the right, an expansion fan moving to the left and a contact discontinuity between the other two waves. The simulation advances with a time step $\Delta t = 0.0001$ and is stopped after 400 time steps. As for the wedge test case, the BDF2 scheme is used to march in time and a GMRES algorithm solves the linear system of equations. In this case, iterations were quick and memory use was mild thus ILU with one level of fill-in was used as preconditioning for better convergence. The Lax-Friedrichs Riemann solver is chosen for the inviscid flux again for stability reasons, and the artificial viscosity takes the parameters $k = 0.05, s_0 = -3, \kappa = 1.0$ for all simulations. Since the problem contains a contact discontinuity, the density is used again as variable for the shock detector. The results in this chapter are shown at the last time step on a mesh containing 200 elements (200 in x and 1 in y), unless mentioned otherwise. This number of elements was chosen for being considered a middle point between a coarse and a fine mesh, and for comparison purposes since the same number was chosen in [57, 71]. Since the mesh is technically 2D with one cell in the y direction, there is a wall at the top and bottom of the domain. It takes a slip adiabatic condition. In the x direction, the left and right boundaries are simply implemented as slip adiabatic walls as well because the simulation is stopped well before waves reach the endpoints of the domain. Due to the simplicity of the problem, a short programme described in Appendix A was written to compute the analytical solution at each

point in the domain at each point in time. This exact solution is used for comparison with the results shown next.

4.1.1. Baseline results at low order

The effect of adding artificial viscosity is investigated first at low order, specifically at $p = 1$. Since the simulation is 1D and the amount of AV used is small, there is not much need to smooth the AV field and the option was thus not used. Figures 4.1(a) and 4.1(b) show the AV magnitude and shock detector results at the last time step. With the parameters chosen, artificial viscosity will only be applied on elements for which the sensor gives a value larger than $s_0 - \kappa = -4$. Comparing the magnitude and shock detector results, this is indeed the case. The shock detector returns a value of -16 about everywhere except where the expansion, contact and shock waves are present. This is expected because the solution has not changed from its smooth initial conditions there. The intensity of artificial viscosity is the highest at the shock, and a plateau can be seen around the contact discontinuity.



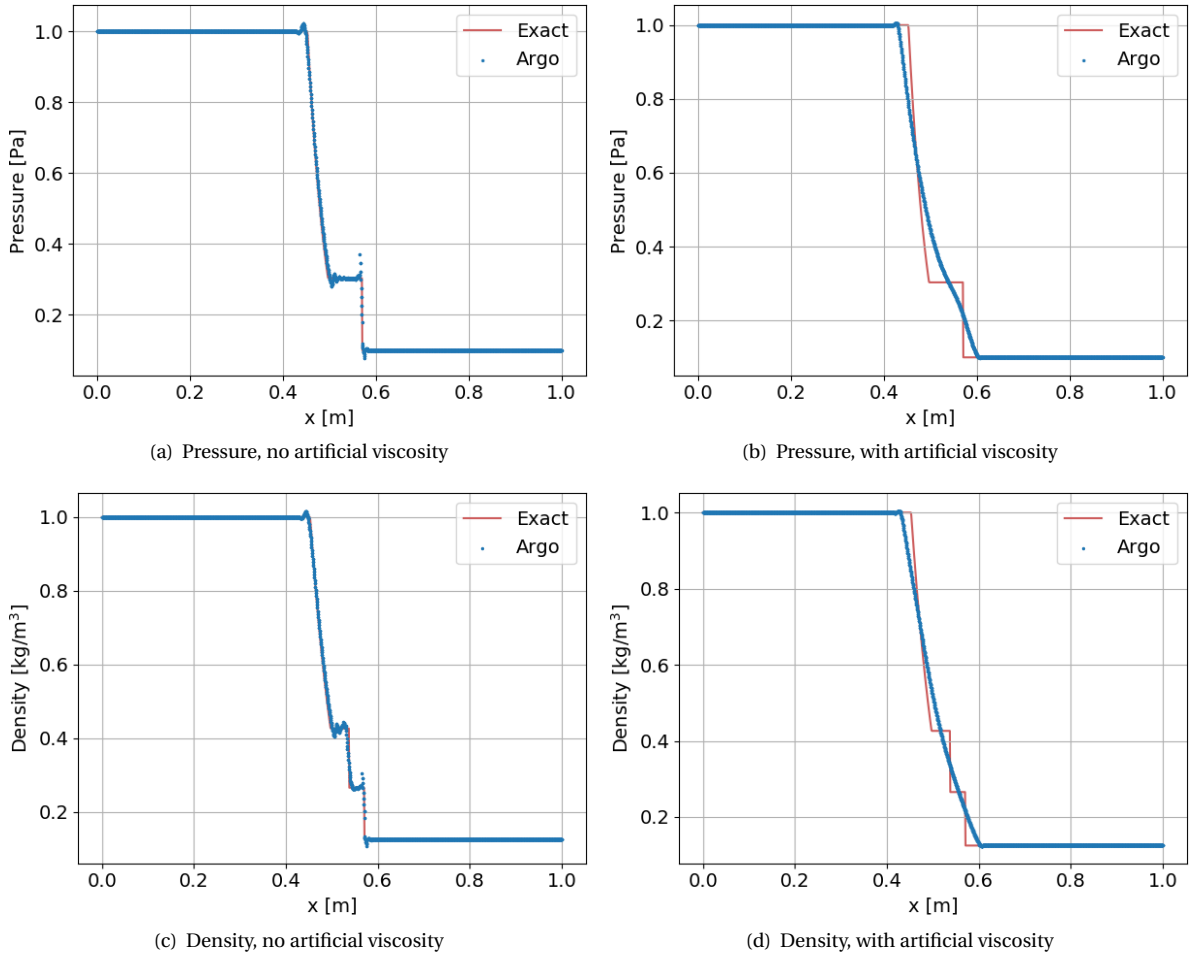
(a) Magnitude

(b) Shock detector

Figure 4.1: Solution to the Sod test case for artificial viscosity variables, at $p = 1$ with 800 DOF, Lax-Friedrichs, $k = 0.05$, $s_0 = -3$, and $\kappa = 1.0$

Figures 4.2(a) and 4.2(b) show the pressure results without and with artificial viscosity, respectively. For roughly $x < 0.45$ and $x > 0.57$, the solution takes the left and right state values given by Equation (4.1). Between approximately $x = 0.45$ and $x = 0.5$, the solution shows the presence of the expansion wave. At $x \approx 0.57$, the sudden jump in pressure corresponds to the shock. At the shock, the artificial viscosity has clearly removed the oscillations but it has also completely smoothed the shock away. Only a very subtle change in curvature hints at the presence of the shock, thus clearly indicating that excessive AV has been applied. At the front of the expansion wave, the overshoot has been diminished but not removed completely. This suggests that insufficient artificial viscosity has been added at that point. As for the wedge test case, the choice between removing completely the oscillations and obtaining a sharp solution at discontinuities is not trivial. The results for the density, shown in Figures 4.2(c) and 4.2(d), have an additional jump between the expansion wave and the shock. This is the contact discontinuity. Not even a subtle change in curvature appears near the two discontinuities, however the overshoot at the expansion wave front remains visible.

To reduce the global error, one would need to increase the threshold such that artificial viscosity is only applied on elements being assigned a sensor value slightly above -4. In that way, AV would be added at the shock and contact discontinuity but not on the expansion wave. Possibly, the magnitude k would have to be reduced as well. These changes would result in a better solution around the shock and contact discontinuities but would increase the overshoot at the expansion wave front. For the current mesh refinement, there is thus a limit to how good the results can be at $p = 1$.



(a) Pressure, no artificial viscosity

(b) Pressure, with artificial viscosity

(c) Density, no artificial viscosity

(d) Density, with artificial viscosity

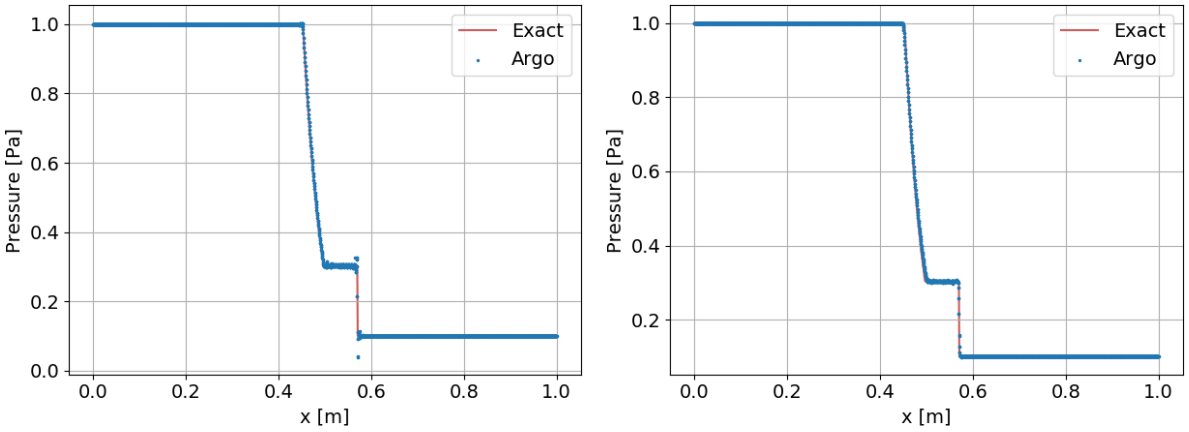
Figure 4.2: Solution to the Sod test case at $p = 1$ with 800 DOF, Lax-Friedrichs, $k = 0.05$, $s_0 = -3$, and $\kappa = 1.0$

4.1.2. Results at high order

Based on the observations made for the wedge test case, it is expected that the shock can be better captured if the order p is refined. For a high-order solution at $p = 4$, Figures 4.3(a) and 4.3(c) show that spurious oscillations appear if no AV is used, as expected. Adding artificial viscosity (with no smoothing) largely removes the oscillations, including the overshoot at the front of the expansion wave. The discontinuities are sharply captured, but the solution is smoother for the pressure (Figure 4.3(b)) than for the density (Figure 4.3(d)). The reason becomes clear when looking at Figure 4.4(a). Artificial viscosity is added at the shock location but not at the contact discontinuity. Therefore, the density has an additional discontinuity which is not dissipated, despite the sensor being able to detect it. The values returned by the shock detector are shown in Figure 4.4(b). It can be seen that the contact discontinuity is much more difficult to identify and that only one element remains above the cut at -4. The AV parameters could have been tuned such that AV is also added at the contact discontinuity location, but this would have resulted in a shock less sharply resolved.

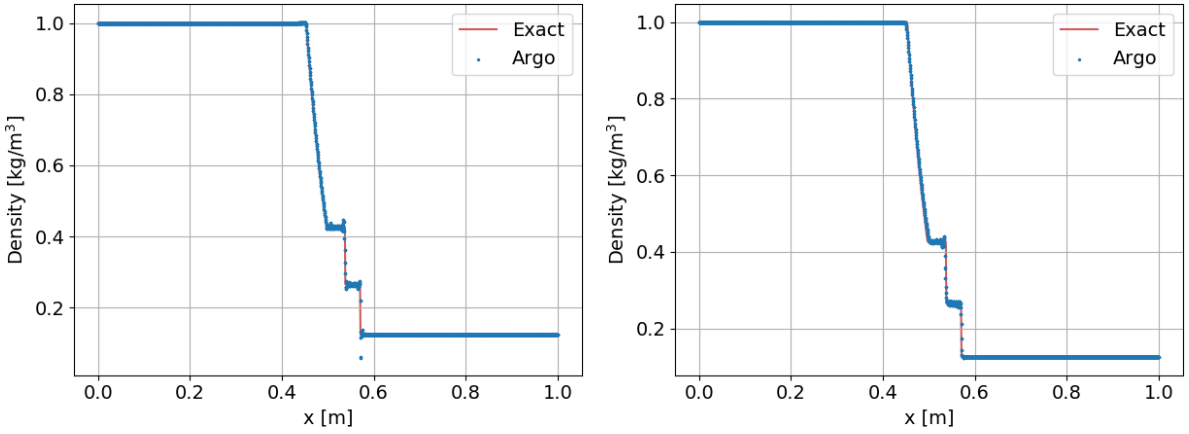
Lv et al. [57] observed a similar shock detector pattern, and noted that the contact would be dissipated at the beginning of the simulation when both the contact and the shock are contained in the same cell. With the detector of Persson and Peraire and a different DG scheme, they obtain similar results for the density. That is the minor overshoot at the expansion wave front is largely removed and the constant regions between the shock and expansion contain spurious oscillations of small amplitude. They also show that this issue can be mitigated with a different threshold. The results for the two other sensors in their study are smoother but match less closely the exact solution. Pressure results are not shown in their paper. In the comparative study of shock detectors by Qiu & Shu [71], the Shu-Osher problem is computed with RKDG and stabilised with a WENO reconstruction method instead of artificial viscosity. For different sensors, the results are less oscillatory and match closely the exact solution, even at lower orders. This highlights that WENO methods

may be more accurate than artificial viscosity, at least for certain problems. It is worth noting that the shock detector from Persson and Peraire was not included in their study.



(a) Pressure, no artificial viscosity

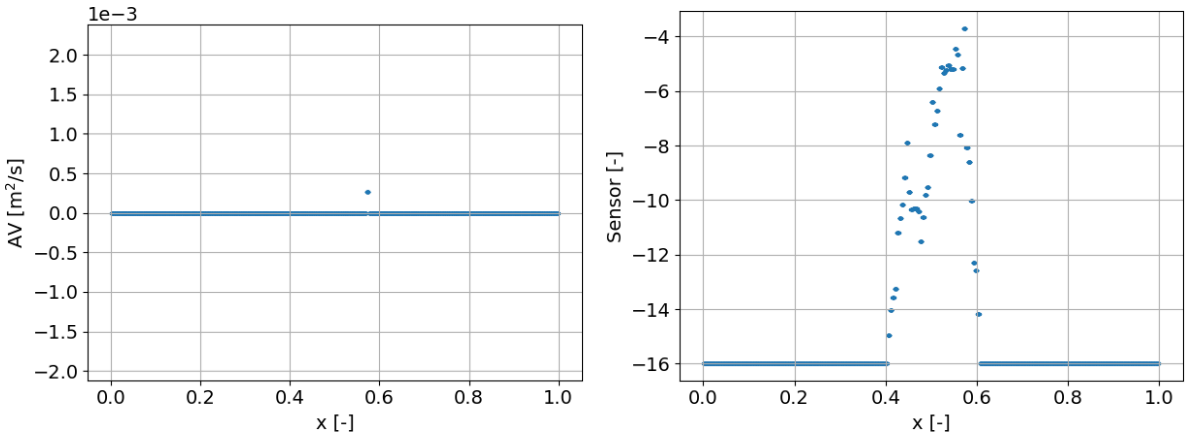
(b) Pressure, with artificial viscosity



(c) Density, no artificial viscosity

(d) Density, with artificial viscosity

Figure 4.3: Solution to the Sod test case at $p = 4$ with 5000 DOF, Lax-Friedrichs, $k = 0.05$, $s_0 = -3$, and $\kappa = 1.0$



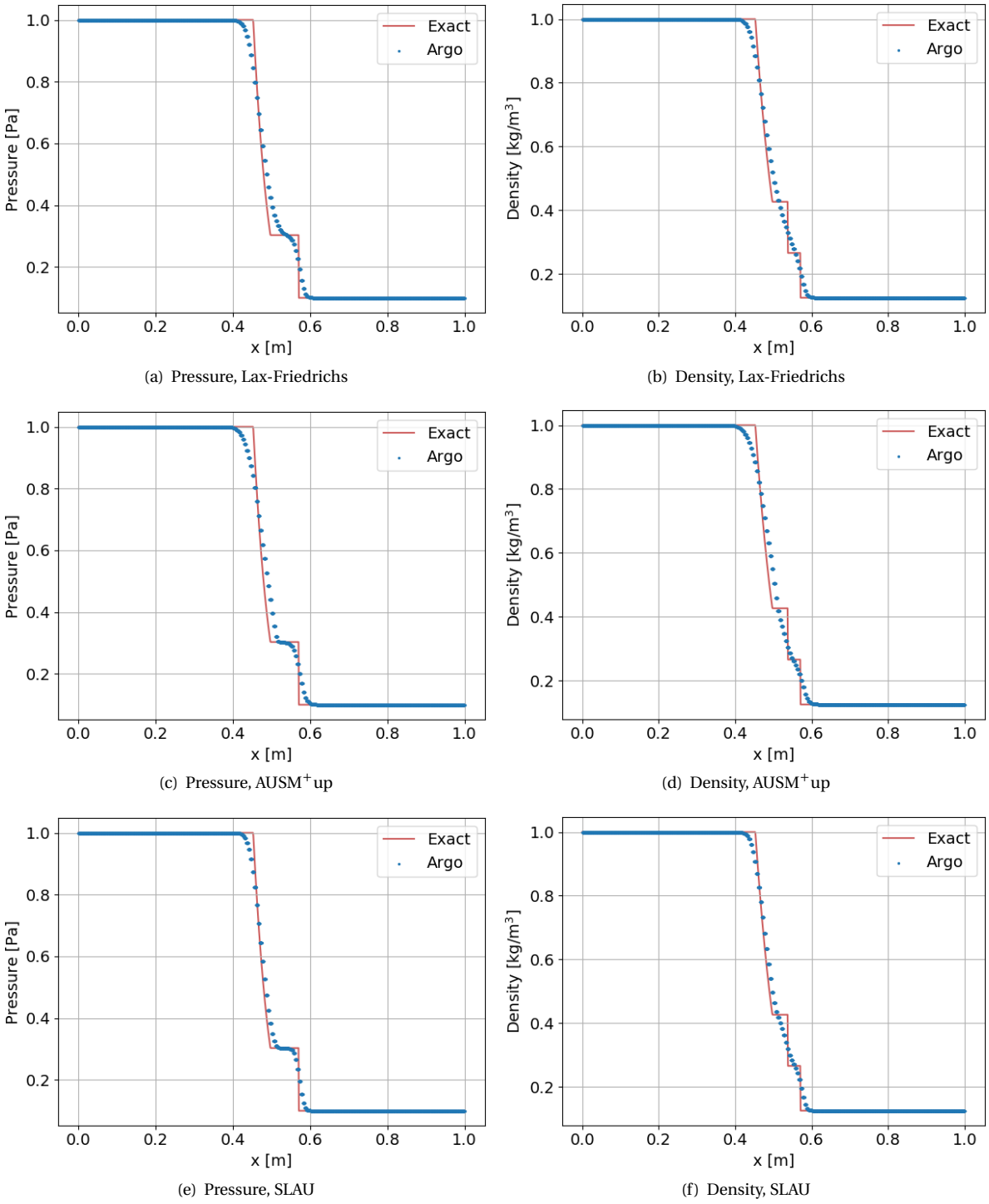
(a) Magnitude

(b) Shock detector

Figure 4.4: Solution to the Sod test case for artificial viscosity variables at $p = 4$, with 5000 DOF, Lax-Friedrichs, $k = 0.05$, $s_0 = -3$, and $\kappa = 1.0$

4.1.3. Results for different Riemann solvers

An important feature in FVM which was not investigated in the previous chapter is the inviscid flux scheme. For the Sod problem, the influence of the inviscid flux scheme was first tested at $p = 0$. Pressure and density are shown for the Lax-Friedrichs, AUSM$^+$up, and SLAU Riemann solvers in Figures 4.5(a) to 4.5(f). As expected, Lax-Friedrichs is the most dissipative of the three, although the result agrees better with the exact solution than that of AUSM$^+$up at the expansion wave front. SLAU is clearly the best of the three.



(a) Pressure, Lax-Friedrichs

(b) Density, Lax-Friedrichs

(c) Pressure, AUSM$^+$up

(d) Density, AUSM$^+$up

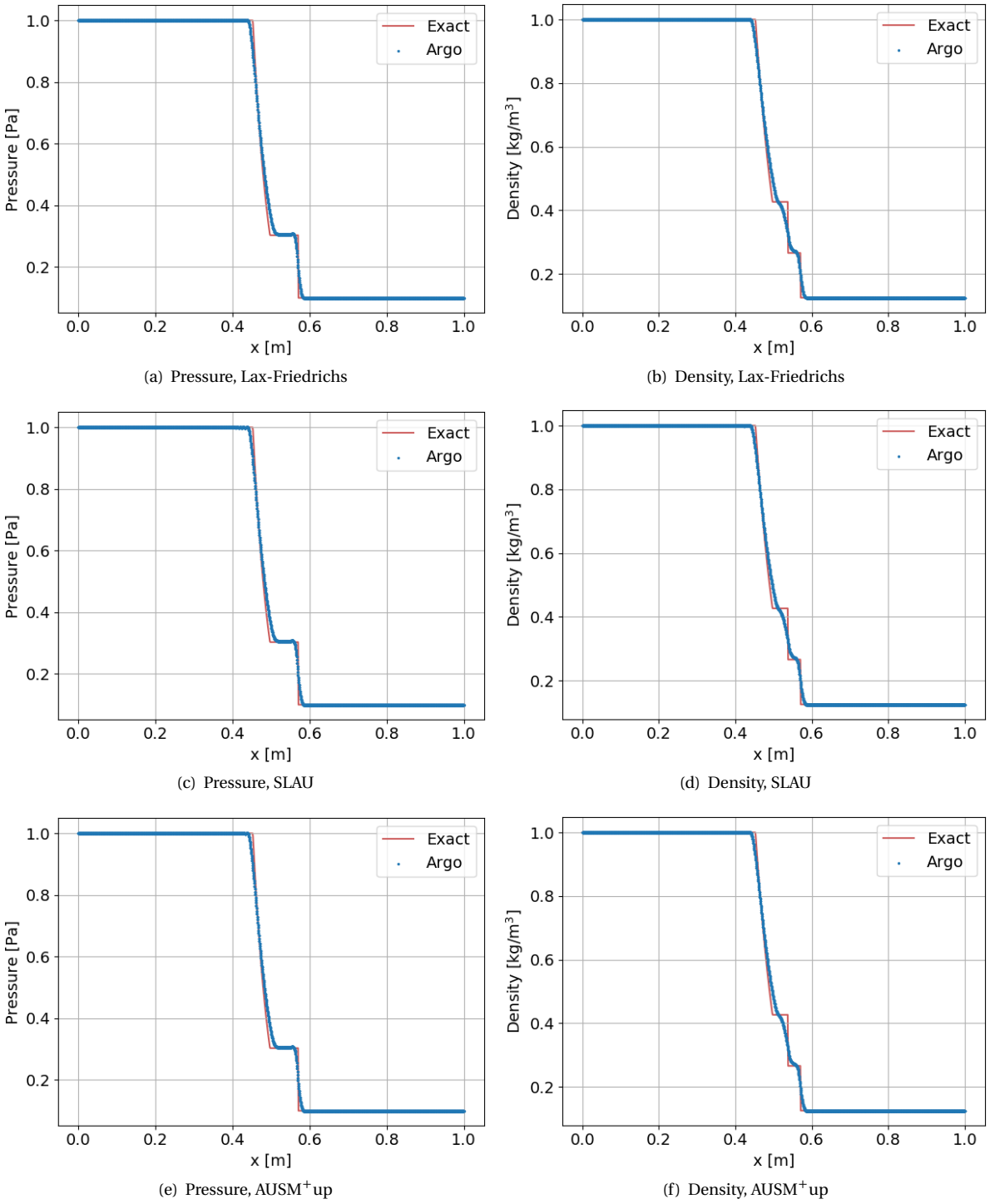
(e) Pressure, SLAU

(f) Density, SLAU

Figure 4.5: Solution to the Sod test case at $p = 0$ for different Riemann solvers, 200 DOF

At higher orders, the Riemann solver is found to have little influence on the results. Note that in this case

it was chosen to smooth the AV field to help with stability, thinking that less dissipative inviscid flux schemes could have been an issue. In fact, smoothing has little effect for the Sod test case and doing it here was not necessary. The threshold was lowered to $s_0 = -4$, also to ensure stability. Since the conclusions were the same at different polynomial orders, the results presented next are only for $p = 2$. Figures 4.6(a), 4.6(c) and 4.6(e) show the pressure for the Lax-Friedrichs, AUSM$^+$up, and SLAU fluxes respectively and no differences can be seen. The same is true for the density, shown in Figures 4.6(b), 4.6(d) and 4.6(f).



(a) Pressure, Lax-Friedrichs

(b) Density, Lax-Friedrichs

(c) Pressure, SLAU

(d) Density, SLAU

(e) Pressure, AUSM$^+$up

(f) Density, AUSM$^+$up

Figure 4.6: Solution to the Sod test case at $p = 2$ for different Riemann solvers, with 1800 DOF $k = 0.05$, $s_0 = -4$, and $\kappa = 1.0$

It is important to stress again how tuning the AV parameters is a non-trivial task. If one could have argued

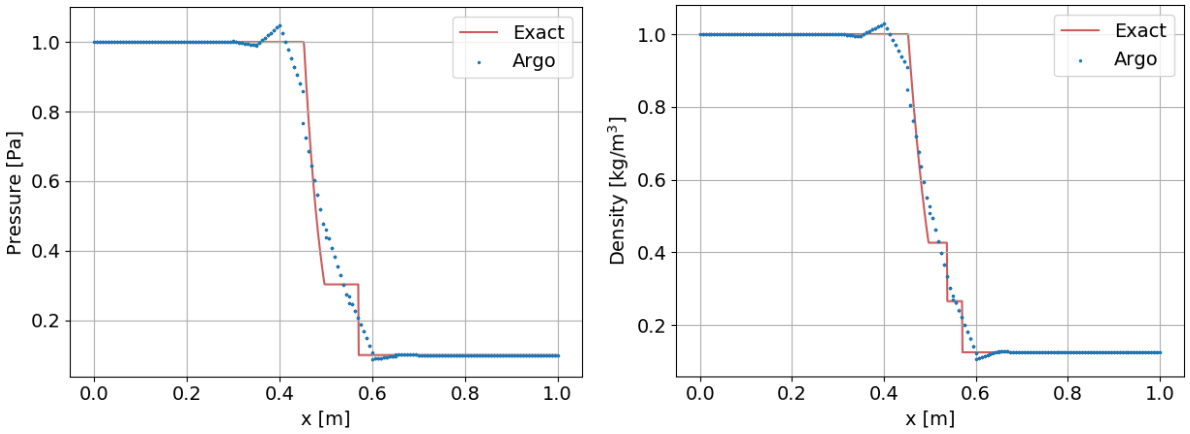
for the wedge test case that the parameters proposed by Barter & Darmofal [5] would have given a smooth (yet wide) shock, in this case the values are clearly inappropriate. The threshold $s_0 = -4.0$ already appears to be somewhat too low, the value proposed by Barter and Darmofal $s_0 = -(4+4.25\log_{10}(p)) = -5.28$ would only make matters worse. This work thus finds that AV values need to be adapted from one problem to another, even if the polynomial order is the same.

4.1.4. Results on a coarse mesh

After seeing that the results match the exact solution well at $p = 2$ and that the Riemann solver has no visible effect except at $p = 0$, it is worthwhile assessing whether similar results can still be achieved with a much coarser mesh. The previous results were obtained with a mesh containing 200 elements. As mentioned before, the key advantage of high-order methods is to be able to lower the number of elements by increasing the order of the method. To investigate this aspect, the problem is simulated again on a coarse mesh containing only 20 elements.

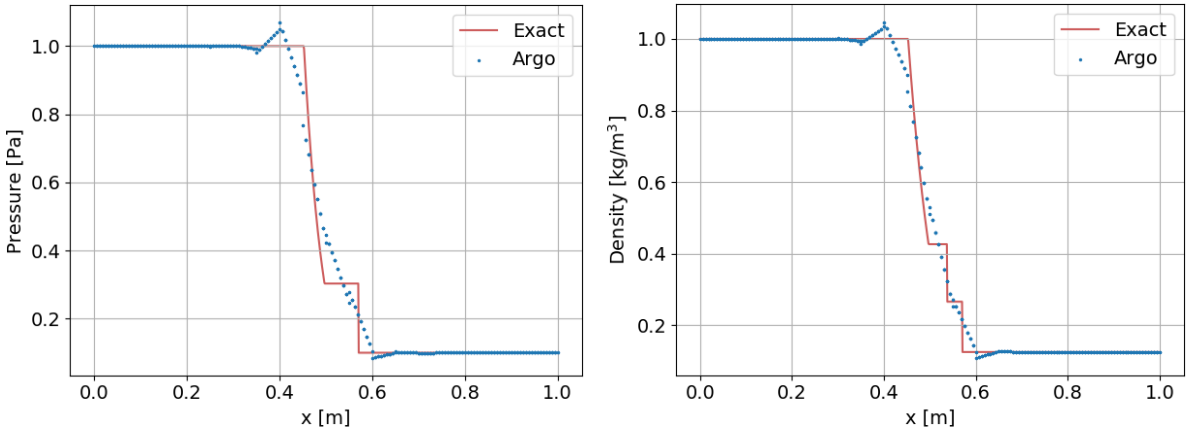
The first results at $p = 1$ are given in Figures 4.7(a) and 4.7(b) using the Lax-Friedrichs flux and in Figures 4.7(c) and 4.7(d) for the SLAU flux. Both use $k = 0.05$, $s_0 = -1.0$, and $\kappa = 1.0$ as AV parameters. As expected, the quality of the solution is considerably lower due to the coarser mesh and the low order. This deterioration is consistent with what Vuik & Ryan [87] obtain on a mesh containing 64 elements. Even with low refinements in mesh and order, the effect of the Riemann solver is barely noticeable. The Lax-Friedrichs solution has slightly damped spurious oscillations compared to the SLAU solution, which again is expected since the Lax-Friedrichs flux is known for being excessively dissipative. One feature worth mentioning is the jump in the expansion wave. This could be an entropy glitch, which can only happen in sonic expansion waves [83]. Since the schemes are entropy-satisfying, this jump tends to zero as h tends to zero and that would explain why it did not appear on the finer mesh.

The artificial viscosity and in particular the shock detector plots, shown in Figures 4.8(a) and 4.8(b) look very different to those for the 200 elements mesh. The shock detector does not return a -16 value for a single element, and the value returned at the shock is much larger. It would make sense to flag the elements corresponding to the three highest sensor values, since that is the region between the shock and expansion wave fronts. This is achieved by setting the threshold to -1 and the interval to 1, so that the cut happens at $s_0 - \kappa = -2$. For the artificial viscosity magnitude, the same trend as in Figure 4.1(a) can be seen.



(a) Pressure, Lax-Friedrichs

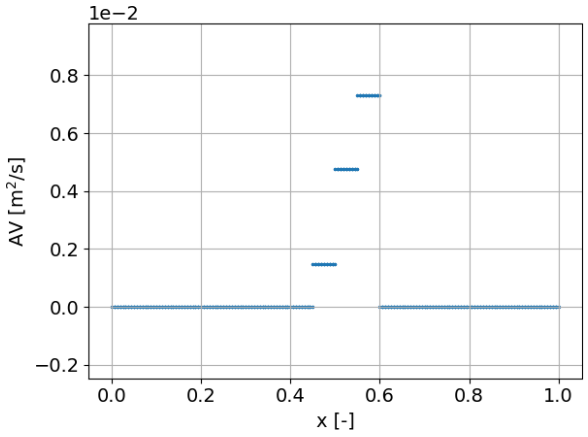
(b) Density, Lax-Friedrichs



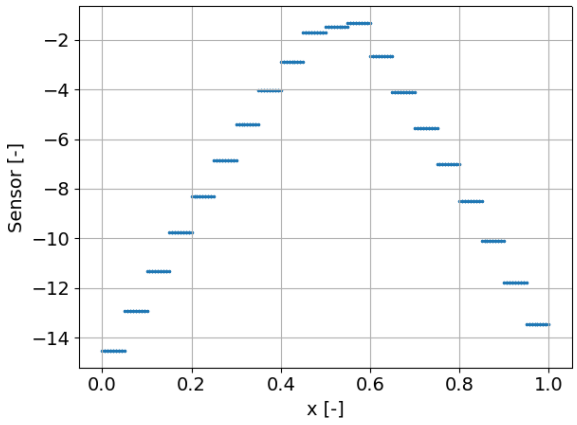
(c) Pressure, SLAU

(d) Density, SLAU

Figure 4.7: Solution to the Sod test case at $p = 1$ for different Riemann solvers, with 80 DOF, $k = 0.05$, $s_0 = -1.0$, and $\kappa = 1.0$



(a) Magnitude

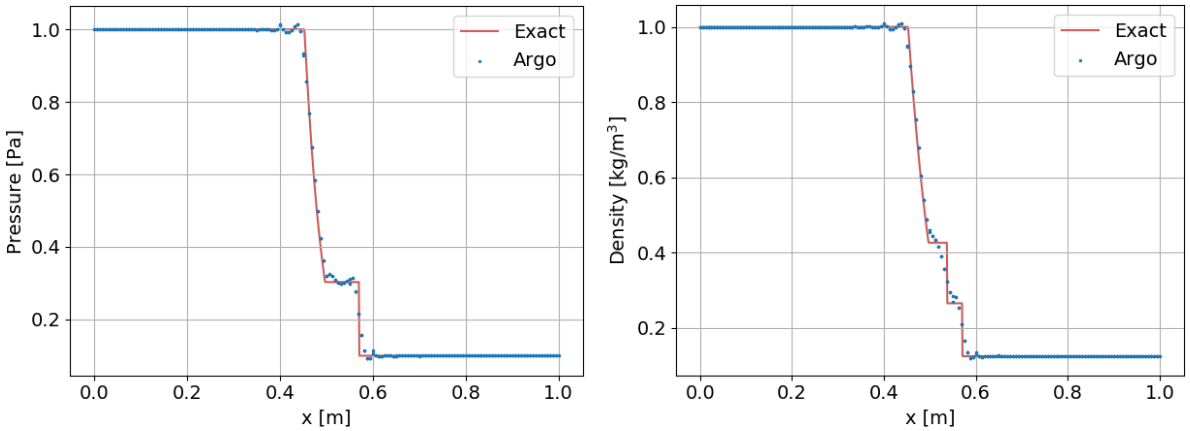


(b) Shock detector

Figure 4.8: Solution to the Sod test case for artificial viscosity variables, obtained at $p = 1$ with 80 DOF Lax-Friedrichs, $k = 0.05$, $s_0 = -1.0$, and $\kappa = 1.0$

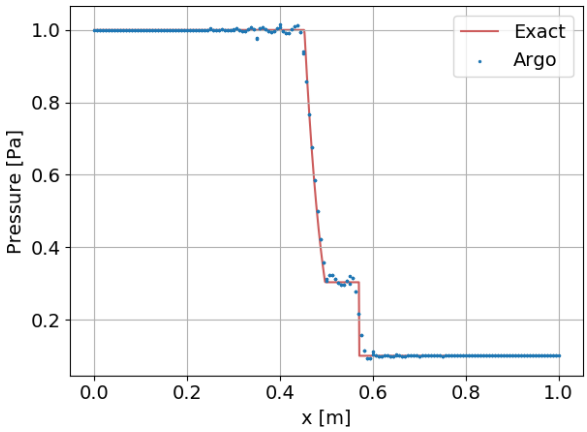
For a high-order p, the quality of the solution should improve significantly. Figures 4.9(a) and 4.9(b) show the pressure and density obtained with the Lax-Friedrichs flux at $p = 4$. Although the solution is indeed a major improvement with respect to that at $p = 1$, spurious oscillations remain and the discontinuities in density in particular are poorly represented. The Riemann solver only has a minor impact again, as can be

seen in Figures 4.9(c) and 4.9(d), which are solutions computed with SLAU. Since spurious oscillations remain relatively strong, Lax-Friedrichs gives better results in this case.

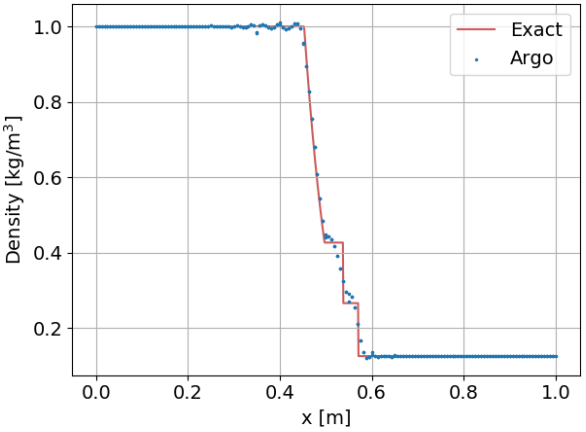


(a) Pressure, Lax-Friedrichs

(b) Density, Lax-Friedrichs



(c) Pressure, SLAU

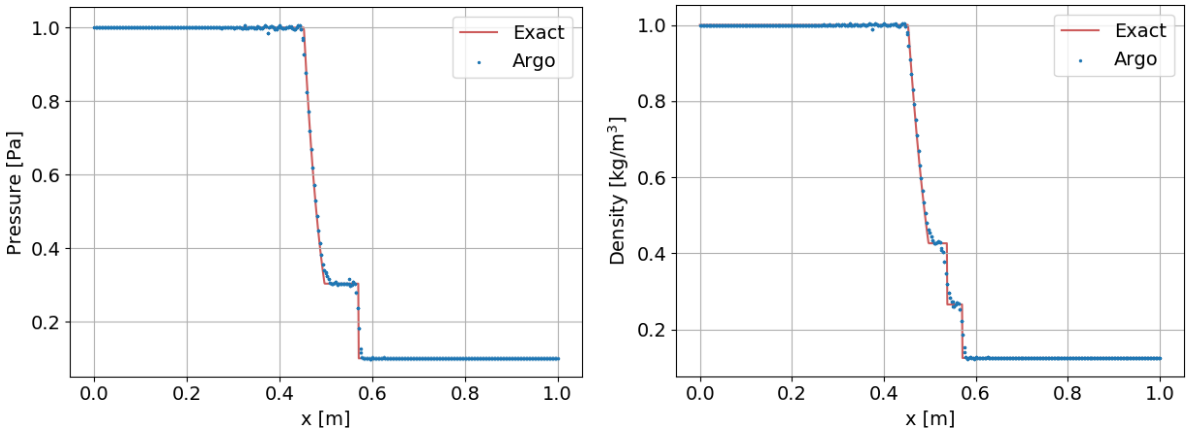


(d) Density, SLAU

Figure 4.9: Solution to the Sod test case at $p = 4$ for different Riemann solvers, with 500 DOF, $k = 0.05$, $s_0 = -3.0$ and $\kappa = 1.0$

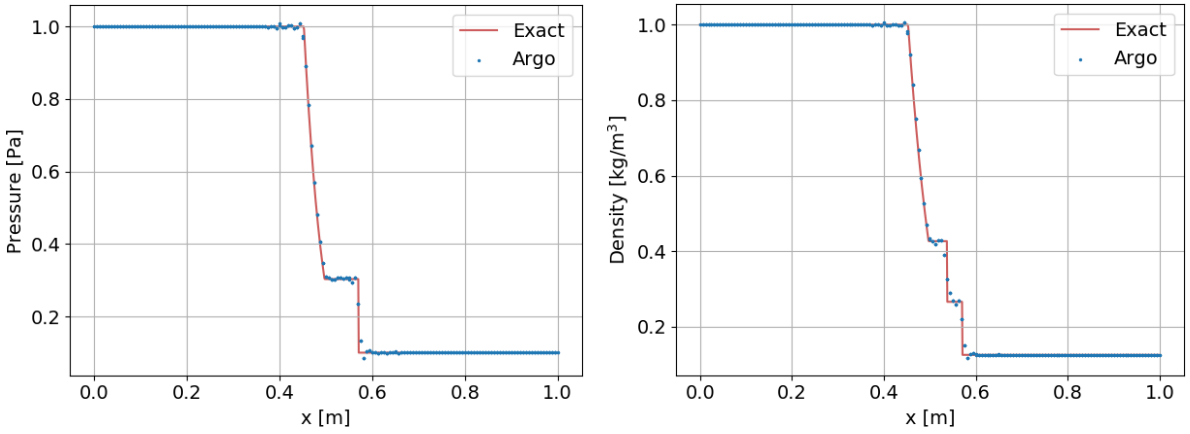
Refining in both h and p is attempted to improve the results. First, the mesh refinement is doubled so that the mesh contains 40 elements. The solution is shown in Figures 4.10(a) and 4.10(b) for the pressure and density, respectively. A significant improvement can be seen.

Increasing the order to $p = 7$ results in less spurious oscillations and discontinuities being better captured with respect to the $p = 4$ solution on 20 elements. This is particularly true for the pressure, shown in Figure 4.11(a), where the shock is well captured. For the density, shown in Figure 4.11(b), capturing the contact discontinuity remains more challenging. Compared to the solution on 40 elements, a similar accuracy is obtained, with stronger spurious oscillations but affecting a smaller region.



(a) Pressure (b) Density

Figure 4.10: Solution to the Sod test case at $p = 4$ with 1000 DOF, SLAU, $k = 0.05$, $s_0 = -3.0$, and $\kappa = 1.0$



(a) Pressure (b) Density

Figure 4.11: Solution to the Sod test case at $p = 7$ with 1280 DOF, Lax-Friedrichs, $k = 0.05$, $s_0 = -3.0$, and $\kappa = 1.0$

4.1.5. Results on an unstructured mesh

Even though the Sod problem is one-dimensional, in this work it is essentially implemented in 2D but with only 1 element in the y coordinate. This means that the shock is always aligned with the mesh. However, misalignments between shock and mesh elements make shock capturing more difficult. Therefore, the performance of the shock capturing is tested very briefly here on an unstructured mesh to visualise the impact of element orientation.

Figure 4.12 shows the result at different time steps. It is clear that depending on the location of the shock with respect to the triangular elements, the method's accuracy deteriorates. As the shock is on one side of triangular elements, peaks appear at the opposite side. Local peaks at one corner of a triangular element can also happen in different element orientations. This feature was already noticed by Atkins & Pampell in [3] and the simulation can become unstable in more extreme cases.

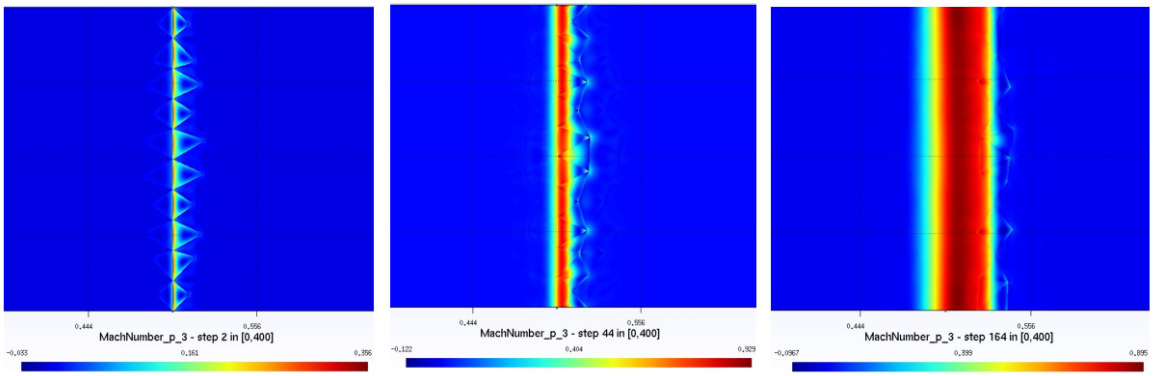


Figure 4.12: Solution to the Sod test case for the Mach number at $p = 3$ for different time steps, on an unstructured mesh, with Lax-Friedrichs, $k = 0.05$, $s_0 = -4.0$, and $\kappa = 1.0$

4.1.6. Summary

It was generally possible to obtain good results for the Sod test case. The contact discontinuity appeared to be more difficult to capture than the shock, but both could be resolved provided sufficient refinement in p and h was used. It was seen that tuning artificial viscosity is not necessarily trivial and that AV parameters may need to be adjusted across different problems, even if set-up parameters such as the order remain the same.

4.2. The Shu-Osher problem

This test is another one-dimensional problem on $x \in [0,1]$ for the Euler equations, from Shu & Osher [79]. The aim is to assess how the shock capturing scheme performs when the smooth region contains some flow variations. In this problem, a density wave pattern goes through a Mach 3 shock. This test case highlights the benefits of high-order methods because much better accuracy can be obtained in the high-frequency oscillations behind the shock without having to refine the mesh [71]. This work uses the problem implementation from [57], with initial conditions

$$\begin{cases} \rho = 3.857143,\ u = 2.629369,\ P = 10.33333, & \text{for } x < 0.125 \\ \rho = 1 + 0.2\sin(50x),\ u = 0,\ P = 1, & \text{for } x \geqslant 0.125. \end{cases} \tag{4.2}$$

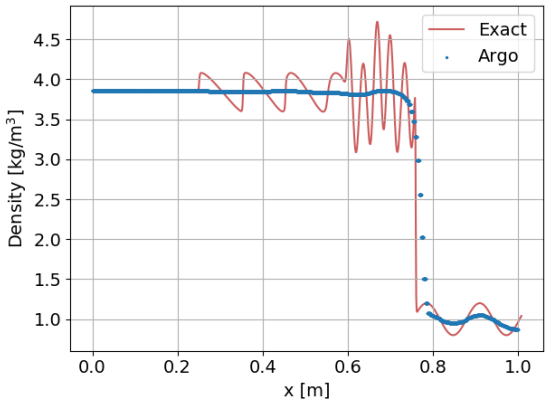
The test is run until $t = 0.18$. The problem reuses the 200 elements mesh from the Sod test case, however the boundary conditions need to be modified. The top, bottom and right boundaries are specified an adiabatic slip wall condition just as in the Sod problem. On the left, a freestream boundary condition is imposed to avoid reflection issues. Also, the Shu-Osher test case is more challenging and required to lower the time step to $\Delta t = 10^{-6}$ for stability. This affected the computational time, thus a Jacobi algorithm was used to solve the linear system instead of GRMES. The Riemann solver was Lax-Friedrichs for stability, the BDF2 scheme was used to march in time, and the AV parameters had to be changed across different simulations. The results are compared to an "exact" solution, computed with a MATLAB code running a 4th order WENO method, and a very fine mesh with $\Delta x = 0.001$.

Simulations of the Shu-Osher problem at $p = 1$ or higher were unstable unless artificial viscosity is added. This is in stark contrast to the Sod problem for which simulations at $p = 7$ could be carried out without artificial viscosity. The first results investigated were at $p = 0$, an order at which no artificial viscosity is added but the simulations are stable. Results at higher orders are analysed after.

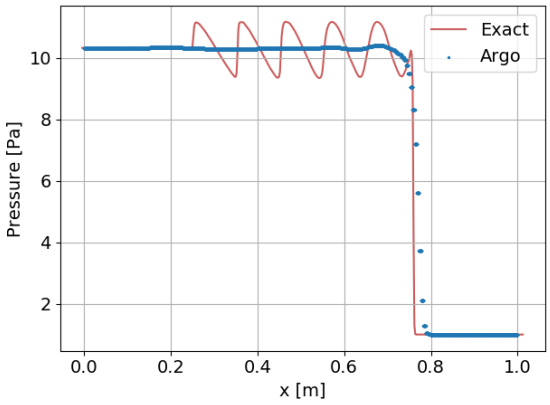
4.2.1. Results at order p=0 for different Riemann solvers

For the Shu-Osher at order $p = 0$, the effect of the chosen inviscid flux is significant. Figures 4.13(a) and 4.13(b) clearly illustrate how Lax-Friedrichs is too dissipative. The pressure and density oscillations from the exact solution are strongly flattened to the point that the solution is almost constant behind the shock and the density sine wave in front of the shock is significantly smoothed as well. Both AUSM$^+$up and SLAU appear to be much better choices than Lax-Friedrichs, with AUSM$^+$up the best of the three. The density and pressure results obtained with AUSM$^+$up are given in Figures 4.13(c) and 4.13(d), whereas Figures 4.13(e) and 4.13(f)

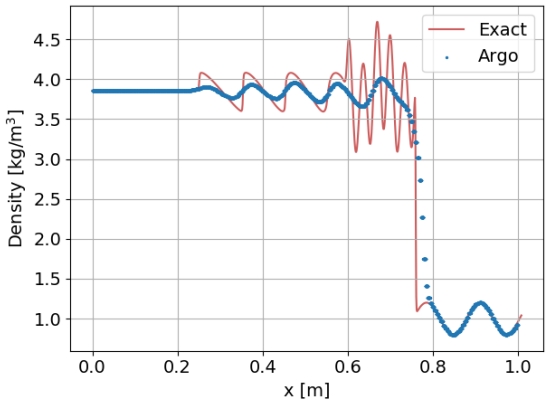
were obtained with SLAU. It is clear that at $p = 0$, the mesh refinement is simply insufficient to reproduce the physical oscillations after the shock, in particular the high frequency ones just after the shock for the density.



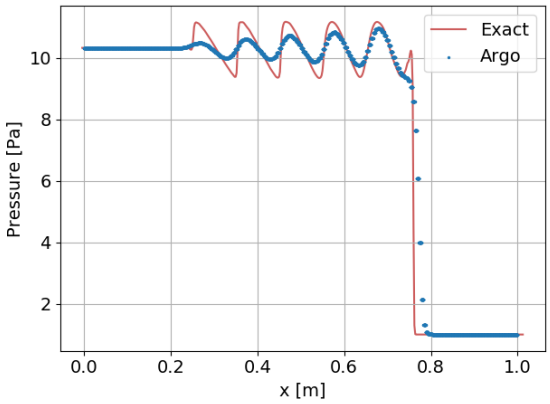
(a) Density, Lax-Friedrichs



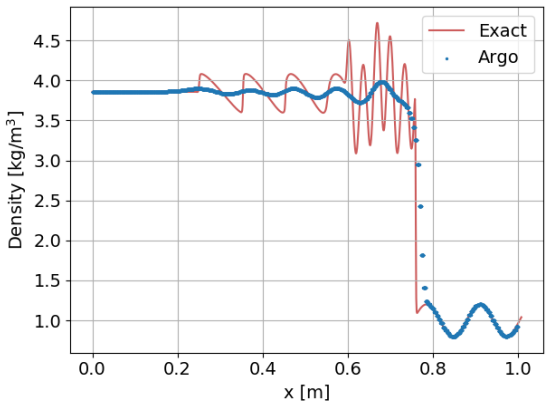
(b) Pressure, Lax-Friedrichs



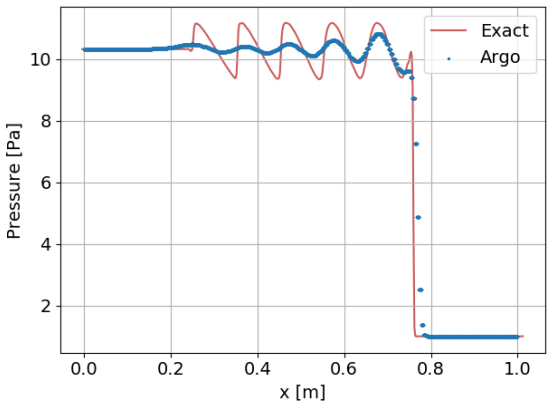
(c) Density, AUSM$^+$up



(d) Pressure, AUSM$^+$up



(e) Density, SLAU



(f) Pressure, SLAU

Figure 4.13: Solution to the Shu-Osher test case at $p = 0$ for different Riemann solvers, 200 DOF

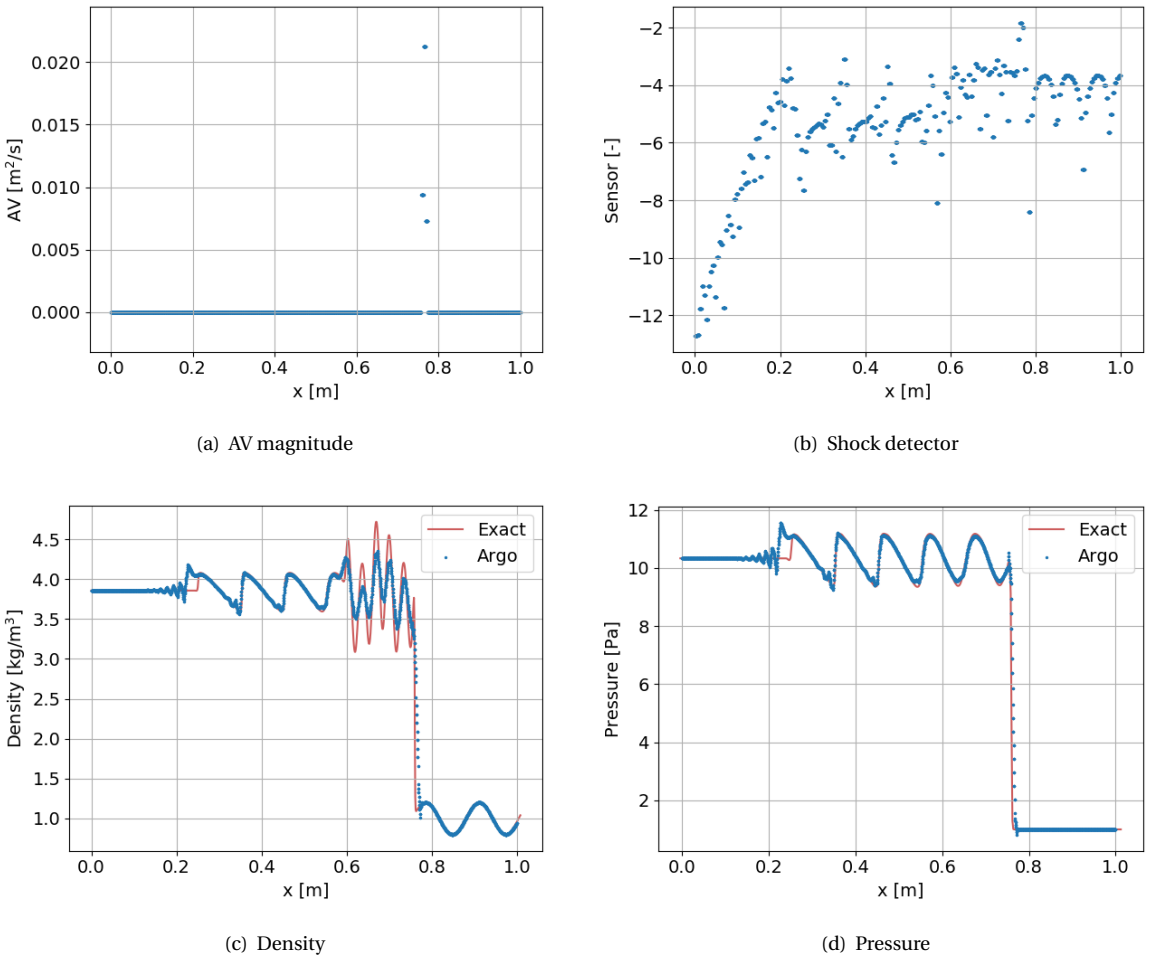
Next, high-orders are used to simulate the Shu-Osher problem. It was particularly difficult to find an appropriate set of AV parameters. The margin between insufficient and excessive artificial viscosity was narrow, and the varying shock strength made setting the AV parameters challenging. In the same simulation, the sensor value at the shock can oscillate between -1.3 and -3. Since the AV parameters are set before the simulation

is started, the real difficulty was in finding values that can account for this variation. Smoothing the AV field was found to make this task even more challenging and was therefore not applied for this test case.

4.2.2. Results at higher orders

As mentioned before, the Shu-Osher test case should highlight the advantage of increasing the order of the method. Only the Lax-Friedrichs flux could be used beyond $p = 0$, the other two leading to numerical instabilities. Nevertheless, increasing the order quickly improved the results. The AV parameters were $k = 0.01$, $s_0 = -2.0$, and $\kappa = 1.0$. At $p = 1$, Figures 4.14(a) and 4.14(b) show that the artificial viscosity is only added at the shock location, and how much the sensor values vary up and down in the domain.

Figures 4.14(c) and 4.14(d) show that the low-frequency oscillations were properly reproduced in both the density and pressure. For the density, the high frequency oscillations are accurately resolved in terms of frequency but not in magnitude. Another important feature to discuss is the spurious oscillation around $x = 0.2$ in both the pressure and density solution. It first appears at the very start of the simulation, when the shock is at $x = 0.125$. Then, the artificial viscosity follows the shock as it moves to the right. The oscillation is therefore not dissipated and is convected to the right, which explains the small spurious oscillations appearing at the two local minima either side of $x = 0.4$. This issue is not unique to this work. Vuik & Ryan [87] observed similar oscillations for different shock detectors on a finer mesh (512 elements), specifically for those which did not flag elements in this region. They appeared in Lv et al. [87] as well while using Persson and Peraire's sensor on the same mesh refinement.



(a) AV magnitude

(b) Shock detector

(c) Density

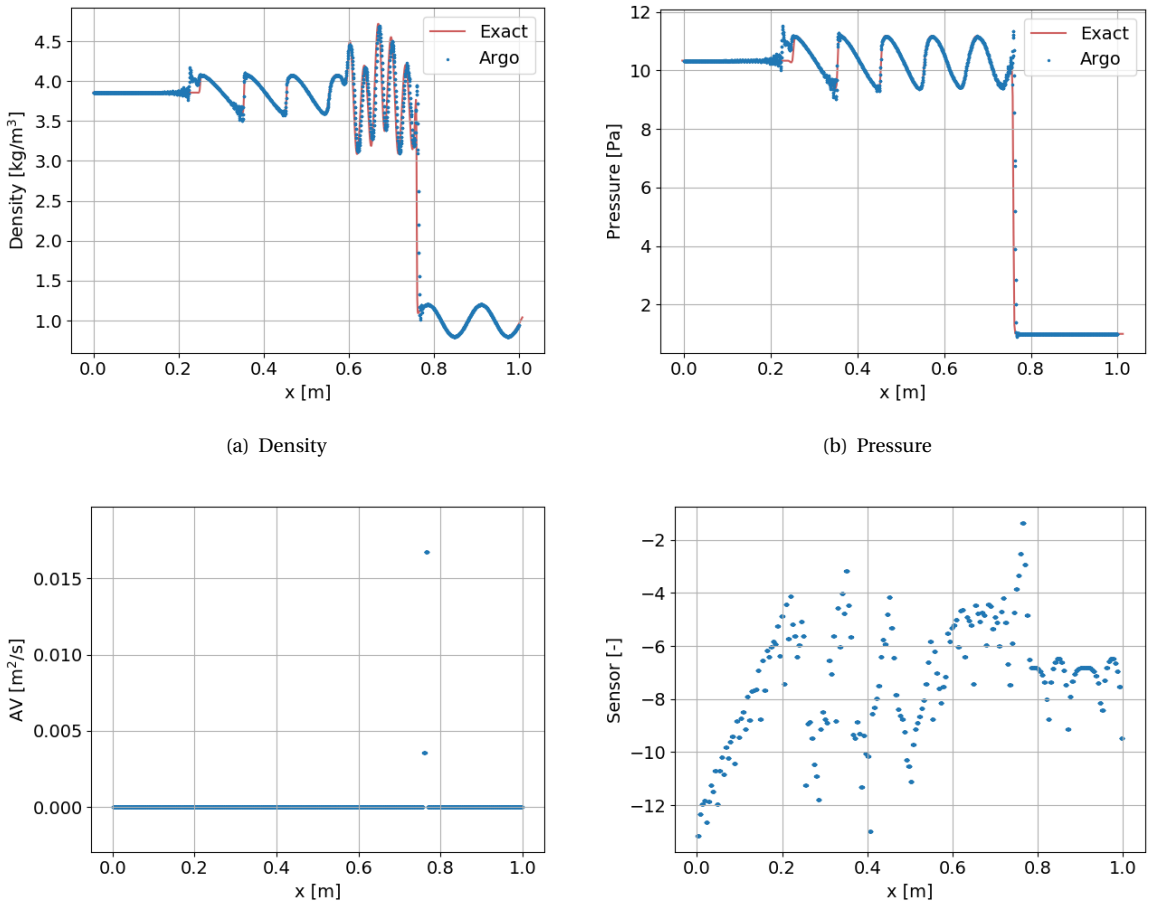
(d) Pressure

Figure 4.14: Solution to the Shu-Osher test case at $p = 1$, with 800 DOF, Lax-Friedrichs, $k = 0.01$, $s_0 = -2.0$, and $\kappa = 1.0$

Increasing the order one step to $p = 2$ is sufficient to resolve both the low and high frequency oscillations of the solution. Spurious oscillations aside, the results match closely the exact solution for both the density

(Figure 4.15(a)) and the pressure (Figure 4.15(b)). The spurious oscillations on the left side of the solution are unfortunately still present. In addition, an overshoot now appears at the shock location. Compared to results in the literature, the agreement between the Argo and exact solution is better but suffers more from spurious oscillations as well. In Lv et al. [57], the $p = 2$ solution on the same mesh resembles more the $p = 1$ solution obtained here. The same conclusion can be drawn from the results of Qiu & Shu [71], stabilised with a WENO method. It is difficult to interpret the results from Vuik & Ryan [87] since they are plotted without the exact solution.

At $p = 2$, the sensor returns lower values overall, as can be seen in Figure 4.15(d). The trend is the same than in Figure 4.14(b) though. Figure 4.15(c) shows that artificial viscosity is applied on an even narrower band. The larger difference between sensor values at the shock and in the rest of the domain means that there is potentially more margin to decrease the threshold and increase the band in which AV is added.



(a) Density

(b) Pressure

(c) AV magnitude

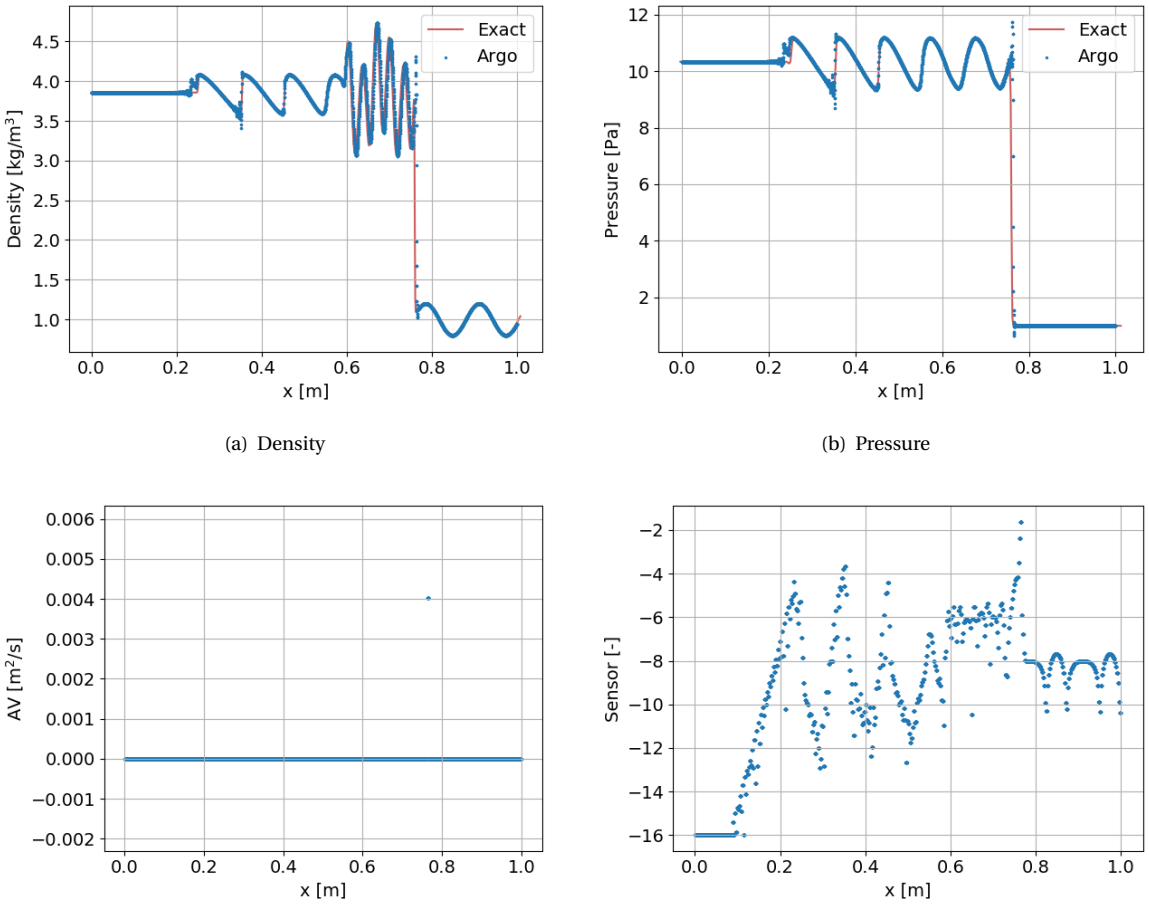
(d) Shock detector

Figure 4.15: Solution to the Shu-Osher test case at $p = 2$, with 1800 DOF, Lax-Friedrichs, $k = 0.01$, $s_0 = -2.0$, and $\kappa = 1.0$

4.2.3. Results on refined meshes

Trying to increase the order further or use different AV parameters and Riemann solvers to remove the spurious oscillations was unsuccessful. Therefore, the logical step to improve results was to increase the mesh refinement. The first refinement investigated consists of 400 elements, thus doubling the number from the baseline mesh. The AV parameters were changed to $k = 0.04$, $s_0 = -1.0$ and $\kappa = 1.0$, thus increasing both the magnitude and threshold, and smoothing was again not applied. The results are shown in Figures 4.16(a) and 4.16(b). An improvement can be observed in the high frequency oscillations of the density, with the solution obtained with Argo now matching very closely the exact solution. In addition, the spurious oscillation close to $x = 0.2$ has reduced. However, the overshoot at the shock has increased, as well as the oscillation at around $x = 0.35$. Lv et al. [57] also show results on a 400 elements mesh, and again their results are smoother but match the exact solution less closely in particular in the high-frequency oscillations.

Refining the mesh in h affects the shock detector in the same direction as refining in p. As can be seen in Figure 4.16(d), overall the values are lowered compared to Figure 4.14(b). With the AV parameters chosen, only one element is flagged as troubled. This is clear in Figure 4.16(c), which shows only one non-zero element. Apart from artificial viscosity being applied to less elements, the amount itself also lowers with decreasing element size, as expected.



(a) Density

(b) Pressure

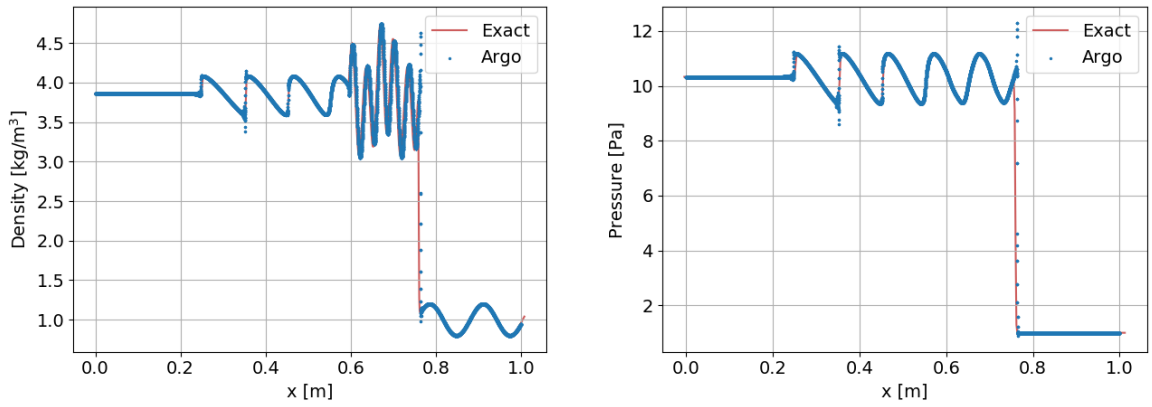
(c) AV magnitude

(d) Shock detector

Figure 4.16: Solution to the Shu-Osher test case at $p = 2$, with 3600 DOF, Lax-Friedrichs, $k = 0.04$, $s_0 = -1.0$, and $\kappa = 1.0$

The second refinement aims to test the problem on a very fine mesh, which includes 1000 elements. The AV magnitude was slightly changed to $k = 0.05$, with the threshold and interval remaining at $s_0 = -1.0$ and $\kappa = 1.0$, respectively. The results are shown in Figures 4.17(a) and 4.17(b) and are consistent with the features observed in the previous mesh refinement. Spurious oscillations aside, the match between the Argo and exact solution is very good. It can be seen how the density and pressure are better resolved at the first

oscillation near $x = 0.2$. In terms of overshoots however, the issues observed for 400 elements have now been accentuated.



(a) Density

(b) Pressure

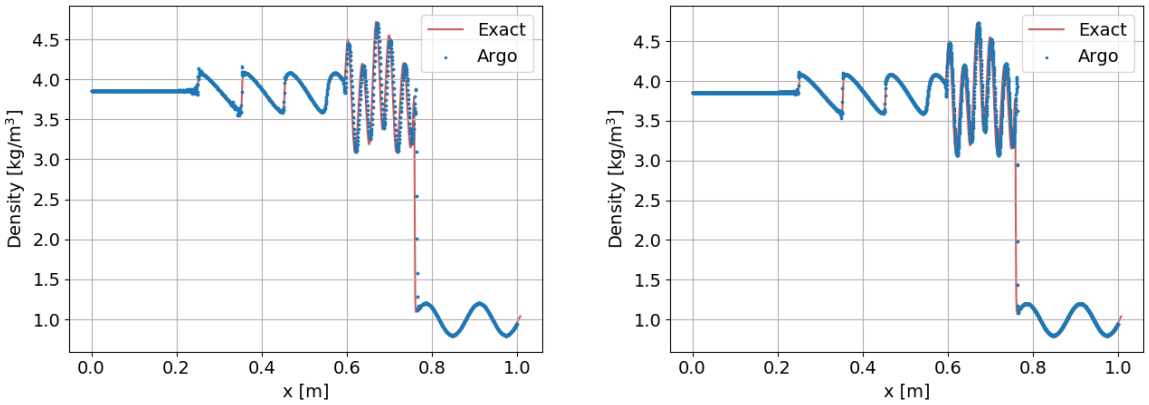
Figure 4.17: Solution to the Shu-Osher test case at $p = 2$, with 9000 DOF, Lax-Friedrichs, $k = 0.05$, $s_0 = -1.0$, and $\kappa = 1.0$

4.2.4. Results obtained with DGFluid

A general difficulty in simulating the Shu-Osher test case is that the initial condition for variables such as the density and pressure are relatively close to zero on the right side of the shock. When spurious oscillations appear around the shock, it is common for such variables to become negative and crash the simulation. To avoid this issue, it is beneficial to clip non-physical values so that variables remain within physical bounds. Rather than implementing this clipping in DGAblation, it was chosen to use modules from DGFluid instead to benefit from the clipping implemented there. For simplicity, using these modules from DGFluid will simply be called using DGFluid. It is worthwhile to remind the reader that using DGFluid means that conservative variables are used instead of the primitive set used up to now.

It became quickly clear that using DGFluid was beneficial for stability. To ensure that this increase in stability was due to the clipping and not to the change of variables, a simulation was run with the clipping deactivated. It crashed at the same point as its equivalent simulation with DGAblation. Switching the clipping back on removed the stability issue and the simulation could run until the end.

With the clipping activated, it then became possible to simulate the Shu-Osher problem at higher orders, however still being stuck on the Lax-Friedrichs flux. The solution at $p = 4$ is shown in Figures 4.18(a) and 4.18(b) on two different meshes for the density only, since the pressure was not found to provide further information. The results for the baseline mesh at 200 elements used $k = 0.07$, $s_0 = -3.0$ and $\kappa = 1$ as AV parameters. On the finer mesh with 400 elements, the parameters were $k = 0.02$, $s_0 = -2.5$ and $\kappa = 2$. Overall, the greater stability resulted in more freedom to choose values. Nevertheless, it was chosen not to smooth the AV field to limit the number of changes with respect to previous simulations. The accuracy of the solution is generally very similar to what was seen at $p = 2$, however now with less spurious oscillations. The overshoots in particular are less severe on these $p = 4$ results, but worsening with mesh refinement as was observed in the $p = 2$ case.



(a) Density, 5000 DOF, $k = 0.07$, $s_0 = -3.0$, and $\kappa = 1.0$

(b) Density, 10000 DOF, $k = 0.02$, $s_0 = -2.5$, and $\kappa = 2.0$

Figure 4.18: Solution to the Shu-Osher test case at $p = 4$, with Lax-Friedrichs

4.2.5. Summary

The main findings of the analysis performed on the Shu-Osher test case are reviewed here. The wedge and Sod test cases had already shown that finding an optimum set of artificial viscosity parameters is not necessarily trivial. The Shu-Osher test case highlighted that when the sensor values corresponding to the shock vary in time, tuning the parameters becomes even more difficult. The parameters were generally set such that the minimum amount of AV necessary is applied, by setting the threshold relatively high. The Riemann solver choice was shown to impact the results significantly at $p = 0$, however could not be assessed at larger orders of magnitude where it is expected to become less important. Furthermore, it was shown that increasing the order is very effective in better resolving physical high frequency oscillations. Finally, the need to clip non-physical values became clear. Indeed, it can be very difficult or even impossible to avoid variables such as the density or pressure to become negative only with artificial viscosity. DGFluid will thus be used in the rest of this work, which will be focused on the final test case and end goal of this thesis.

5

Application to a viscous, steady half-cylinder test case

The 2D half-cylinder test case is a classic problem for the assessment of hypersonic flows around a blunt body. This chapter applies the knowledge gathered in the previous test cases in order to predict the wall heat flux. Indeed, this problem sees strong variations in the transient, a normal shock on the stagnation line and oblique shocks of varying inclination elsewhere, and difficulties caused by the unstructured mesh. In addition, the problem now contains inherent viscosity and thus a boundary layer around the cylinder wall, which proved to be a significant aspect to consider for the shock capturing method. The Mach number had to be limited at 3.25 however, moving directly to hypersonic flows proving too big of a step and resulting in numerically unstable simulations. Nevertheless, obtaining the heat flux at this relatively low Mach number proved challenging and this chapter explains the steps that were taken to achieve it.

5.1. Problem set-up

A semi-circular domain is considered for the half-cylinder test case, as shown in Figure 5.1. The problem data is taken from Knight et al. [42], with the exception of a reduced Mach number and a higher wall temperature to reduce the difficulties in obtaining a solution. The boundary conditions are given in Table 5.1. Note that the flow is supersonic at the Outlet, except for a very small region near the Wall where the flow is subsonic and the static pressure is imposed. The value was chosen to be equal to the freestream, which is not a particularly good choice but this will be discussed in Section 5.2.1. Preliminary results with a few different meshes showed that the one in Figure 5.2 would be best to start on this test case. It contains a local refinement around the shock and the boundary layer. In the latter region, structured cells are used, with a height at the wall of 10^{-4}. Overall, the mesh contains about 7000 elements, with a mix of mainly quadrilaterals and some tetrahedra.

Table 5.1: Boundary conditions for the half-cylinder test case. NA: Not Applicable.

Parameter	Freestream	Wall	Outlet
Velocity [m/s]	1956	No-slip	NA
Static pressure [Pa]	476	NA	NA
Static temperature [K]	901	1000	NA

It was chosen to perform steady simulations due to the nature of the solution, unless mentioned otherwise. Generally, GMRES was chosen to solve the linear system with Jacobi or ILU preconditioning. The Lax-Friedrichs scheme was used for all simulations in this chapter, as well as the smoothing step for the artificial viscosity field. All simulations are at $p = 1$ and had to be started from an homogeneous initial condition since the diffusive term of the Interior Penalty method is inconsistent at $p = 0$.

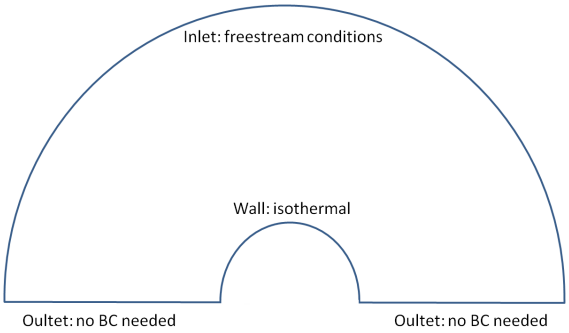


Figure 5.1: Problem set-up for the half-cylinder test case

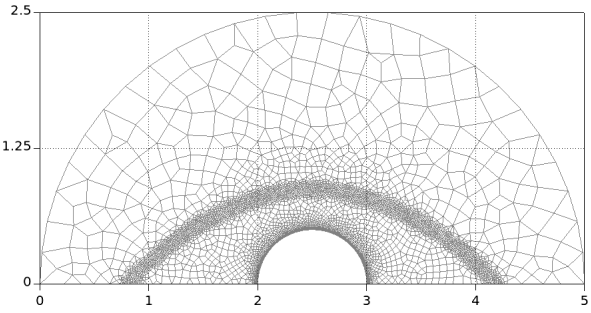


Figure 5.2: Coarse mesh with local refinements around the shock and boundary layer

5.2. Results at Mach number 3.25

A number of changes were implemented to achieve a satisfying heat flux at $M = 3.25$. This section explains the reasoning followed and the steps that were taken.

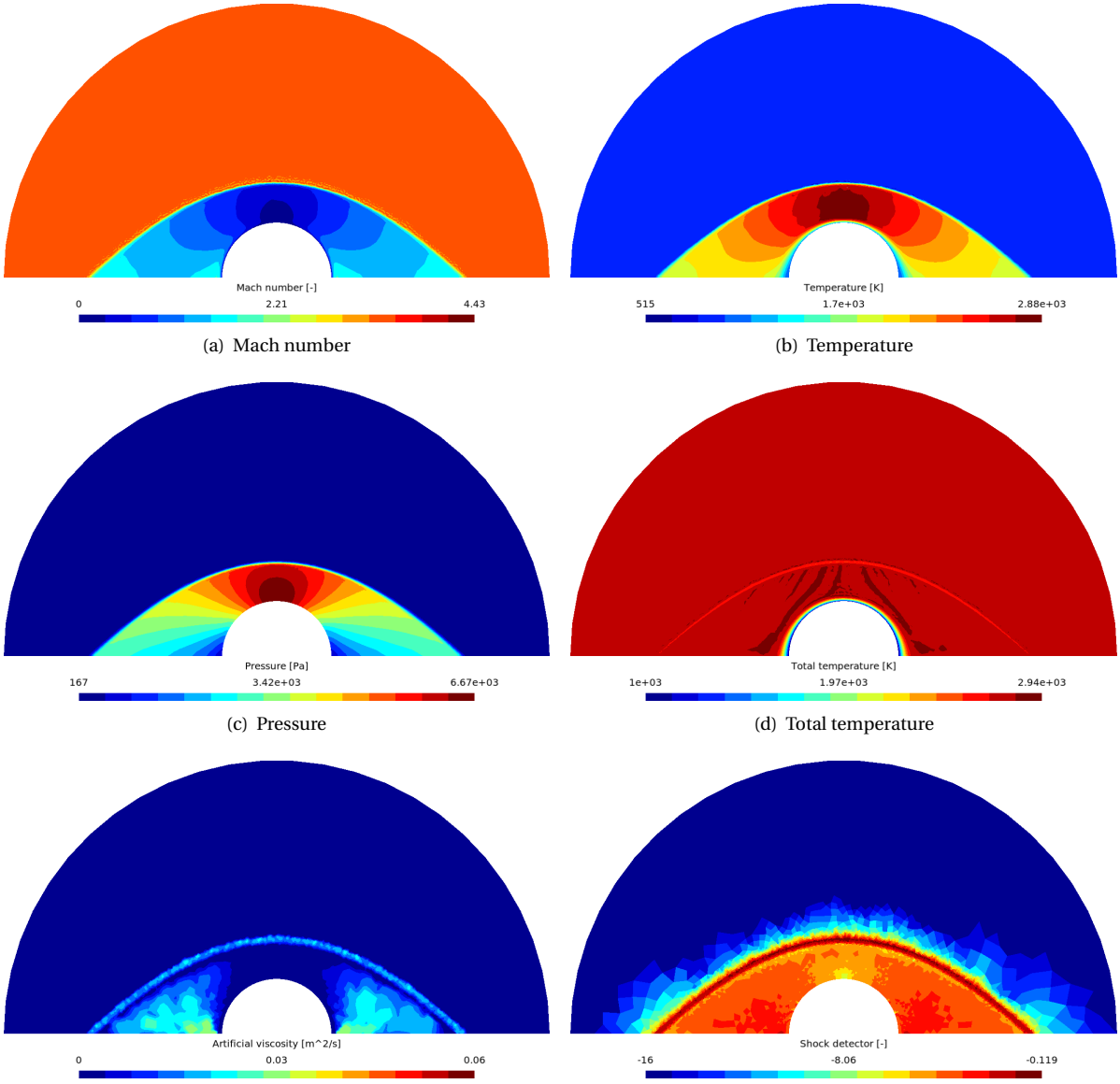
A first solution was obtained to be used as baseline, with artificial viscosity parameters $k = 0.1$, $s_0 = -2.5$ and $\kappa = 2.0$. The results are shown in Figures 5.3(a) to 5.3(f). In Figure 5.3(a), the Mach number distribution is shown. The expected pattern is seen, where the flow sees a much stronger shock in the middle and, as a result, part of the flow behind the shock is subsonic. A region of low Mach number is also present along the cylinder wall. Although the solution appears to be smooth and the shock sharply captured, oscillations are present which can peak locally up to $M = 4.43$. A careful look at the solution also reveals a few asymmetries, which are in fact easier to see in Figure 5.3(b). The temperature field highlights a rather large boundary layer. Considering that boundary layer thickness is supposed to increase with increasing Mach number, and that results in [14, 15, 42] show a thinner boundary layer for higher Mach numbers, it is clear that an excessive thickness has been obtained. The pressure distribution is shown in Figure 5.3(c), and is affected by dips in pressure to 167Pa while the theoretical minimum in the domain should be 476Pa. Figure 5.3(d) shows the total temperature. As expected, the total temperature drops in the boundary layer. What is less clear is why does the total temperature drop and recover in the shock. One could imagine that this is caused by the application of artificial viscosity in the shock. However, Figure 5.3(e) shows that substantial amounts of artificial viscosity are added in the post-shock region, on both sides of the cylinder. This suggests that there should be clear variations in total temperature in much of the post-shock region if that hypothesis was true. Instead, the drop and recovery can be better explained by looking at the formula used to compute it, which is given by

$$T_0 = T + \frac{\mathbf{u}^2}{2c_p}. \tag{5.1}$$

What happens in the shock is that T increases and $\frac{\mathbf{u}^2}{2c_p}$ decreases. However, they do so at different rates such that inside the shock the second term dominates first and the temperature after. Note that these rates are not expected to be fully accurate due to the artificial viscosity and possibly due to the set of conservative variables used as well (see Section 2.2.1 for the note on variables). In addition, the total temperature in the post-shock region is close but not identical to that upstream of the shock. Due to spurious oscillations, the field is not entirely homogeneous downstream of the shock.

The results related to artificial viscosity specifically are shown in Figures 5.3(e) and 5.3(f). The former shows that artificial viscosity is added in a thin layer around the shock. There is also a large region in the post-shock area where artificial viscosity is applied. Looking at the sensor values, a peak around the shock is seen as expected. However, large values remain in the post-shock region. This is rather unwanted since there are no shocks to be dissipated there.

The heat flux at the wall is shown in Figure 5.4(a). Note that the angle is zero at the stagnation point and $\pm\pi/2$ at the bottom right/left corners. Since the Mach number had to be reduced, the result cannot be compared to those shown in Knight et al. [42]. Nevertheless, a qualitative analysis can be made. It is well known that the result should resemble a Gauss-type bell curve, thus the overall trend of the result is roughly correct. However, the oscillations around the angle -1 and 1 should clearly not be there. The reason why they appear will be explained later. Furthermore, the local minima at the stagnation point is unexpected. In theory, that is where the heat flux should peak. To support this qualitative analysis, it is also possible to estimate the heat flux q at the stagnation point by means of an empirical expression given by



(a) Mach number

(b) Temperature

(c) Pressure

(d) Total temperature

(e) Artificial viscosity

(f) Shock detector

Figure 5.3: Solution to the half-cylinder test case with density as variable for the shock detector, with $p = 1$, Lax-Friedrichs, $k = 0.1$, $s_0 = -2.5$ and $\kappa = 2.0$

$$q = 0.57 Pr^{-0.6} (\rho_{ed} \mu_{ed})^{1/2} \sqrt{\frac{du_{ed}}{dx}} (h_{aw} - h_w) \tag{5.2}$$

where Pr is the Prandtl number and is 0.71 for air, the subscript ed denotes the value of the variable at the edge of the boundary layer and is here assumed to be equal to its post-shock value, h_{aw} is the enthalpy for an adiabatic wall and h_w is the enthalpy at the wall [38]. The post-shock values are estimated using normal shock relations:

$$\frac{\rho_{ed}}{\rho_1} \approx \frac{(\gamma+1)M_1^2}{2+(\gamma-1)M_1^2} \tag{5.3}$$

$$\frac{P_{ed}}{P_1} \approx 1 + \frac{2\gamma}{\gamma+1}(M_1^2 - 1) \tag{5.4}$$

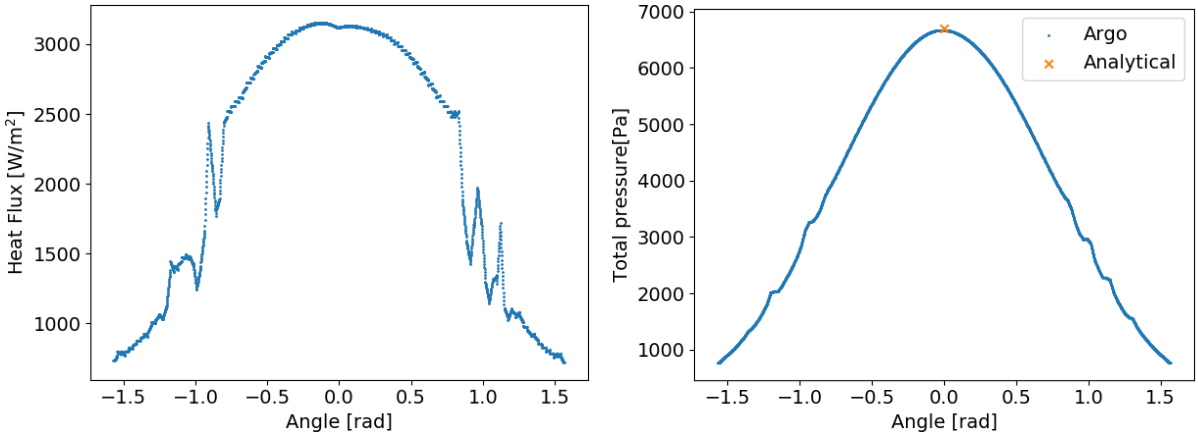
Note that this step is not entirely correct since the formulas are derived for inviscid flow, hence the approximation rather than equality sign. In theory, the recovery factor r is used to compute the adiabatic wall enthalpy through $h_{aw} = h_{ed} + r\frac{u_{ed}^2}{2}$. In this case, the assumption $h_{aw} \approx h_0$ can be made without any important loss of accuracy, with $h_0 = c_p T_0$ being the total enthalpy outside the boundary layer [38]. The total temperature is found using the isentropic relation

$$T_0 = T\left(1 + \frac{\gamma - 1}{2} M^2\right). \tag{5.5}$$

The velocity gradient at the edge of the boundary layer is approximated with

$$\frac{du_{ed}}{dx} = \frac{1}{R}\sqrt{\frac{2(P_{ed} - P_\infty)}{\rho_{ed}}} \tag{5.6}$$

where R is the radius of the body and the infinity subscript refers to freestream conditions [38]. It is worth noting that the latter equation is derived from Newtonian theory, which works better at large Mach number and for 3D bodies [37]. It is argued that the conditions of the test case at hand are at the limit of validity of the empirical expression. The results should thus be treated with caution. According to Equation (5.2), the heat flux at the stagnation point should be in the order of $10^4 W/m^2$. To gain confidence on this estimate, it was verified that this formula gives an order of magnitude consistent with the results from Knight et al. [42], which was the case. Therefore, the heat flux result shown in Figure 5.4(a) is most likely about an order of magnitude too low. There are many reasons which could explain this difference, including insufficient refinement in h or p, the use of the Lax-Friedrichs flux, or even the small amount of artificial viscosity added in the boundary layer. A minor dip at the stagnation point such as that in Figure 5.4(a) was also observed by Burgess & Mavriplis [13].



(a) Heat flux

(b) Total pressure

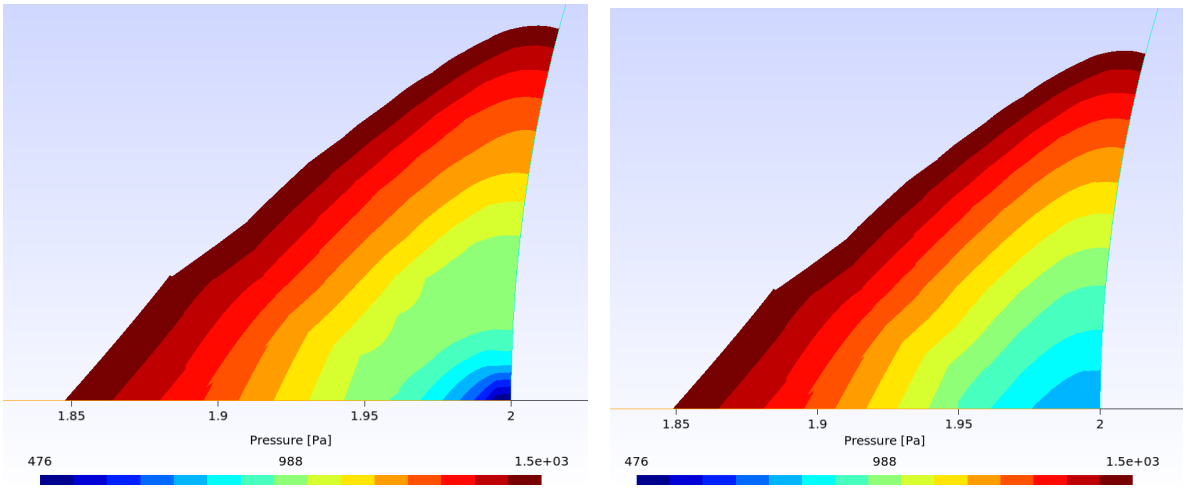
Figure 5.4: Solution to the half-cylinder test case on the wall, with with $p = 1$, Lax-Friedrichs, $k = 0.1$, $s_0 = -2.5$ and $\kappa = 2.0$

The total pressure at the wall is shown in Figure 5.4(b) and compared to the theoretical value at the stagnation point, computed with the normal shock relations. This analytical point should therefore not match exactly with the results. The total pressure at the wall is much smoother than the heat flux, which is expected since there is no gradient involved. Nevertheless, it is not as smooth as what is obtained in Knight et al. [42], Barter & Darmofal [5] or Ching et al. [15]. At the stagnation point, the result comes close to the analytical value, thus indicating that the pressure is at least much better estimated than the heat flux. Spurious oscillations aside, asymmetries appear in both the total pressure and heat flux results. Due to the randomness of the elements in the mesh, it was expected that symmetry issues would appear. It typically occurs in the literature as well, particularly at low orders. For example, Gnoffo & White [28] and Nompelis et al. [64] see this behaviour in FVM. In DG, Barter & Darmofal [5] obtained subtle asymmetries at $p = 3$ and in Ching et al. [15], asymmetries were strong at low order and coarse mesh, but reduced significantly with order and mesh refinement.

In the next sections, the changes implemented to improve the results are discussed. The first change is related to the Outlet boundary condition.

5.2.1. Changing the implementation of the Outlet boundary condition in Argo

In Argo, the implementation of the Outlet boundary condition was inadequate for this test case. Indeed, the code checked whether the flow was subsonic or supersonic and in the former case, imposed a specified outlet static pressure. The imposed value would be equal to the freestream pressure. The problem is that very close to the wall, the flow becomes subsonic and an outlet static pressure is thus imposed. This behaviour is shown in Figure 5.5(a), which zooms near the bottom left corner of the cylinder. It can be seen that very near $x = 2$, the freestream pressure is imposed at the outlet. However, it does not make sense to impose a pressure in that region. The code was thus modified such that the pressure is always interpolated in the full outlet region. The result of the adapted outlet boundary condition is shown in Figure 5.5(b), and the pressure can be seen to behave as expected.



(a) Original implementation

(b) Modified implementation

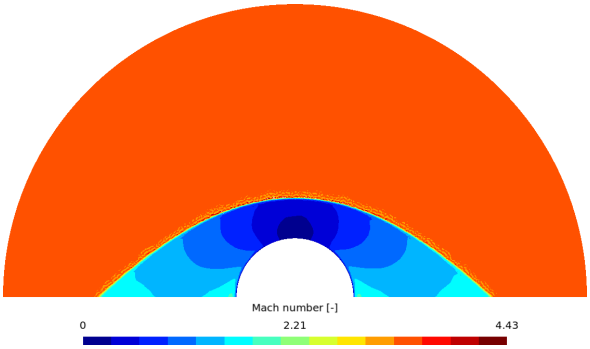
Figure 5.5: Pressure implementation of the outlet boundary condition

Away from this corner, the pressure fields in both figures look identical and the difference between them at the bottom corner would not be visible without zooming in. The impact of this modification is only local and therefore does not provide any significant improvement to the heat flux results.

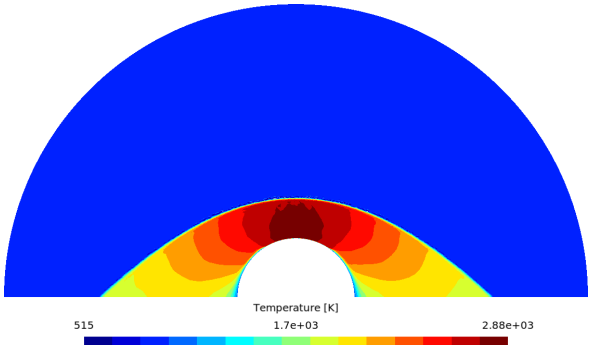
5.2.2. Changing the variable for shock detection

While analysing Figure 5.3(e), it was observed that AV is added around the cylinder. This results in a thick boundary layer, as can be seen in the temperature field (Figure 5.3(b)). More importantly, gradients in the boundary layer are steep and need to be resolved accurately in order to obtain useful heat flux predictions. Using density as variable for the shock detector appears to be a rather poor choice, since the significant variation in the boundary layer causes the shock detector values to increase. As a consequence, the elements in the boundary layer are flagged and AV is applied in that region. This thinking is supported by Ching et al. [15, p59], who reported that "the amount of AV present in boundary layers should be minimized". To avoid this issue, one could consider using a different variable with less variation near the cylinder wall for the detection of discontinuities.

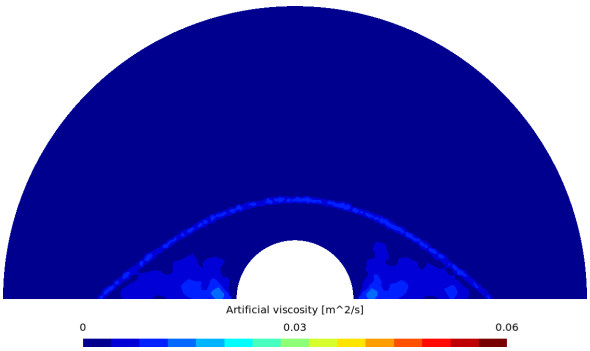
In Figure 5.3(c), it appears the pressure does not change when getting near the cylinder wall, as opposed to the Mach number or the temperature. This is further confirmed in Figure 5.5(b), which zooms in near the cylinder wall. Therefore, the pressure appears as a suitable candidate for the shock detector. Unfortunately, the results obtained with $k = 0.05$, $s_0 = -2.5$ and $\kappa = 2.0$ show that this step only partially solves the issue, arguably leading to worse results overall. Figure 5.6(a) shows the Mach number distribution while the temperature field is shown in Figure 5.6(b). The results for the Mach number are very similar to what was obtained in Figure 5.3(a). However, the temperature shows a major difference in the boundary layer. Around the front of the cylinder, the thickness of the boundary layer has significantly reduced compared to Figure 5.3(b). This can clearly be attributed to the change attempted, since the shock detector now returns lower values in that region. This can be clearly seen by comparing Figure 5.6(d) with Figure 5.3(f), in particular at the front of the cylinder. Comparing Figure 5.6(c) with Figure 5.3(e) also reveals that much less artificial viscosity is added in the boundary layer.



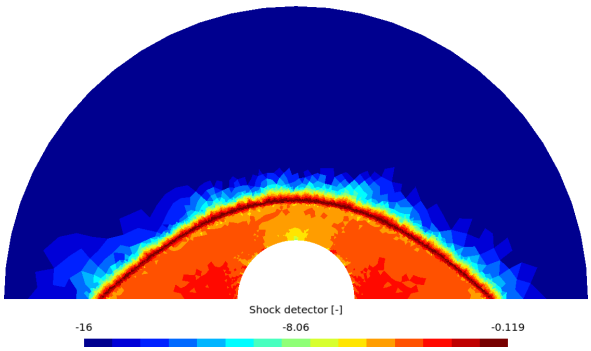
(a) Mach number



(b) Temperature



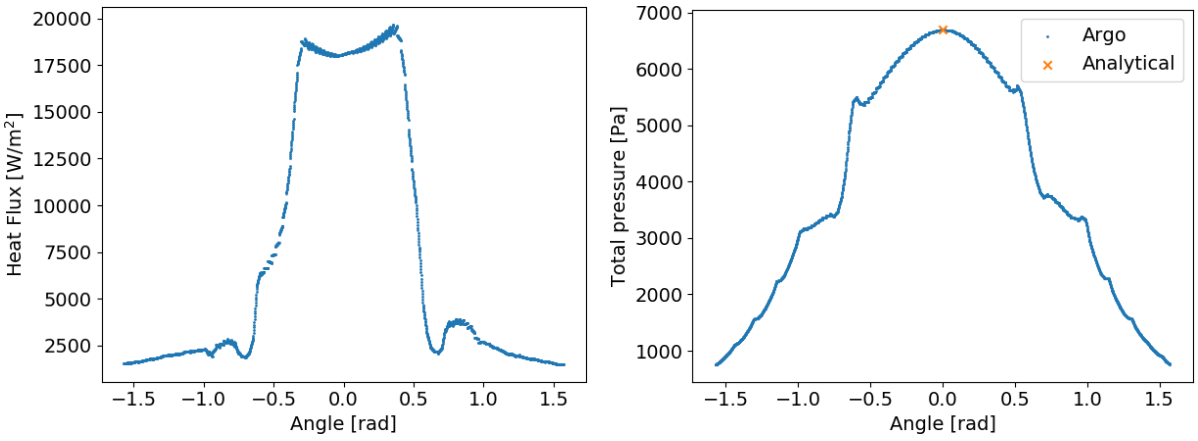
(c) Artificial viscosity



(d) Shock detector

Figure 5.6: Solution to the half-cylinder test case with pressure as variable for the shock detector, with $p = 1$, Lax-Friedrichs, $k = 0.05$, $s_0 = -2.5$ and $\kappa = 2.0$

The main issue with these results is that the boundary layer thickness suddenly grows behind the frontal region. There is no physical reason for the boundary layer to grow in such fashion, it is in fact provoked by small amounts of artificial viscosity. The asymmetry in the solution is clear in Figure 5.6(b) because the point at which the boundary layer starts growing is higher on the left than on the right. As expected, Figure 5.7(a) shows how this sudden growth is detrimental to the wall heat flux prediction. The heat flux values increase suddenly and significantly in the the small transition region where the boundary layer changes from its thinnest to thickest sizes. Even the total pressure along the cylinder wall is affected in the same way (although to a lesser extent), as can be seen in Figure 5.7(b). However, it is worthwhile noting that the heat flux's order of magnitude is now consistent with Equation (5.2). This suggests, unsurprisingly, that the artificial viscosity was severely affecting the wall heat flux accuracy.



(a) Heat flux (b) Total pressure

Figure 5.7: Solution to the half-cylinder test case on the wall with pressure as variable for the shock detector, with $p = 1$, Lax-Friedrichs, $k = 0.05$, $s_0 = -2.5$ and $\kappa = 2.0$

5.2.3. Refining the mesh

Apart from the issue of sudden thickness change just discussed, the heat flux results in Figures 5.4(a) and 5.7(a) show a dip on the stagnation point which is particularly significant in the latter figure. Furthermore, heat flux and total pressure plots shown up to now generally contain more oscillations than they should have. It would be useful to refine the mesh and assess whether these oscillations decrease, whether the thickness increase of the boundary layer would also be caused by insufficient refinement, and whether the left-right symmetry improves.

The problem was thus run on a finer mesh, shown in Figure 5.8, to assess whether the strong oscillations and asymmetry reduce. In this new mesh, a particular attention was placed in refining the mesh elements in the shock and boundary layer regions. The total number of elements increases to about 15000, with the height of elements at the cylinder wall one order of magnitude lower (10^{-5}). With this finer mesh, the artificial viscosity parameters were changed to $k = 0.15$, $s_0 = -2.5$ and $\kappa = 1.0$.

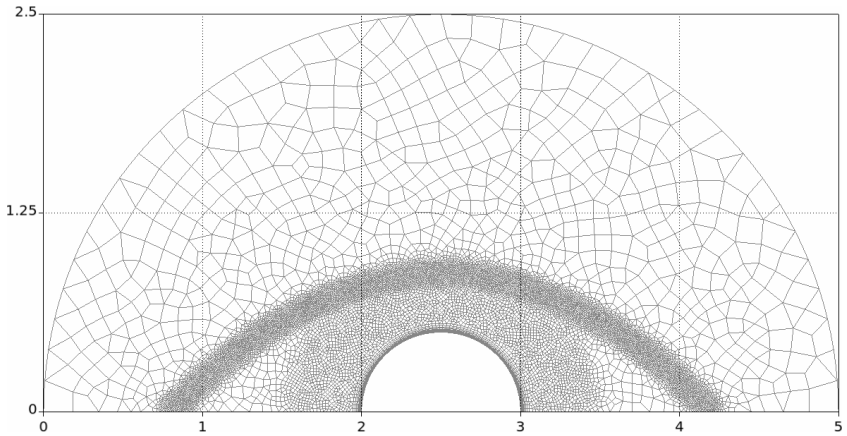
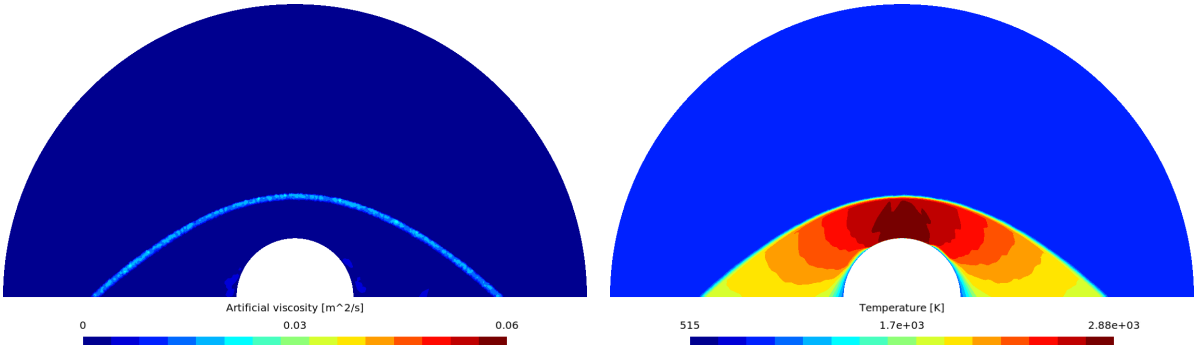


Figure 5.8: Fine unstructured mesh with local refinement around the shock location

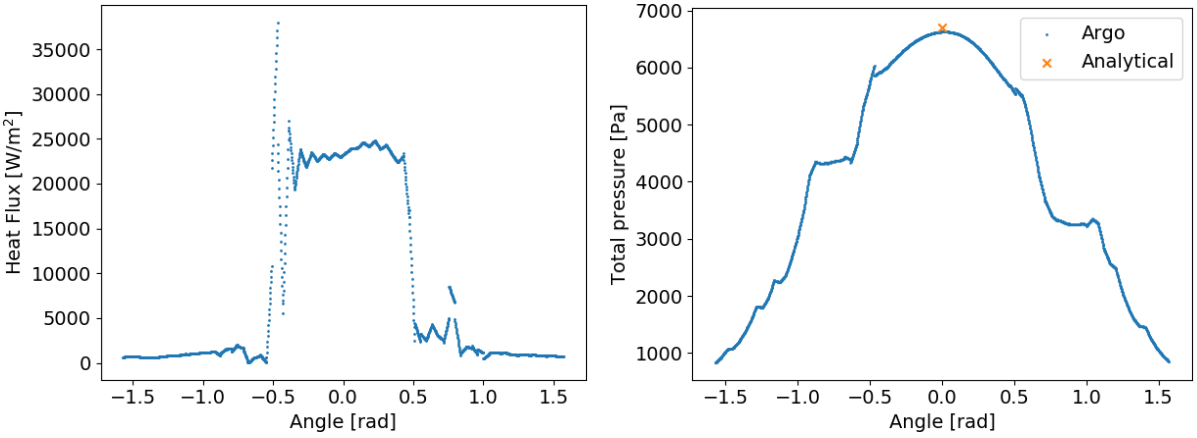
Figure 5.9(a) show the artificial viscosity field obtained with the refined mesh. There is still some AV applied in the boundary layer, however it has been largely removed from the rest of the post-shock region. Unfortunately, one can see that in the boundary layer the AV is not symmetric, thus it is expected to see asymmetries in the rest of the results as well. The temperature field, shown in Figure 5.9(b), shows indeed a strong asymmetry. Such results suggest improvements will not be obtained in the prediction of heat flux and total pressure at the cylinder surface.



(a) Artificial viscosity (b) Temperature

Figure 5.9: Solution to the half-cylinder test case on refined mesh, with $p = 1$, Lax-Friedrichs, $k = 0.15$, $s_0 = -2.5$ and $\kappa = 1.0$

Figures 5.10(a) and 5.10(b) confirm that indeed the solution at the wall has not improved. The heat flux maintains a similar magnitude at the front of the cylinder, but this is essentially the only positive comment that can be made about it. A similar conclusion can be drawn for the total pressure.



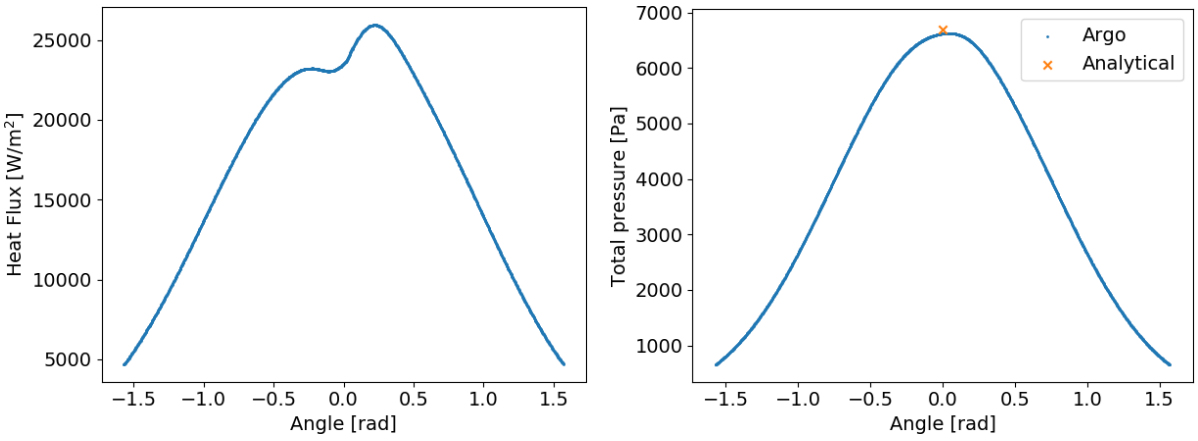
(a) Heat flux (b) Total pressure

Figure 5.10: Solution to the half-cylinder test case on the wall, with $p = 1$, Lax-Friedrichs, $k = 0.15$, $s_0 = -2.5$ and $\kappa = 1.0$

5.2.4. Removing AV entirely from the boundary layer

A recurring issue in all the results shown so far is that AV is added in the boundary layer. It seems that even small amounts of artificial viscosity are sufficient to pollute the quantities of interest in the boundary layer. Changing the variable used for shock detection was shown to not be sufficient to avoid dissipating in the boundary layer. To achieve zero AV in the boundary layer, the shock detector values were analysed in more detail. The maximum values in the shock and in the boundary layer were extracted and defined the upper and lower bounds in which AV would be applied. That is, the threshold s_0 was defined to be at the middle of the two bounds and κ was half the interval between the two bounds. Defining the AV parameters as such, the mesh elements in the boundary layer are not flagged and those in the shock receive the maximum AV magnitude specified by k. This procedure led to $k = 0.17$, $s_0 = -2.0$, $\kappa = 0.5$.

Overall, the AV footprint is smaller and the magnitude has been slightly increased to compensate. The wall heat flux and total pressure in Figures 5.11(a) and 5.11(b) show that when artificial viscosity is removed entirely from the boundary layer, smooth results are obtained. The magnitude for both the heat flux and total pressure appear to be right, thus the only area where an improvement is clearly needed is symmetry. For that, the best option is to test what happens when a symmetrical mesh is used, which is done next.



(a) Heat flux

(b) Total pressure

Figure 5.11: Solution to the half-cylinder test case on the wall with no artificial viscosity in the boundary layer, with $p = 1$, Lax-Friedrichs, $k = 0.17$, $s_0 = -2.0$, and $\kappa = 0.5$

The symmetrical mesh is shown in Figure 5.12, and although it contains a strong refinement around the cylinder wall, no particular arrangement of elements was made for the shock. It also happens to be the coarsest mesh of all used thus far, with around 3000 elements. It is worthwhile noting that Ching et al. [15] have been able to predict accurately the heat flux for a $M = 17$ case with coarser meshes.

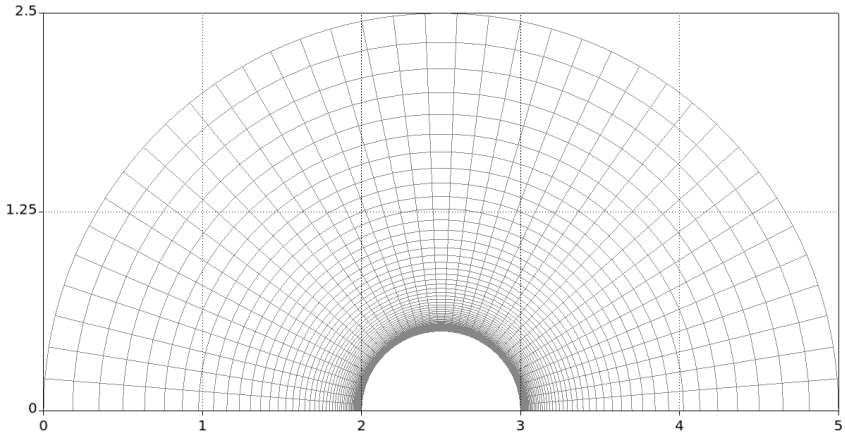


Figure 5.12: Structured, coarse mesh for the half-cylinder test case

The challenge with using such mesh was that if the AV parameters were chosen such that no AV is added in the boundary layer, the stabilisation was insufficient around the shock and the simulation crashed. Therefore, it was decided to first let the simulation run until the shock was well resolved and the convergence stalled. For that, the AV parameters used were $k = 0.05$, $s_0 = -1.5$, and $\kappa = 1.0$. Overall, less AV is needed than for the unstructured mesh. Also, it is important to note that all previous results in this chapter obtained a similarly unsatisfying convergence. The resulting heat flux is shown in Figure 5.13. Despite the symmetrical mesh, asymmetries occur in the solution. These are in fact due to the lack of convergence of the residuals. After the simulation was stopped, the AV parameters were changed to $k = 0.05$, $s_0 = -1.2$, and $\kappa = 0.5$ in order to not dissipate at all in the boundary layer, and the simulation was restarted. It then converged very quickly. The progression of residuals is shown in Figure 5.14, where the typical slow convergence until stalling followed by a very rapid convergence can be observed.

Both the total pressure and the heat flux results were smooth, as can be seen in Figures 5.15(a) and 5.15(b). Qualitatively, heat flux results in literature have a trend more similar to what is obtained for the total pressure, meaning that the slope is lower on the sides and the peak is narrower in the middle section. This difference could be explained by the low order or inviscid flux employed. Unfortunately, higher orders and other Riemann solvers turned simulations unstable and this could thus not be assessed.

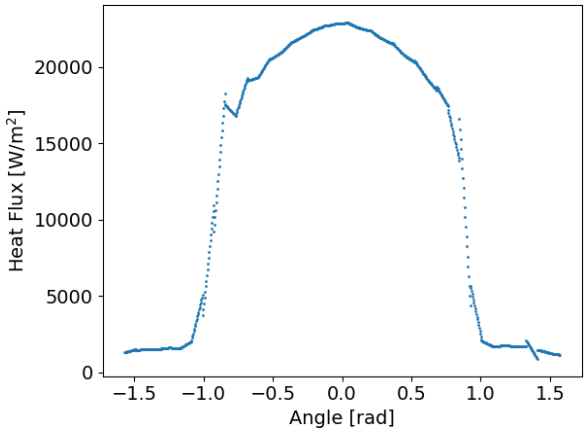


Figure 5.13: Heat flux for the half-cylinder test case on the wall, before full convergence, with $p = 1$, Lax-Friedrichs, $k = 0.05$, $s_0 = -1.5$, and $\kappa = 1.0$

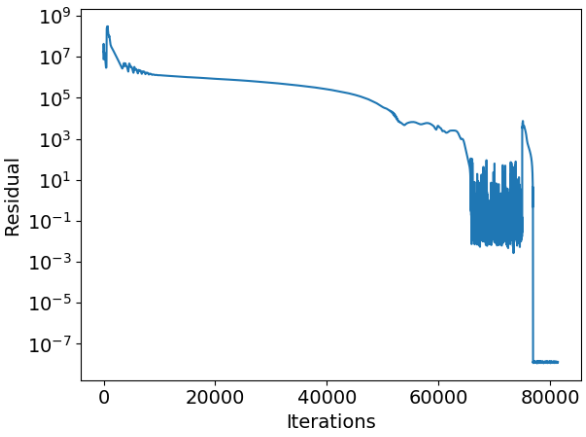
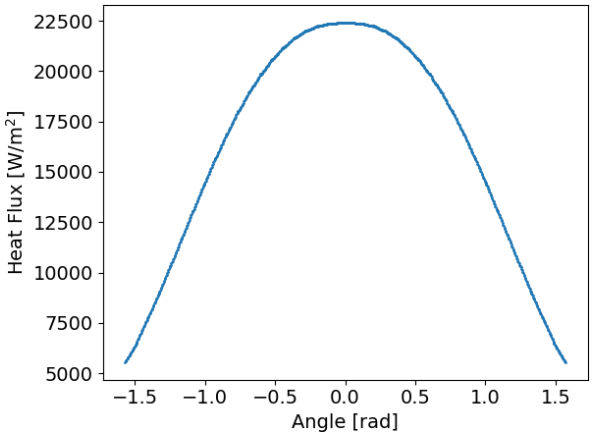
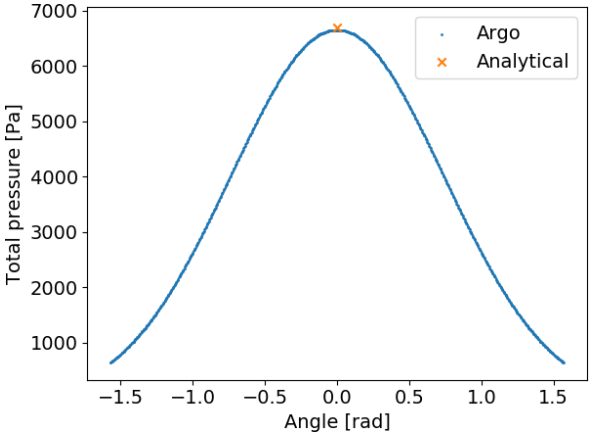


Figure 5.14: Convergence for the half-cylinder test case



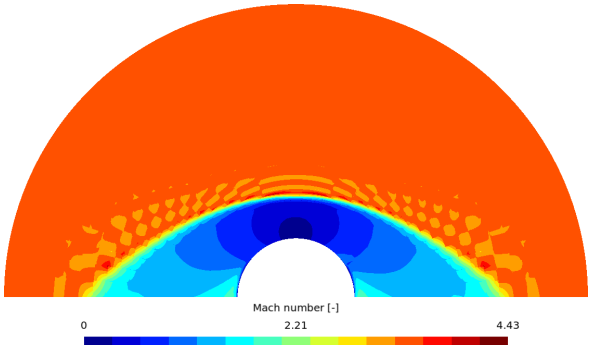
(a) Heat flux, after full convergence



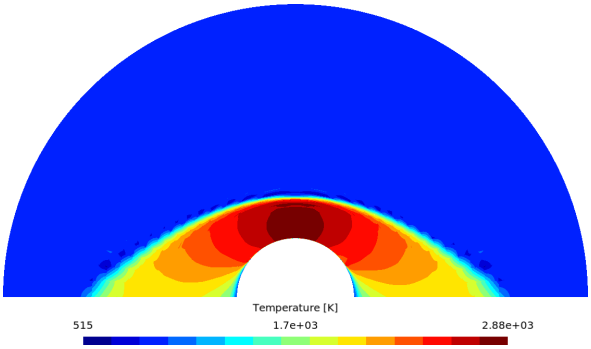
(b) Total pressure, after full convergence

Figure 5.15: Solution to the half-cylinder test case on the wall on a structured mesh, with $p = 1$, Lax-Friedrichs, $k = 0.05$, $s_0 = -1.2$, and $\kappa = 0.5$

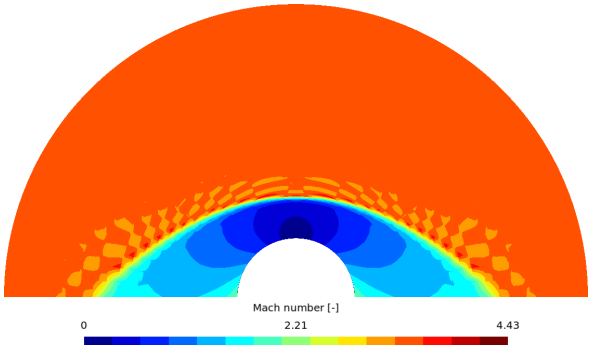
The Mach number and temperature distributions are shown in Figures 5.16(c) and 5.16(d) to highlight that despite obtaining satisfying convergence and heat flux and pressure results, the solution around the shock is not as good as it could be. Figures 5.16(a) and 5.16(b) are provided for comparison and correspond to the solution at the moment the convergence stagnates and the simulation is stopped. One can see that since the AV parameters were tuned with the aim of avoiding dissipating in the boundary layer, spurious oscillations remain present around the shock. This is particularly visible in Figure 5.16(c), the Mach number distribution, with Figure 5.16(a) provided for comparison. Potentially, this could be partly fixed by increasing the AV magnitude. It can also be seen, mostly in the temperature field, how the boundary layer thickness is now significantly thinner with respect to that in Figure 5.16(b).



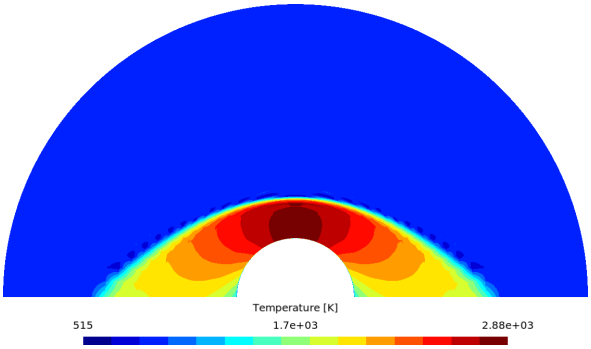
(a) Mach number, with AV in the boundary layer ($k = 0.05$, $s_0 = -1.5$, and $\kappa = 1.0$)



(b) Temperature, with AV in the boundary layer ($k = 0.05$, $s_0 = -1.5$, and $\kappa = 1.0$)



(c) Mach number, without AV in the boundary layer ($k = 0.05$, $s_0 = -1.2$, and $\kappa = 0.5$)



(d) Temperature, without AV in the boundary layer ($k = 0.05$, $s_0 = -1.2$, and $\kappa = 0.5$)

Figure 5.16: Solution to the half-cylinder test case with and without AV in the boundary layer, with $p = 1$ and Lax-Friedrichs

5.3. Summary

Overall, this chapter has highlighted the challenges in obtaining a reasonable heat flux result. Symmetry was found to be a significant issue with non-symmetric meshes, but the most important insight is that no artificial viscosity at all should be applied in the boundary layer. For that, changing the variable for shock detection is not sufficient and the AV parameters need to be chosen carefully so as to not dissipate near the wall.

6

Automating the AV threshold

Previous chapters have been focused entirely on applying existing methods to different test cases in order to understand Argo's shock capturing capabilities. Although some adjustments were made, the artificial viscosity method itself was left untouched. Nevertheless, the information gathered highlighted an opportunity to improve the method. This improvement, which consists of making the AV threshold free of user input, has been implemented and tested. The first section places this change in context, presents the strategy used to automate the threshold and briefly mentions the impact of this change. The impact is shown more specifically in the following sections, which show the results achieved with it.

6.1. Concept implementation

It is now clear that a key drawback of AV methods is that the user-defined parameters can be challenging to tune. It would thus be beneficial to find an algorithm which could tune these parameters automatically. An optimisation procedure was proposed by Ching et al. [14] to tune the amount and width, for which the results were shown on a single test case. A later paper by Ching et al., reference [15], did not even mention this procedure, suggesting that it still lacks robustness. Lv et al. [57] derived a formula based on the Euler equations to tune the sensor threshold, for their entropy-residual shock detector. A more general approach, based on outlier detection, was proposed by Vuik & Ryan [86]. However, wedge test case results indicate that setting the threshold automatically based on an outlier approach would probably not perform well. Indeed, such method would work best if most of the domain is attributed the lowest sensor value of -16 and only around the shock would the values be much higher. In this case, however, the transition region prevents the elements at the shock from standing out. This could potentially be the case for the half-cylinder as well, where the sensor values remain rather large behind the shock.

Furthermore, the experience gained in tuning the AV parameters through the different test cases indicated that, when possible, a good strategy is to set the threshold s_0 and interval κ such that the maximum amount of AV is applied where the sensor value s_e is maximum. In other words, the shock detector assigns a value s_e to each element of the mesh. This value is a measure of smoothness in the solution and it was observed s_e is maximum at the shock location. The parameters s_0 and κ are thus chosen such that the largest AV magnitude defined by k is applied at the shock location, that is $s_0 + \kappa = s_{e,max}$. The Shu-Osher problem showed that this strategy fails when $s_{e,max}$ varies as the simulation runs due to changing shock strength, since the parameters need to be set before the simulation starts. This issue inspired the creation of a new strategy to tune the threshold automatically. The idea is in fact very simple and carries out the manual strategy just described automatically and at each time step. At each time step, it finds the maximum sensor value in the domain and subtracts the interval value (defined by the user) to obtain the threshold. That is

$$s_0 = s_{e,max} - \kappa. \tag{6.1}$$

The method as such would add artificial viscosity everywhere if the flow is uniform in the whole domain. To avoid this unwanted behaviour, the code prevents the threshold to take any value below -5. Although only one out of three parameters is now free of user input, the difference it makes in choosing the parameters is noticeable. Three values to specify result in a very large number of combinations, with sometimes many giving satisfying results. Moreover, it may be that a combination needs to be chosen in order for the simulation

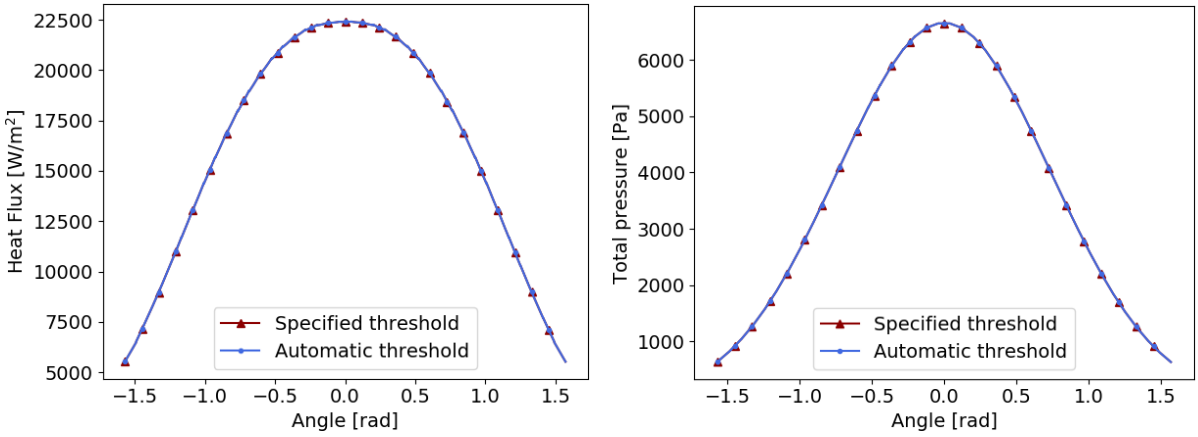
to be stable or to avoid dissipating in the boundary layer, leading to suboptimal smoothing at the shock. Using Equation (6.1), the user does not need to think about sensor values and a small interval κ can usually be used since the method will follow variations in the shock sensor. If the AV field around the shock is clearly too thin, one simply needs to increase κ. If the AV field looks right but oscillations still occur, one simply needs to increase the magnitude k. In addition to easing the manual tuning of AV parameters, this method proved to be clearly beneficial in terms of stability. Simulations that would previously crash were now achieved with little effort. The next sections discuss the results.

6.2. Results for the half-cylinder test case

This section aims at testing the new method on the half-cylinder test case, for which many limitations were encountered. The first step was to repeat a previous successful simulation to check whether the same results are obtained. Then, the automatic tuning of the threshold is applied to new simulations.

6.2.1. Verification

Before this method was used on a new simulation, it was decided to test whether the heat flux result obtained in Figure 5.15(a) could be reproduced. Since the AV parameters had been very carefully selected to avoid dissipating in the boundary layer, it is worth ensuring that this method will not negatively impact this effort. Therefore, starting from the solution shown in Figure 5.13, the simulation was run in the identical setup as that of Figure 5.15(a). It converged quickly to the solution shown in Figures 6.1(a) and 6.1(b). The heat flux and pressure look identical as the previous results.



(a) Heat flux

(b) Total pressure

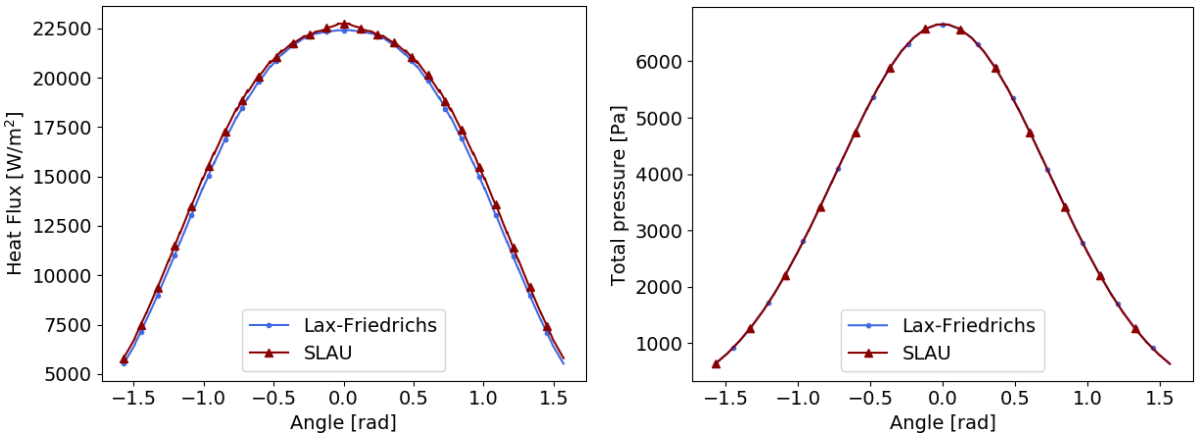
Figure 6.1: Solution to the half-cylinder test case on the wall, with $p = 1$, Lax-Friedrichs, $k = 0.05$, and $\kappa = 0.4$

The logical next step was to try running simulations which crashed without the automatic threshold strategy, to test its potential. A simulation at $p = 2$ and another with the inviscid scheme SLAU were achieved and their outcome is presented below.

6.2.2. Half-cylinder simulation with SLAU flux

Previous chapters have confirmed the trend observed in literature that the choice of inviscid flux scheme has little effect, especially at high-orders. The heat flux and total pressure at the cylinder wall, shown in Figures 6.2(a) and 6.2(b) respectively, are in line with this. The only difference between Lax-Friedrichs and SLAU is in the heat flux, where SLAU results in a subtle cusp at the stagnation point and a minor increment overall. Note that the AV parameters had to be changed to $k = 0.05$ and $\kappa = 0.7$, thus the footprint was slightly increased to compensate the low dissipation associated to SLAU. The simulation used an homogeneous initial condition with freestream values.

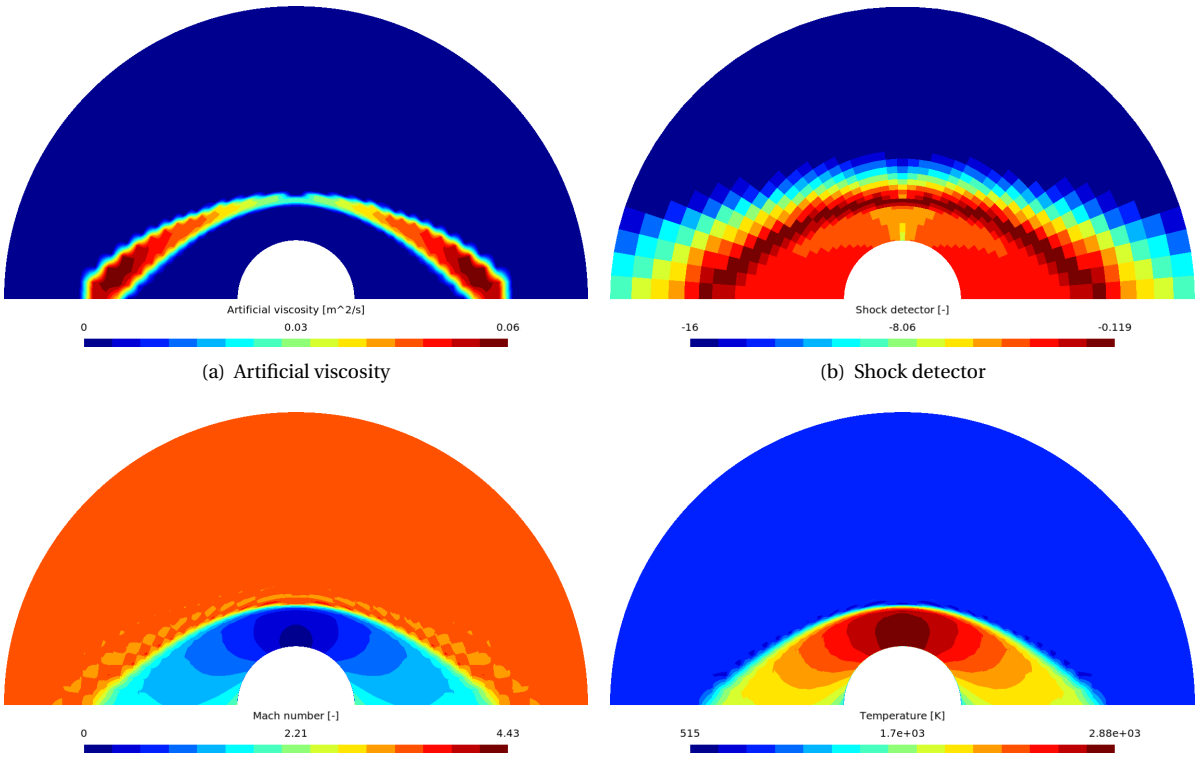
The artificial viscosity and shock detector on the domain are plotted in Figures 6.3(a) and 6.3(b). The artificial viscosity footprint is larger than before around the shock, but is only applied there. A small dip can also be seen on the stagnation line, possibly causing the cusp observed in the heat flux. The shock detector further suggests the heat flux cusp is caused by the shock capturing, with a minimum at the stagnation point.



(a) Heat flux (b) Total pressure

Figure 6.2: Solution to the half-cylinder test case on the wall, with $p = 1$, SLAU, $k = 0.05$ and $\kappa = 0.7$

The Mach number distribution, shown in Figure 6.3(c), still presents spurious oscillations around the shock. Their magnitude has decreased a little compared to Lax-Friedrichs results, but this is more likely due to the different AV parameters. The temperature field, Figure 6.3(d), is very similar but the contours are smoother.



(a) Artificial viscosity (b) Shock detector

(c) Mach number (d) Temperature

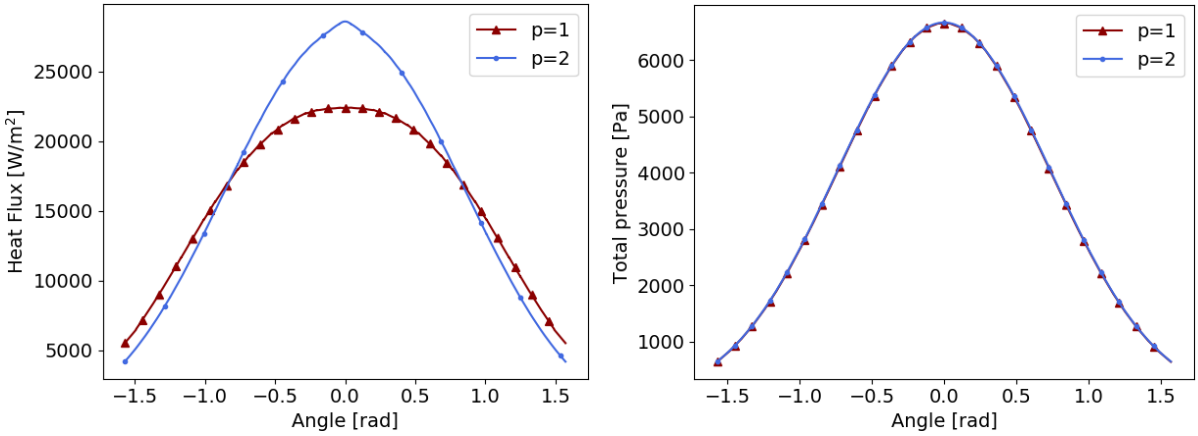
Figure 6.3: Solution to the half-cylinder test case, with $p = 1$, SLAU, $k = 0.05$ and $\kappa = 0.7$

6.2.3. Half-cylinder simulation at p=2

With the new method, a $p = 2$ solution could be obtained with little effort, with Lax-Friedrichs and the parameters $k = 0.05$ and $\kappa = 0.5$. As usual with steady cases, the initial condition for the $p = 2$ simulation was a converged $p = 1$ solution. The rest of the simulation set-up was kept identical. The heat flux and total pressure on the cylinder surface are shown in Figures 6.4(a) and 6.4(b), respectively. While the total pressure

result did not change from the $p = 1$ solution, the magnitude and trend of the heat flux did. The trend now looks more similar to that of the total pressure and to results from literature. Although the order of magnitude is the same, the heat flux is now greater at the stagnation point and lower on the sides. This is expected since at $p = 2$ the DG method should better capture the strong gradients in that region. Nevertheless, if the trend and magnitude are now more accurate, the small cusp at the curve's peak is an issue that already appeared in Figure 6.2(a). This feature is worth consideration and is discussed in the next paragraph.

In a finite volume method, the cause of this cusp could quickly be associated to the carbuncle phenomenon, a "local unpleasant displacement of the bow shock wave shape" [58, p1]. This error at the shock propagates downstream and deteriorates the quality of the solution at the wall. According to MacCormack [58], significant errors at the wall can occur even if a clear carbuncle does not appear. This could be what happens in previous results. Despite the limited results in DG, the appearance of a cusp at the stagnation point can be found in the literature as well. Barter [4] obtained a very similar cusp for a test case at $M = 17.6$ and, although does not offer an explanation for it, mentions mesh resolution as a potential cause for an inaccuracy in his pressure at the wall. If mesh resolution is negatively affecting the pressure, it is likely to do so for the heat flux as well. In another work by Ching et al. [15], also on a $M = 17.6$ case, the cusp appears when using a discrete artificial viscosity field. This suggests that, whether due to the mesh or the smoothing, inaccuracies in the shock capturing can result in this error at the stagnation point.



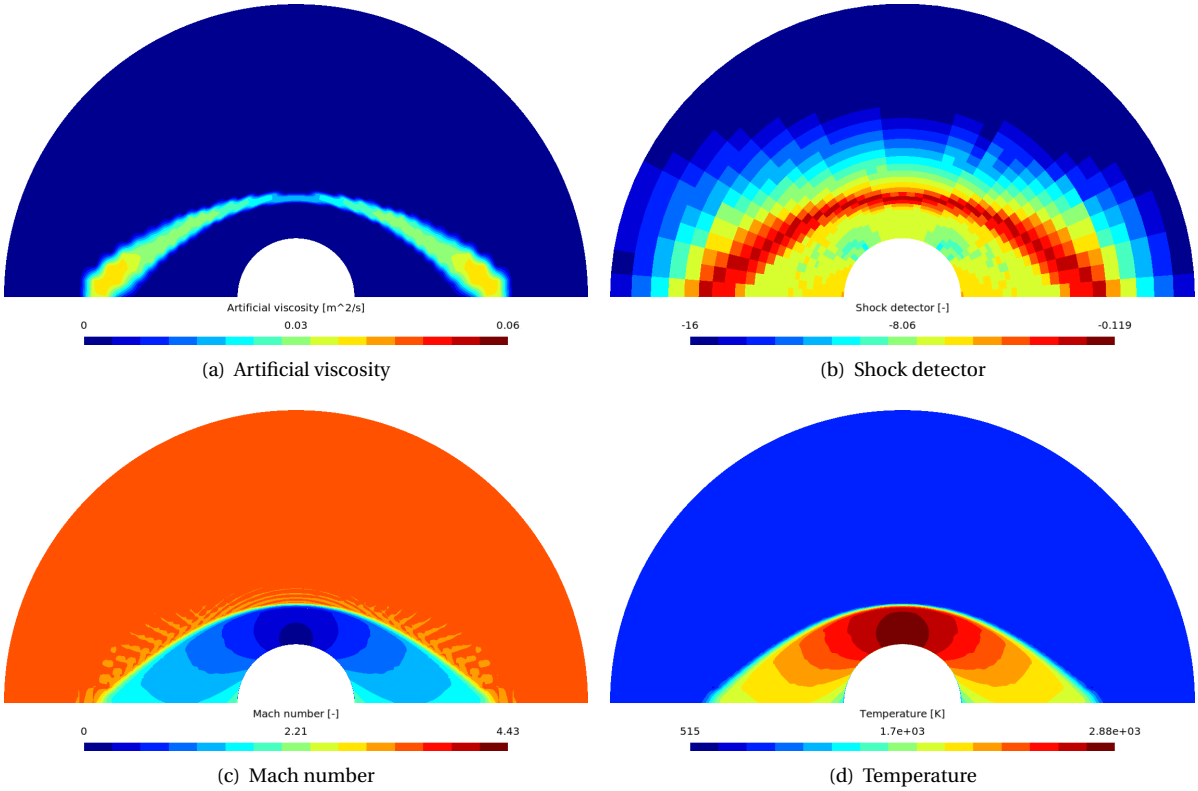
(a) Heat flux

(b) Total pressure

Figure 6.4: Solution to the half-cylinder test case on the wall, with $p = 2$, Lax-Friedrichs, $k = 0.05$ and $\kappa = 0.5$

It is interesting to look again at the artificial viscosity and shock detector plots, Figures 6.5(a) and 6.5(b), for the $p = 2$ case. The artificial viscosity field does not look very different, however the magnitude has decreased due to the h/p scaling. The shock detector shows that a much greater difference exists between elements in the shock and those in smooth regions. This makes it easier to apply AV without dissipating in regions where it is unwanted. The Mach number distribution, in Figure 6.5(c), shows that spurious oscillations have decreased. Figure 6.5(d), the temperature field, is slightly smoother than results with SLAU, and features reduced spurious oscillations as well. Results at $p = 2$ are thus an improvement overall. It is worthwhile noting that this heat flux result is, to the best of the author's knowledge, the first to be obtained with the Lax-Friedrichs flux and Argo's specific smoothing technique of the AV field. It shows that it is possible to successfully simulate this challenging test case with DG and a combination of simple methods. This work also documented the steps followed to achieve such result, information generally lacking in the literature. Hopefully, this will help future research in DG methods oriented towards heat flux predictions.

Further investigation of the half-cylinder test case is left for future work. The logical step would be to further increase the order and the Mach number. However, it is chosen to dedicate the remainder of this chapter on the unsteady problem that first motivated the automatic tuning of the threshold, the Shu-Osher test case.



(a) Artificial viscosity

(b) Shock detector

(c) Mach number

(d) Temperature

Figure 6.5: Solution to the half-cylinder test case, with $p = 2$, Lax-Friedrichs, $k = 0.05$ and $\kappa = 0.5$

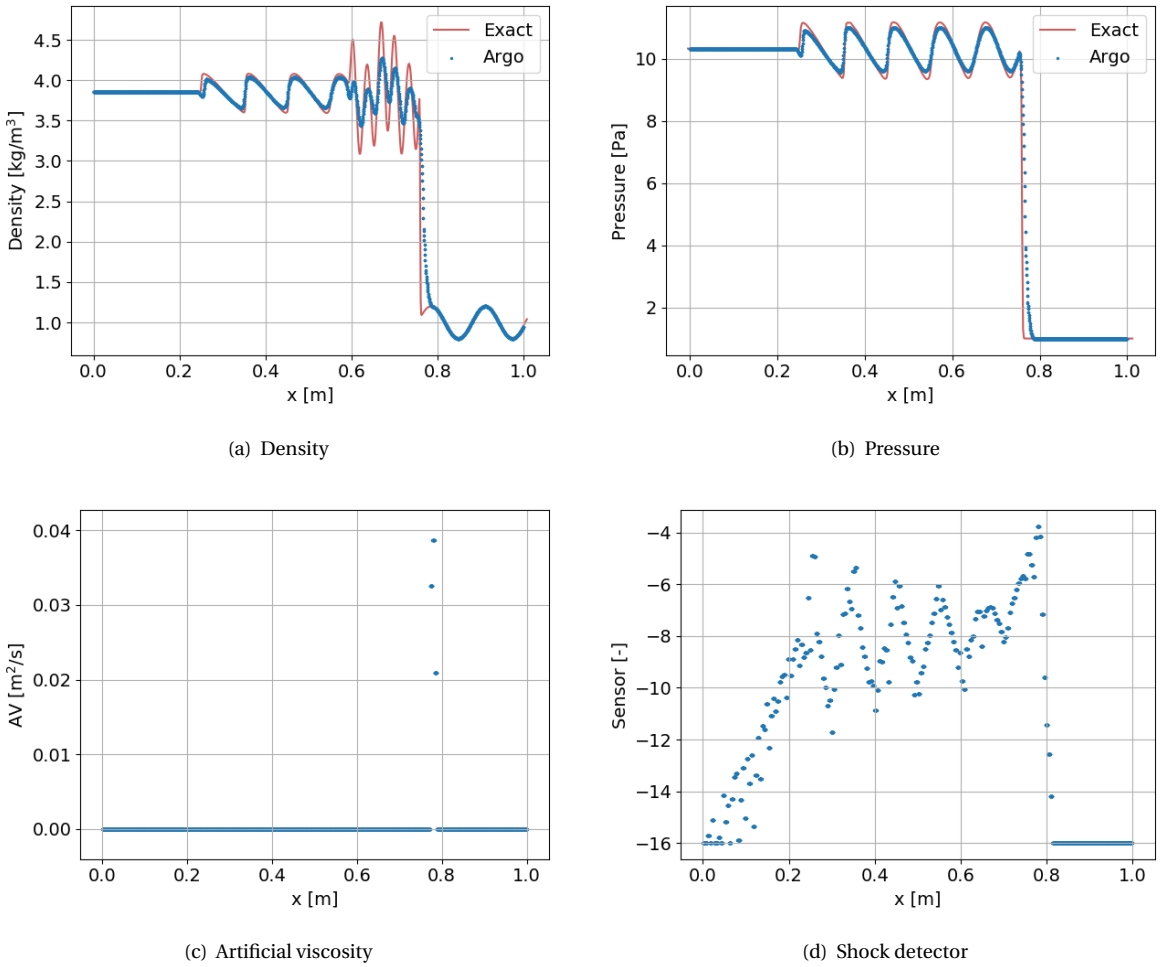
6.3. The Shu-Osher problem, revisited

The Shu-Osher problem is expected to greatly benefit from the automatic threshold tuning. The AV parameters can now be chosen focusing entirely on removing oscillations since the threshold will adapt to the simulation needs at each time step. Decent results were obtained with the Lax-Firedrichs flux at $p = 2$ previously, except in terms of spurious oscillations. The first simulation thus decided to assess whether those oscillations would now be removed.

6.3.1. Shu-Osher simulation at p=2

The simulation was run with Lax-Friedrichs and AV parameters $k = 0.05$ and $\kappa = 0.5$. Worth noting is that the interval κ is smaller than that used in Chapter 4. Since the threshold adapts to the shock sensor results, the interval does not need to account for that variation and can thus be reduced. The results shown in Figures 6.6(a) and 6.6(b) confirm that the solution is much smoother. Unfortunately, the smoothing has been excessively strong, to the point that the solution looks similar to that at $p = 1$ obtained previously. It was mentioned before that Lv et al. [57] and Vuik & Ryan [86] have already suggested a method to tune the threshold automatically. The results from Lv et al. are similar to those shown below, although the latter are slightly more dissipated. Vuik & Ryan's results come very close to the exact solution but their mesh contains 512 elements. The results obtained in this work could most likely be improved by fine-tuning the AV parameters.

In addition to density and pressure results, the artificial viscosity magnitude and shock detector are plotted in Figures 6.6(c) and 6.6(d), respectively. Figure 6.6(c) shows a familiar behaviour in that artificial viscosity is only applied at the shock. One key difference appears in the shock detector compared to results obtained in Chapter 4. Upstream of the shock, the shock detector returns -16 rather than oscillating values. This is because the variable for the shock detector is now the pressure, which is constant in that region. Assuming only the shock is of interest, more data would be needed to assess whether the pressure or the density is the best variable for shock detection in the Shu-Osher problem.



(a) Density

(b) Pressure

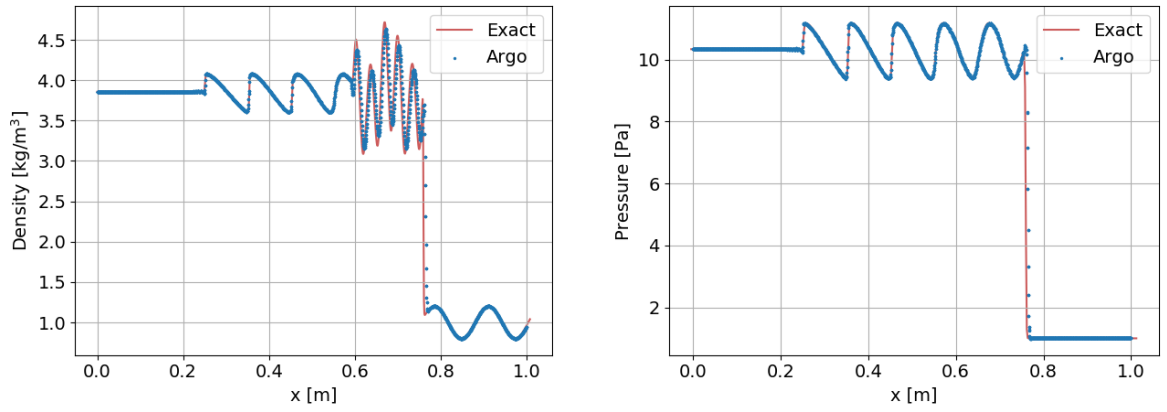
(c) Artificial viscosity

(d) Shock detector

Figure 6.6: Solution to the Shu-Osher test case at $p = 2$, with 1800 DOF, Lax-Friedrichs, $k = 0.05$ and $\kappa = 0.5$

6.3.2. Shu-Osher simulation at p=4

Results at $p = 4$ were also achieved before, even though the AV parameters were particularly difficult to tune. In this case, the simulation worked with little effort, clearly highlighting the improvements in stability that this automated threshold strategy brings. The $p = 4$ solution with $k = 0.02$ and $\kappa = 0.5$ is shown in Figures 6.7(a) and 6.7(b) for the pressure and density, respectively. The match between the exact and Argo solution is much better than that at $p = 2$, as expected. The solution can also be seen to be free of overshoots and spurious oscillations, which previously polluted the solution. Nevertheless, the high frequency oscillations are not captured with the same accuracy as before. Despite the low values for k and κ, the solution still appears more dissipated than in Figure 4.18(a). The agreement with the exact solution is very similar to what Lv et al. [57] obtained. The pressure results match the exact solution closely, as expected.

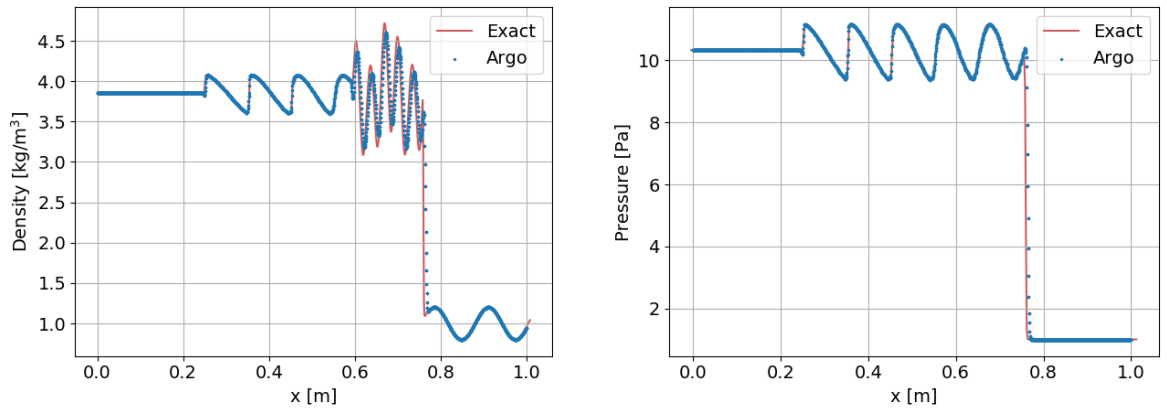


(a) Density (b) Pressure

Figure 6.7: Solution to the Shu-Osher test case at $p = 4$, with 5000 DOF, Lax-Friedrichs, $k = 0.02$ and $\kappa = 0.5$

6.3.3. Shu-Osher simulation with Roe flux

In Chapter 4, the only inviscid flux that allowed stable simulations was the Lax-Friedrichs scheme. Due to the improved stability of the automatic threshold tuning, it is now possible to run simulations with different Riemann solvers. Figures 6.8(a) and 6.8(b) show the density and pressure results respectively, achieved with the Roe scheme. The results appear identical to those with Lax-Friedrichs. Further confirmation that the inviscid flux has little impact on the solution at high orders is thus obtained.



(a) Density (b) Pressure

Figure 6.8: Solution to the Shu-Osher test case at $p = 4$, with 5000 DOF, Roe, $k = 0.05$ and $\kappa = 0.5$

6.4. Summary

To conclude this chapter on the implementation and application of a new strategy to set the threshold automatically and adaptively, the main findings are summarised. Overall, removing one user-defined parameter from the set was found to provide a clear benefit, the task of finding an appropriate set of AV parameters being greatly simplified. Moreover, the continuous adaptation of the threshold to the problem needs improved stability significantly. Overall, these benefits allowed to simulate cases that were not achieved without the method. Clearly, testing the method on two test cases is not sufficient to draw strong conclusions about its robustness. One area where it might fail is with interacting shocks of different strengths. The double Mach reflection problem by Woodward & Colella [89] is recommended to test this, but will not be investigated in this work.

7

Conclusions and recommendations

This work studied the performance of artificial viscosity as a shock capturing method for a discontinuous Galerkin discretisation. The multi-physics CFD platform Argo developed at Cenaero was used for all computations. The key aspect of the artificial viscosity technique is the presence of three user-defined parameters, and the tuning of those. The three parameters k, s_0 and κ define the magnitude, location, and width of the artificial viscosity. A number of test cases with increasing difficulty were investigated to understand the capabilities and limitations of the computational tool. The first application was a steady supersonic wedge test case governed by the Euler equations. A parametric study at Mach 2 showed that specifying the user-defined parameters for the artificial viscosity is not a trivial task. Generally, two of the three parameters, the interval and the threshold, are chosen such that artificial viscosity is applied where desired. The third parameter, the magnitude, specifies how much artificial viscosity is applied. One could apply a large quantity on a small region, a small quantity over a large region, or choose a middle point. Although a number of options appeared better than others, spurious oscillations remained in each of the parameter combinations tried. Furthermore, it appeared that the user may need to trade-off shock thickness with removing spurious oscillations. To obtain a significant improvement to the results, it was necessary to increase the polynomial order or refine the mesh. Even then, the shock strength varied across the length of the shock, leading to a non-homogeneous post-shock solution. Increasing the Mach number highlighted the difficulties in keeping the simulation stable, requiring to use artificial viscosity on a large region and increasing the shock thickness significantly yet still obtaining strong spurious oscillations.

The 1D unsteady, inviscid test cases of Sod [80] and Shu-Osher [79] were then simulated for different mesh refinements, polynomial orders from 0 to 7, and Riemann solvers Lax-Friedrichs, SLAU, and AUSM$^+$up. The general conclusion for the first which has constant shock strength was that Argo can obtain good results as long as a sufficient refinement in mesh size or order is used. The shock was better captured than the contact discontinuity, and the Riemann solver had little effect at polynomial orders larger than zero. The Shu-Osher test case was more challenging due to the varying shock strength. As opposed to the first unsteady problem, this test case could not be run without artificial viscosity for polynomial orders larger than 0. It highlighted the difficulty of tuning the artificial viscosity parameters when the shock strength varies in time since the chosen set needs to account for variations in the shock detector. Furthermore, it became clear that a method keeping flow variables within physical bounds is necessary for strong shocks. As for the first unsteady problem, good results were obtained with sufficient order and mesh resolution and the inviscid flux choice did not have much effect except at polynomial order zero were it was significant. Nevertheless, spurious oscillations remained.

Prediction of the heat flux at the cylinder wall in supersonic flow proved challenging as well. The heat flux often suffered from strong oscillations even when plots of solution variables in the domain and the total pressure at the wall appeared smooth. The key change which allowed to obtain a smooth heat flux profile was to tune artificial viscosity parameters such that no non-physical dissipation at all is added in the boundary layer. The unstructured, asymmetrical mesh led to a strongly asymmetrical result but this was fixed with a symmetrical mesh. Despite promising results, attempts at different Riemann solvers or high orders were unsuccessful.

A method was developed to tune automatically the threshold, allowing the artificial viscosity to adapt to changing problem needs as the simulation progresses. This change made the tuning of artificial viscosity pa-

rameters much easier and provided clear stability benefits. At polynomial order 2, the heat flux appeared accurate and was the first achieved with the combination of the Lax-Friedrichs flux, the Interior Penalty method, and a simpler smoothing technique of the artificial viscosity field in discontinuous Galerkin methods. A comparison with a state-of-the-art finite volume method would help evaluate the accuracy of the obtained results.

Considering the results achieved with artificial viscosity and its recent progress with respect to other stabilisation techniques in DG, it is worth building on recent efforts to make artificial viscosity more robust rather than switching attention to other methods. A number of recommendations are proposed for future work. First, the wedge test case could be revisited with a focus on Mach 4 and higher. With a method keeping variables within physical bounds and the automatic threshold, the challenges encountered in the early stages of this work could be significantly eased. It would also be valuable to further analyse the Shu-Osher problem. With the automatic threshold, the results were found to be excessively dissipated. Further tuning the AV parameters should allow to resolve this issue, but it would be useful to confirm this with actual results. For the half-cylinder test case, wall heat flux results at the stagnation point were verified through an approximate formula at the limit of its range of validity. It is important to confirm the accuracy of the wall heat flux with a proven method. In addition, the orders considered in this work remained rather low. It would be useful to assess whether higher polynomial orders could reduce issues such as asymmetries seen when using unstructured meshes. Furthermore, hypersonic Mach numbers were not achieved in this work. Typically, hypersonic simulations are obtained with order sequencing meaning that a simulation at order 1 uses a solution at order 0 as initial condition. The discretisation of the diffusive term in Argo is inconsistent at order 0, thus preventing the use of order sequencing. It is likely that using a diffusive scheme consistent at order 0 would make such challenging problems easier to simulate. The main recommendation to achieve hypersonic speeds would thus be to start by implementing a different discretisation of the diffusive term such as the second scheme of Bassi and Rebay (BR2) [7]. A challenging task would be to achieve an accurate heat flux on a fully unstructured mesh with similar levels of refinement, in order to bring the computational capabilities closer to being applied on complex geometries. Reducing the significant refinements in the boundary layer currently needed would also be beneficial.

The automatic threshold tuning needs to be further analysed in various test cases. Although its performance appeared as promising in this work, robustness issues could occur in problems where different shocks with different strengths occur. The strategy could also be expanded such that all parameters are tuned automatically and in a robust manner. This would be a significant achievement and artificial viscosity methods would become much easier to use in practise. However, this is a challenging task for which much effort is likely still needed. Furthermore, it would be worthwhile to compare the performance of the automatic threshold tuning proposed in this work to the more complicated outlier approach found in the literature. With the artificial viscosity method used in this work, it was argued that the outlier approach would likely not work well on certain problems due to insufficient contrast in the shock detector results between the shock and the rest of the domain. However, this hypothesis remains to be tested. Assessing how the automatic threshold tuning works with different shock detectors would be useful too. Potentially, other detectors could provide better contrast between discontinuous and smooth regions. Suggestions would be the detector by Lv et al. [57] which was found to work better on certain test cases and the detector by Ching et al. [15] which has been used successfully on several hypersonic problems. It is argued that a detector achieving sufficient contrast between shock and boundary layer regions is key to avoid dissipating in the latter region. This issue has been tackled successfully in [15] by removing all the smallest magnitudes of artificial viscosity typically corresponding with those in the boundary layer but this is not an ideal solution as it adds arbitrary parameters.

Once the artificial viscosity method becomes reliable on simple test cases, the physical complexity of the problem simulated needs to be increased. Investigating how the artificial viscosity method used in this work affects and interacts with turbulence remains to be done. Furthermore, high-temperature effects, chemistry, and/or material ablation could be added to the problems being simulated. A difficulty could appear if the shock detector provokes artificial viscosity being applied in the chemically reacting region. Finally, it is worth adding a note on cost. For the Sod test case using Lax-Friedrichs and with artificial viscosity activated, the simulation takes 1.1s to run on 20 elements at $p = 1$ (80 DOF). This increases to 2.66s at $p = 2$ (180 DOF) and 6.35s at $p = 3$ (320 DOF). If the order is kept at $p = 1$ but the number of elements is increased to 40 (160 DOF), the run time is 1.54s. For 200 elements (800 DOF), it is 4.85s. In this very brief analysis, it is clear that refining in p increases cost more rapidly than reducing h. Discontinuous Galerkin methods may thus need to be combined with hp adaptation techniques in order to become an attractive alternative to low-order finite volume methods.

Bibliography

[1] William Ailor, Wayne Hallman, Gary Steckel, and Michael Weaver. Analysis of reentered debris and implications for survivability modeling. In *4th European Conference on Space Debris*, volume ESA SP-587, July 2005.

[2] Douglas N. Arnold, Franco Brezzi, Bernardo Cockburn, and L. Donatella Marini. Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM Journal on Numerical Analysis*, 39(5):1749–1779, jan 2002. doi: 10.1137/s0036142901384162.

[3] Harold Atkins and Alyssa Pampell. Robust and accurate shock capturing method for high-order discontinuous Galerkin methods. In *20th AIAA Computational Fluid Dynamics Conference*. American Institute of Aeronautics and Astronautics, jun 2011. doi: 10.2514/6.2011-3058.

[4] Garrett E. Barter. *Shock Capturing with PDE-Based Artificial Viscosity for an Adaptive Higher-Order Discontinuous Galerkin Finite Element Method*. PhD thesis, Massachusetts Institute of Technology, 2008.

[5] Garrett E. Barter and David L. Darmofal. Shock capturing with PDE-based artificial viscosity for DGFEM: Part i. formulation. *Journal of Computational Physics*, 229(5):1810–1827, mar 2010. doi: 10.1016/j.jcp.2009.11.010.

[6] Timothy J. Barth and Herman Deconinck, editors. *High-Order Methods for Computational Physics*. 9. Springer Berlin Heidelberg, 1999. ISBN 978-3-662-03882-6. doi: 10.1007/978-3-662-03882-6.

[7] Francesco Bassi, Stefano Rebay, G Mariotti, Savini Pedinotti, and M Savini. A high-order accurate discontinuous finite element method for inviscid and viscous turbomachinery flows. In R. Decuypere and G. Dibelius, editors, *Proceedings of the 2nd European Conference on Turbomachinery, Fluid Dynamics and Thermodynamics*, pages 99–108. Technologisch Instituut, Antwerpen, Belgium, 1997.

[8] Ankush Bhatia, Subrata Roy, and Ryan Gosse. 2-D hypersonic non-equilibrium flow simulation using r-p adaptive time-implicit discontinuous Galerkin method. In *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*. American Institute of Aeronautics and Astronautics, jan 2013. doi: 10.2514/6.2013-302.

[9] Rupak Biswas, Karen D. Devine, and Joseph E. Flaherty. Parallel, adaptive finite element methods for conservation laws. *Applied Numerical Mathematics*, 14(1-3):255–283, apr 1994. doi: 10.1016/0168-9274(94)90029-9.

[10] Jean-Baptiste Brachet. *A dynamic multiscale viscosity algorithm for shock capturing in Runge-Kutta Discontinuous Galerkin methods*. PhD thesis, Massachusetts Institute of Technology, 2005.

[11] Michael J. Brazell and Dimitri J. Mavriplis. 3D mixed element discontinuous Galerkin with shock capturing. In *21st AIAA Computational Fluid Dynamics Conference*. American Institute of Aeronautics and Astronautics, jun 2013. doi: 10.2514/6.2013-3064.

[12] A. Burbeau, P. Sagaut, and Ch.-H. Bruneau. A problem-independent limiter for high-order Runge–Kutta discontinuous Galerkin methods. *Journal of Computational Physics*, 169(1):111–150, may 2001. doi: 10.1006/jcph.2001.6718.

[13] Nicholas Burgess and Dimitri Mavriplis. Computing shocked flows with high-order accurate discontinuous Galerkin methods. In *42nd AIAA Fluid Dynamics Conference and Exhibit*. American Institute of Aeronautics and Astronautics, jun 2012. doi: 10.2514/6.2012-2715.

[14] Eric Ching, Yu Lv, and Matthias Ihme. Development of discontinuous Galerkin method for hypersonic heating prediction. In *55th AIAA Aerospace Sciences Meeting*. American Institute of Aeronautics and Astronautics, jan 2017. doi: 10.2514/6.2017-0311.

[15] Eric J. Ching, Yu Lv, Peter Gnoffo, Michael Barnhardt, and Matthias Ihme. Shock capturing for discontinuous Galerkin methods with application to predicting heat transfer in hypersonic flows. *Journal of Computational Physics*, 376:54–75, jan 2019. doi: 10.1016/j.jcp.2018.09.016.

[16] Bernardo Cockburn and Chi-Wang Shu. TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws II: General framework. *Mathematics of Computation*, 52(186): 411, apr 1989. doi: 10.2307/2008474.

[17] Bernardo Cockburn and Chi-Wang Shu. The runge-kutta local projection p^1-discontinuous-Galerkin finite element method for scalar conservation laws. *ESAIM: Mathematical Modelling and Numerical Analysis - Modélisation Mathématique et Analyse Numérique*, 25(3):337–361, 1991.

[18] Bernardo Cockburn and Chi-Wang Shu. Runge-Kutta discontinuous Galerkin methods for convection-dominated problems. *Journal of Scientific Computing*, 16(3):173–261, 2001. doi: 10.1023/a:1012873910884.

[19] 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. *Journal of Computational Physics*, 84(1):90–113, sep 1989. doi: 10.1016/0021-9991(89)90183-6.

[20] Bernardo Cockburn, Suchung Hou, and Chi-Wang Shu. The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws. IV: The multidimensional case. *Mathematics of Computation*, 54(190):545, apr 1990. doi: 10.2307/2008501.

[21] Vít Dolejší and Miloslav Feistauer. *Discontinuous Galerkin Method*. Springer International Publishing, 2015. doi: 10.1007/978-3-319-19267-3.

[22] Zhifang Du and Jiequan Li. A hermite WENO reconstruction for fourth order temporal accurate schemes based on the GRP solver for hyperbolic conservation laws. *Journal of Computational Physics*, 355:385–396, feb 2018. doi: 10.1016/j.jcp.2017.11.023.

[23] Michael Dumbser and Raphaël Loubère. A simple robust and accurate a posteriori sub-cell finite volume limiter for the discontinuous Galerkin method on unstructured meshes. *Journal of Computational Physics*, 319:163–199, aug 2016. doi: 10.1016/j.jcp.2016.05.002.

[24] Michael Dumbser, Olindo Zanotti, Raphaël Loubère, and Steven Diot. A posteriori subcell limiting of the discontinuous Galerkin finite element method for hyperbolic conservation laws. *Journal of Computational Physics*, 278:47–75, dec 2014. doi: 10.1016/j.jcp.2014.08.009.

[25] Irene Pontijas Fuentes, Davide Bonetti, Federico Letterio, Gonzalo Vicario de Miguel, Gonzalo Blanco Arnao, Pedro Palomo, Cristina Parigini, Stijn Lemmens, Tobias Lips, and Ronny Kanzler. Upgrade of ESA's debris risk assessment and mitigation analysis (DRAMA) tool: Spacecraft entry survival analysis module. *Acta Astronautica*, dec 2017. doi: 10.1016/j.actaastro.2017.12.001.

[26] Anne Gelb and Eitan Tadmor. Enhanced spectral viscosity approximations for conservation laws. *Applied Numerical Mathematics*, 33(1-4):3–21, may 2000. doi: 10.1016/s0168-9274(99)00067-7.

[27] Christophe Geuzaine and Jean-François Remacle. Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, 79(11):1309–1331, sep 2009. doi: 10.1002/nme.2579.

[28] Peter Gnoffo and Jeffery White. Computational aerothermodynamic simulation issues on unstructured grids. In *37th AIAA Thermophysics Conference*. American Institute of Aeronautics and Astronautics, jun 2004. doi: 10.2514/6.2004-2371.

[29] S.K. Godunov. Difference method for the numerical computation of discontinuous solutions of fluid dynamics equations. *Mat. Sb.*, 47:271–306, 1959.

[30] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, third edition, 1996. ISBN 9780801854149.

[31] Sophie Gérald. *Méthode de Galerkin Discontinue et intégrations explicites-implicites en temps basées sur un découplage des degrés de liberté. Applications au système de Navier-Stokes.* PhD thesis, Université Pierre et Marie-Curie, 2014.

[32] Ami Harten. ENO schemes with subcell resolution. *Journal of Computational Physics*, 83(1):148–184, jul 1989. doi: 10.1016/0021-9991(89)90226-x.

[33] Ami Harten, Bjorn Engquist, Stanley Osher, and Sukumar R Chakravarthy. Uniformly high order accurate essentially non-oscillatory schemes, III. *Journal of Computational Physics*, 71(2):231–303, aug 1987. doi: 10.1016/0021-9991(87)90031-3.

[34] Amiram Harten, Peter D. Lax, and Bram van Leer. On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *SIAM Review*, 25(1):35–61, jan 1983. doi: 10.1137/1025002.

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

[36] Zhen-Hua Jiang, Chao Yan, and Jian Yu. Implicit high-order discontinuous Galerkin method with HWENO type limiters for steady viscous flow simulations. *Acta Mechanica Sinica*, 29(4):526–533, jul 2013. doi: 10.1007/s10409-013-0042-1.

[37] John D. Anderson Jr. *Fundamentals of Aerodynamics.* McGraw-Hill Science/Engineering/Math, fifth edition, 2001.

[38] John D. Anderson Jr. *Hypersonic and High-Temperature Gas Dynamics.* American Institute of Aeronautics and Astronautics, second edition, jan 2006. doi: 10.2514/4.861956.

[39] Keiichi Kitamura and Eiji Shima. Towards shock-stable and accurate hypersonic heating computations: A new pressure flux for AUSM-family schemes. *Journal of Computational Physics*, 245:62–83, jul 2013. doi: 10.1016/j.jcp.2013.02.046.

[40] A. Klöckner, T. Warburton, and J. S. Hesthaven. Viscous shock capturing in a time-explicit discontinuous Galerkin method. *Mathematical Modelling of Natural Phenomena*, 6(3):57–83, 2011. doi: 10.1051/mmnp/20116303.

[41] Heiner Klinkrad. *Space Debris: Models and Risk Analysis.* Springer Berlin Heidelberg, 2006. ISBN 9783540376743. doi: 10.1007/3-540-37674-7.

[42] Doyle Knight, José Longo, Dimitris Drikakis, Datta Gaitonde, Andrea Lani, Ioannis Nompelis, Bodo Reimann, and Louis Walpot. Assessment of CFD capability for prediction of hypersonic shock interactions. *Progress in Aerospace Sciences*, 48-49:8–26, jan 2012. doi: 10.1016/j.paerosci.2011.10.001.

[43] L. Krivodonova, J. Xin, J.-F. Remacle, N. Chevaugeon, and J.E. Flaherty. Shock detection and limiting with discontinuous Galerkin methods for hyperbolic conservation laws. *Applied Numerical Mathematics*, 48 (3-4):323–338, mar 2004. doi: 10.1016/j.apnum.2003.11.002.

[44] Lilia Krivodonova. Limiters for high-order discontinuous Galerkin methods. *Journal of Computational Physics*, 226(1):879–896, sep 2007. doi: 10.1016/j.jcp.2007.05.011.

[45] Peter D. Lax. Weak solutions of nonlinear hyperbolic equations and their numerical computation. *Communications on Pure and Applied Mathematics*, 7(1):159–193, feb 1954. doi: 10.1002/cpa.3160070112.

[46] Bram Van Leer, James L. Thomas, Philip L. Roe, and Richard W. Newsome. A comparison of numerical flux formulas for the Euler and Navier-Stokes equations. In *8th Computational Fluid Dynamics Conference*, pages 36–41. American Institute of Aeronautics and Astronautics, jun 1987. doi: 10.2514/6.1987-1104. URL `http://hdl.handle.net/2027.42/76365`.

[47] Meng-Sing Liou. A sequel to AUSM: AUSM+. *Journal of Computational Physics*, 129(2):364–382, dec 1996. doi: 10.1006/jcph.1996.0256.

[48] Meng-Sing Liou. A sequel to AUSM, part II: AUSM+up for all speeds. *Journal of Computational Physics*, 214(1):137–170, may 2006. doi: 10.1016/j.jcp.2005.09.020.

[49] Meng-Sing Liou and Jack R. Edwards. AUSM schemes and extensions for low Mach and multiphase flows. In *Lecture Series 1999-03: 30th Computational Fluid Dynamics*. von Karman Institute for Fluid Dynamics, 1999.

[50] Meng-Sing Liou and Christopher J. Steffen. A new flux splitting scheme. *Journal of Computational Physics*, 107(1):23–39, jul 1993. ISSN 0021-9991. doi: 10.1006/jcph.1993.1122.

[51] Tobias Lips and Bent Fritsche. A comparison of commonly used re-entry analysis tools. *Acta Astronautica*, 57(2-8):312–323, jul 2005. doi: 10.1016/j.actaastro.2005.03.010.

[52] Xiaodong Liu, Yidong Xia, and Hong Luo. A reconstructed discontinuous Galerkin method for compressible turbulent flows on 3D curved grids. *Computers & Fluids*, 160:26–41, jan 2018. doi: 10.1016/j.compfluid.2017.10.014.

[53] Manuel R. López-Morales and Antony Jameson. Stabilization of high-order methods for unstructured grids with local fourier spectral filtering: high-re simulations in coarse meshes. In *22nd AIAA Computational Fluid Dynamics Conference*. American Institute of Aeronautics and Astronautics, jun 2015. doi: 10.2514/6.2015-2447.

[54] Hong Luo, Joseph D. Baum, and Rainald Löhner. A hermite WENO-based limiter for discontinuous Galerkin method on unstructured grids. *Journal of Computational Physics*, 225(1):686–713, jul 2007. doi: 10.1016/j.jcp.2006.12.017.

[55] Hong Luo, Yidong Xia, Shujie Li, Robert Nourgaliev, and Chunpei Cai. A hermite WENO reconstruction-based discontinuous Galerkin method for the Euler equations on tetrahedral grids. *Journal of Computational Physics*, 231(16):5489–5503, jun 2012. doi: 10.1016/j.jcp.2012.05.011.

[56] Yu Lv and Matthias Ihme. Discontinuous Galerkin method for multicomponent chemically reacting flows and combustion. *Journal of Computational Physics*, 270:105–137, aug 2014. doi: 10.1016/j.jcp.2014.03.029.

[57] Yu Lv, Yee Chee See, and Matthias Ihme. An entropy-residual shock detector for solving conservation laws using high-order discontinuous Galerkin methods. *Journal of Computational Physics*, 322:448–472, oct 2016. doi: 10.1016/j.jcp.2016.06.052.

[58] Robert MacCormack. The carbuncle CFD problem. In *49th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*. American Institute of Aeronautics and Astronautics, jan 2011. doi: 10.2514/6.2011-381.

[59] Andreas Meister, Sigrun Ortleb, and Thomas Sonar. Application of spectral filtering to discontinuous galerkin methods on triangulations. *Numerical Methods for Partial Differential Equations*, 28(6):1840–1868, oct 2011. doi: 10.1002/num.20705.

[60] Andreas Meister, Sigrun Ortleb, Thomas Sonar Sonar, and Martina Wirz. An extended discontinuous Galerkin and spectral difference method with modal filtering. *ZAMM - Journal of Applied Mathematics and Mechanics / Zeitschrift für Angewandte Mathematik und Mechanik*, 93(6-7):459–464, aug 2012. doi: 10.1002/zamm.201200051.

[61] G. Moretti and M. Abbett. A time-dependent computational method for blunt body flows. *AIAA Journal*, 4(12):2136–2141, dec 1966. doi: 10.2514/3.3867.

[62] David Moro, Ngoc Cuong Nguyen, and Jaime Peraire. Dilation-based shock capturing for high-order methods. *International Journal for Numerical Methods in Fluids*, 82(7):398–416, mar 2016. doi: 10.1002/fld.4223.

[63] J. Nitsche. Über ein variationsprinzip zur lösung von dirichlet-problemen bei verwendung von teilräumen, die keinen randbedingungen unterworfen sind. *Abhandlungen aus dem Mathematischen Seminar der Universität Hamburg*, 36(1):9–15, jul 1971. doi: 10.1007/bf02995904.

[64] Ioannis Nompelis, Travis Drayna, and Graham Candler. Development of a hybrid unstructured implicit solver for the simulation of reacting flows over complex geometries. In *34th AIAA Fluid Dynamics Conference and Exhibit*. American Institute of Aeronautics and Astronautics, jun 2004. doi: 10.2514/6.2004-2227.

[65] Assad A. Oberai and John Wanderer. A dynamic multiscale viscosity method for the spectral approximation of conservation laws. *Computer Methods in Applied Mechanics and Engineering*, 195(13-16): 1778–1792, feb 2006. doi: 10.1016/j.cma.2005.05.035.

[66] Stanley Osher and Fred Solomon. Upwind difference schemes for hyperbolic systems of conservation laws. *Mathematics of Computation*, 38(158):339–339, may 1982. doi: 10.1090/s0025-5718-1982-0645656-0.

[67] Renato Paciorri and Aldo Bonfiglioli. Basic elements of unstructured shock-fitting: Results achieved and future developments. In Marcello Onofri and Renato Paciorri, editors, *Shock Fitting: Classical Techniques, Recent Developments, and Memoirs of Gino Moretti*, Shock Wave and High Pressure Phenomena, pages 59–84. Springer International Publishing, 2017. doi: 10.1007/978-3-319-68427-7.

[68] Andreas E. Papoutsakis, Ioannis Nompelis, and John A. Ekaterinaris. Discontinuous Galerkin discretization of chemically reacting flows. In *52nd Aerospace Sciences Meeting*. American Institute of Aeronautics and Astronautics, jan 2014. doi: 10.2514/6.2014-0068.

[69] Per-Olof Persson and Jaime Peraire. Sub-cell shock capturing for discontinuous Galerkin methods. In *44th AIAA Aerospace Sciences Meeting and Exhibit*. American Institute of Aeronautics and Astronautics, jan 2006. doi: 10.2514/6.2006-112.

[70] Jianxian Qiu and Chi-Wang Shu. Hermite WENO schemes and their application as limiters for Runge–Kutta discontinuous Galerkin method: one-dimensional case. *Journal of Computational Physics*, 193(1):115–135, jan 2004. doi: 10.1016/j.jcp.2003.07.026.

[71] Jianxian Qiu and Chi-Wang Shu. A comparison of troubled-cell indicators for Runge-Kutta discontinuous Galerkin methods using weighted essentially nonoscillatory limiters. *SIAM Journal on Scientific Computing*, 27(3):995–1013, jan 2005. doi: 10.1137/04061372x.

[72] Jianxian Qiu, Boo Cheong Khoo, and Chi-Wang Shu. A numerical study for the performance of the Runge–Kutta discontinuous Galerkin method based on different numerical fluxes. *Journal of Computational Physics*, 212(2):540–565, mar 2006. doi: 10.1016/j.jcp.2005.07.011.

[73] W.H. Reed and T.R. Hill. Triangular mesh methods for the neutron transport equation. techreport LA-UR-73-479, Los Alamos Scientific Laboratory, 1973.

[74] P.L. Roe. Approximate Riemann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics*, 135(2):250–258, aug 1997. doi: 10.1006/jcph.1997.5705.

[75] Pierre Schrooyen. *Numerical Simulation of Aerothermal Flows Through Ablative Thermal Protection Systems*. PhD thesis, Université Catholique de Louvain, 2015.

[76] Eiji Shima and Keiichi Kitamura. Parameter-free simple low-dissipation AUSM-family scheme for all speeds. *AIAA Journal*, 49(8):1693–1709, aug 2011. doi: 10.2514/1.j050905.

[77] Chi-Wang Shu. High order WENO and DG methods for time-dependent convection-dominated PDEs: A brief survey of several recent developments. *Journal of Computational Physics*, 316:598–613, jul 2016. doi: 10.1016/j.jcp.2016.04.030.

[78] Chi-Wang Shu and Stanley Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of Computational Physics*, 77(2):439–471, aug 1988. doi: 10.1016/0021-9991(88)90177-5.

[79] Chi-Wang Shu and Stanley Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes, II. *Journal of Computational Physics*, 83(1):32–78, jul 1989. doi: 10.1016/0021-9991(89)90222-2.

[80] Gary A Sod. A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws. *Journal of Computational Physics*, 27(1):1–31, apr 1978. doi: 10.1016/0021-9991(78)90023-2.

[81] Joseph L Steger and R.F Warming. Flux vector splitting of the inviscid gasdynamic equations with application to finite-difference methods. *Journal of Computational Physics*, 40(2):263–293, apr 1981. doi: 10.1016/0021-9991(81)90210-2.

[82] Eitan Tadmor. Convergence of spectral methods for nonlinear conservation laws. *SIAM Journal on Numerical Analysis*, 26(1):30–44, feb 1989. doi: 10.1137/0726003.

[83] Eleuterio F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Springer Berlin Heidelberg, 2009. doi: 10.1007/b79761.

[84] Bram van Leer. Flux-vector splitting for the Euler equations. In *Eighth International Conference on Numerical Methods in Fluid Dynamics*, volume 170, pages 507–512. Springer Berlin Heidelberg, 1982. doi: 10.1007/3-540-11948-5_66.

[85] J. VonNeumann and R. D. Richtmyer. A method for the numerical calculation of hydrodynamic shocks. *Journal of Applied Physics*, 21(3):232–237, mar 1950. doi: 10.1063/1.1699639.

[86] M. J. Vuik and J. K. Ryan. Automated parameters for troubled-cell indicators using outlier detection. *SIAM Journal on Scientific Computing*, 38(1):A84–A104, jan 2016. doi: 10.1137/15m1018393.

[87] Mathea J. Vuik and Jennifer K. Ryan. Multiwavelet troubled-cell indicator for discontinuity detection of discontinuous Galerkin schemes. *Journal of Computational Physics*, 270:138–160, aug 2014. doi: 10.1016/j.jcp.2014.03.047.

[88] Z.J. Wang, Krzysztof Fidkowski, Rémi Abgrall, Francesco Bassi, Doru Caraeni, Andrew Cary, Herman Deconinck, Ralf Hartmann, Koen Hillewaert, H.T. Huynh, Norbert Kroll, Georg May, Per-Olof Persson, Bram van Leer, and Miguel Visbal. High-order CFD methods: current status and perspective. *International Journal for Numerical Methods in Fluids*, 72(8):811–845, jan 2013. doi: 10.1002/fld.3767.

[89] Paul Woodward and Phillip Colella. The numerical simulation of two-dimensional fluid flow with strong shocks. *Journal of Computational Physics*, 54(1):115–173, apr 1984. doi: 10.1016/0021-9991(84)90142-6.

[90] Yidong Xia, Hong Luo, and Robert Nourgaliev. An implicit hermite WENO reconstruction-based discontinuous Galerkin method on tetrahedral grids. *Computers & Fluids*, 96:406–421, jun 2014. doi: 10.1016/j.compfluid.2014.02.027.

[91] Yidong Xia, Xiaodong Liu, Hong Luo, and Robert Nourgaliev. A third-order implicit discontinuous Galerkin method based on a hermite WENO reconstruction for time-accurate solution of the compressible Navier-Stokes equations. *International Journal for Numerical Methods in Fluids*, 79(8):416–435, jun 2015. doi: 10.1002/fld.4057.

[92] Jian Yu, Chao Yan, and Rui Zhao. Assessment of shock capturing schemes for discontinuous Galerkin method. *Applied Mathematics and Mechanics*, 35(11):1361–1374, nov 2014. doi: 10.1007/s10483-014-1875-7.

[93] Jian Yu, Chao Yan, and Zhenhua Jiang. Revisit of dilation-based shock capturing for discontinuous Galerkin methods. *Applied Mathematics and Mechanics*, 39(3):379–394, mar 2018. doi: 10.1007/s10483-018-2302-7.

[94] Hongqiang Zhu, Yue Cheng, and Jianxian Qiu. A comparison of the performance of limiters for Runge-Kutta discontinuous Galerkin methods. *Advances in Applied Mathematics and Mechanics*, 5(03):365–390, jun 2013. doi: 10.4208/aamm.2012.m22.

[95] Jun Zhu and Jianxian Qiu. Hermite WENO schemes and their application as limiters for Runge-Kutta discontinuous Galerkin method, III: Unstructured meshes. *Journal of Scientific Computing*, 39(2):293–321, jan 2009. doi: 10.1007/s10915-009-9271-7.

[96] Jun Zhu, Xinghui Zhong, Chi-Wang Shu, and Jianxian Qiu. Runge-Kutta discontinuous Galerkin method with a simple and compact hermite WENO limiter on unstructured meshes. *Communications in Computational Physics*, 21(03):623–649, feb 2017. doi: 10.4208/cicp.221015.160816a.

A

Exact solver for the Riemann problem

This appendix presents the solver used to obtain an exact solution to the Sod problem by sampling the domain at each time step. The method follows the strategy from Toro [83].

A.1. The Riemann problem

The Riemann problem models a tube in one dimension with a diaphragm in the middle. The gas on the left and the right sides of the diaphragm can have different properties. At time $t = 0$, the diaphragm is removed. Assuming a simple configuration in which initially the gas is at rest on both sides, the density is identical on both sides but the pressure is higher on the left side than on the right, the solution will develop as shown in Figure A.1. After $t = 0$, four regions appear separated by three waves. A shock moving to right, an expansion wave moving to the left, and a contact discontinuity separating the flow between the expansion and the shock in two regions.

The state vector W takes different sets of values depending on the region it is in. Before the shock or the expansion wave, that is in regions designated by W_L and W_R, the state vector simply takes the initial conditions. After these two waves, in the regions designated by W_L^* and W_R^*, the velocity and pressure take the same value on both sides of the contact discontinuity. However, this is not the case for the density. In the expansion fan, the properties smoothly vary from their values in W_L to their new state W_L^*. The next sections discuss the solver implementation of the Riemann problem just described.

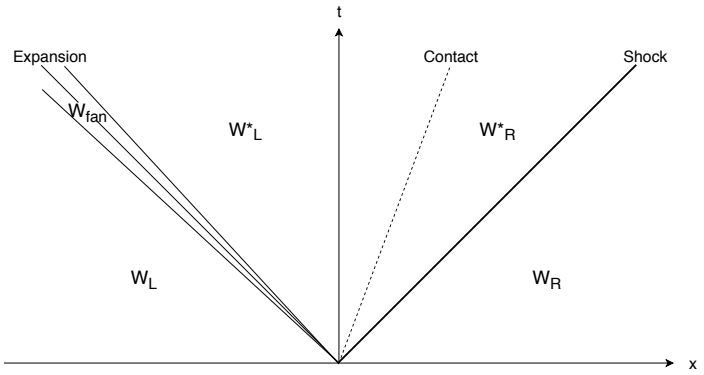


Figure A.1: An x-t diagram of the Riemann problem with a right shock and left expansion wave

A.2. Inputs and initial computations

The solver takes as input the velocity u, the density ρ and pressure P on the left and right states denoted with subscripts L and R, respectively. These form the state vectors W_L and W_R. The speed of sound is computed using

$$c_{L/R} = \sqrt{\gamma \frac{P_{L/R}}{\rho_{L/R}}}, \tag{A.1}$$

where γ is the ratio of specific heats and is equal to 1.4 for air. The pressure and velocity in the regions on both sides of the contact discontinuity are then computed. These are denoted with a $*$ superscript and do not change in time. The pressure P for which Equation (A.2) is satisfied defines P^*:

$$f_L + f_R + (u_R - u_L) = 0, \tag{A.2}$$

where

$$f_L = \begin{cases} (P - P_R)\sqrt{\frac{A_L}{P+B_L}} & \text{for} \quad P > P_L \\ 2\frac{a_L}{\gamma-1}\left(\frac{P}{P_L}\right)^{\left(\frac{\gamma-1}{2\gamma}-1\right)} & \text{otherwise} \end{cases} \tag{A.3}$$

$$f_R = \begin{cases} (P - P_R)\sqrt{\frac{A_R}{P+B_R}} & \text{for} \quad P > P_R \\ 2\frac{a_R}{\gamma-1}\left(\frac{P}{P_R}\right)^{\left(\frac{\gamma-1}{2\gamma}-1\right)} & \text{otherwise} \end{cases} \tag{A.4}$$

and

$$A_{L/R} = \frac{2}{(\gamma+1)\rho_{L_R}} \tag{A.5}$$

$$B_{L/R} = \frac{\gamma-1}{(\gamma+1)P_{P_{L/R}}}. \tag{A.6}$$

The value of P^* is found using the $fzero$ function in MATLAB, with initial guess the average of P_L and P_R. One can then find u^* with

$$u^* = 0.5(u_R + u_L) + 0.5(f_R - f_L). \tag{A.7}$$

The solver then enters a double loop, where it finds the exact solution for the variables ρ, u and P at each point with coordinates (x, t).

A.3. Solution sampling

At each of the space-time points, the solver uses a tree to find where on Figure A.1 it is and applies the corresponding equations. The tree is shown in Figure A.2, for the problem on the left side of the contact discontinuity. The first step to navigate the tree is to define $S = x/t$, where x is the position in the domain and t is the time.

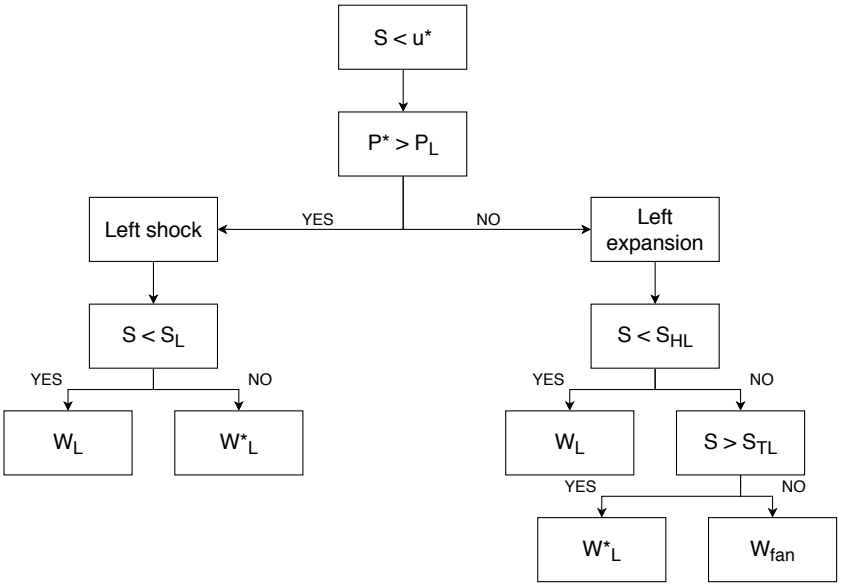


Figure A.2: Solution sampling procedure at point (x,t)

$S < u^*$

If $S < u^*$, the point is in the left side of the contact discontinuity. There can be a shock or an expansion.

If $P^* > P_L$: left shock

To determine on which side of the shock the point is, the first step is to compute the shock speed through:

$$S_L = u_L - c_L\sqrt{\frac{\gamma+1}{2\gamma}\frac{P^*}{P_L} + \frac{\gamma-1}{\gamma+1}}. \tag{A.8}$$

If $S < S_L$, the point is on the left of the shock thus the solution simply takes the initial conditions on the left state. That is $\rho = \rho_L$, $u = u_L$ and $P = P_L$. Instead, if $S > S_L$, the point will take the post shock solution, which corresponds to star values. The velocity u^* and pressure p^* are already known, the density is given by

$$\rho^* = \rho_L \frac{\frac{\frac{P^*}{P_L}+\gamma-1}{\gamma+1}}{\frac{\gamma-1}{\gamma+1}\frac{P^*}{P_L}+1} \tag{A.9}$$

If $P^* < P_L$: left expansion wave

The expansion wave has three possible cases. The point can be before, after, and inside the expansion. Therefore, the head and tail of the expansion need to be found. The head and tail speeds are given by

$$S_{HL} = u_L - c_L \tag{A.10}$$
$$S_{TL} = u^* - c_L^*, \tag{A.11}$$

where

$$c_L^* = c_L\left(\frac{P^*}{P_L}\right)^{\frac{\gamma-1}{2\gamma}}. \tag{A.12}$$

If $S < S_{HL}$, the solution takes the initial conditions ρ_L, u_L, P_L. If $S > S_{TL}$, the solution takes the post expansion wave values. Again, u^* and P^* are known but ρ^* is unknown. It can be computed with

$$\rho_L^* = \rho_L\left(\frac{P^*}{P_L}\right)^{\frac{1}{\gamma}}. \tag{A.13}$$

In the last case, the point is in the expansion fan itself and all quantities need to be computed. They are given by:

$$\rho = \rho_L \left(\frac{2}{\gamma+1} + \frac{(\gamma-1)c_L}{\gamma+1}(u_L - S) \right)^{\frac{2}{\gamma-1}}, \tag{A.14}$$

$$u = \frac{2}{\gamma+1} \left(c_L + \frac{\gamma-1}{2} u_L + S \right), \tag{A.15}$$

$$P = P_L \left(\frac{2}{\gamma+1} + \frac{\gamma-1}{(\gamma+1)c_L}(c_L - S) \right)^{\frac{2\gamma}{\gamma-1}}. \tag{A.16}$$

The same procedure is applied for $S > u^*$, one only needs to replace the subscripts L by R to obtain the corresponding equations.