Construction of very high order residual distribution schemes for steady inviscid flow problems on hybrid unstructured meshes

Remi Abgrall, Adam Larat, Mario Ricchiuto

INRIA

Construction of very high order residual distribution schemes for steady inviscid flow problems on hybrid unstructured meshes

Rémi Abgrall — Adam Larat — Mario Ricchiuto

N° 7236

Rapport de recherche

Construction of very high order residual distribution schemes for steady inviscid flow problems on hybrid unstructured meshes

Rémi Abgrall , Adam Larat , Mario Ricchiuto

Abstract: In this paper we consider the very high order approximation of solutions of the Euler equations. We present a systematic generalization of the Residual Distribution method of [5] to very high order of accuracy, by extending the preliminary work discussed in [18] to systems and hybrid meshes. We present extensive numerical validation for the third and fourth order cases with Lagrange finite elements. In particular, we demonstrate that we an both have a non oscillatory behavior, even for very strong shocks and complex flow patterns, and the expected accuracy on smooth problems.

Key-words: Very high order schemes for compressible fluid mechanics, hybrid unstructured meshes, non oscillatory schemes.

Développement de schémas distribuant ;e résidu d'ordre très élevés pour le calcul d'écoulements compressible sur des maillages non structurés hybrides

Résumé : Dans ce rapport, nous considérons le problème de l'approximation des équations d'Euler aux moyens de schémas d'ordre très élevés. Nous présentons une généralisation systématique des schémas décrits dans [5] permettant de construire des schémas d'ordre (très) élevé utilisant des maillages non structurés hybrides. On montre que le schéma obtenu est stable, même dans le cas de d'écoulements compliqués, et atteint effectivement la précision recherchée sur des solutions régulières.

Mots-clés : Schémas compacts d'ordre élevé pour la mécanique des fluides compressibles, maillages non structurés hybrides, méthodes non oscillantes

Contents

1   Introduction

In the recent years, there has been a strong effort to develop robust and higher order (> 2) schemes for hyperbolic equations, such as the Euler equations, on unstructured grids.

Examples are the ENO/WENO schemes [1, 2] and the Discontinuous Galerkin schemes [3]. In the ENO/WENO case, the equations are approximated by a finite volume scheme where the entries of the flux are evaluated by a high order reconstruction polynomial. The latter is obtained from the cell-data that are interpreted as approximation of the average value of the solution on control volumes. In our opinion, the main drawback of this approach is its algorithmic complexity and the non compact nature of the computational stencil : the average value of the solution in a cell is updated by using its neighbors, and the neighbors of neighbors, and so on, depending on the expected accuracy. The non compactness of the stencil is also a serious drawback for the parallelization of the code.

In the case of DG schemes, the solution is approximated by a local polynomial that is discontinuous across the interface of the elements of the mesh. The solution is updated by means of a local Galerkin form of the equations. The discontinuous nature of the representation requires the use of numerical fluxes when integration by parts is performed on the flux divergence term. The DG approach involves a very local formulation, and it is indeed quite flexible. However, it has one main drawback in the fast growth of the number of degrees of freedom (see also the discussion in [4]).

In this paper we have chosen to use a different strategy based on the Residual Distribution (RD) approach of [5]. In the RD method, the stencil is very local, as in DG, but the number of degrees of freedom grows less quickly. The price to pay is to impose the continuity of the approximation (see however [6, 7]), as in standard finite element methods. Indeed, the RD schemes can be seen as finite elements where the test functions may depend on the solution. This class of scheme is having a growing interest (see [8, 9, 10, 11, 12, 13, 14, 15, 5, 16, 17], etc.). Most of the existing work, however, is limited to second order of accuracy, with the exception of the work discussed in [15, 14, 17], and, more recently, in [18]. In this paper, we extend the preliminary results presented in the last reference by discussing their application to the case of the Euler equation, and by presenting an extensive numerical evaluation of the performance of the schemes.

The structure of the paper is as follows. In section §3 we recall the construction of very high order RD schemes, following the preliminary work presented in [18] that we extend here to hybrid meshes. Starting from the general form of the RD discretization, we introduce the conditions leading to very high order of accuracy and monotonicity, and finally present the basic construction used in the paper. Some scalar numerical tests are also discussed to demonstrate the validity of the approach described. The extension of the schemes to hyperbolic systems, and in particular to the Euler equations, is the object of section §4. All the discretization steps described in section §3 are revisited and details concerning the implementation are given. An extensive numerical validation is

presented in section §5. The paper is ended with some conclusive remarks, a summary of the future and ongoing developments of the work presented here.

2   Mathematical problem

We are interested in the numerical approximation of steady hyperbolic problems of the form

$$\text{div } \mathbf{f}(\mathbf{u}) = S(\mathbf{u}) \tag{1a}$$

which are defined on an open set $\Omega \subset \mathbb{R}^d$, $d = 2, 3$ with weak Dirichlet boundary conditions,

$$u = g. \tag{1b}$$

defined on the inflow boundary[1]

$$\partial\Omega^- = \{x \in \partial\Omega, \vec{n} \cdot \nabla_{\mathbf{u}}\mathbf{f} < 0\}.$$

In (1), the vector of unknown $\mathbf{u}$ belongs to $\mathbb{R}^p$, and the flux $\mathbf{f}$ is

$$\mathbf{f} = (f_1, \ldots, f_d).$$

In (1a), S is a source term which here only depends on the unknown $\mathbf{u}$.

The main target example we are interested in is the system of the Euler equations with the vector of unknown

$$\mathbf{u} = (\rho, \rho\,\vec{u}, E)^T$$

where ρ is the density, $\vec{u}$ is the local flow speed, and E is the total energy. In the particular case $d = 2$, setting $\vec{u} = (u_1, u_2)^T$, the flux can be written as

$$f_1 = \begin{pmatrix} \rho u_1 \\ \rho u_1^2 + p \\ \rho u_1 u_2 \\ u_1(E + p) \end{pmatrix}, \qquad f_2 = \begin{pmatrix} \rho u_2 \\ \rho u_1 u_2 \\ \rho u_2^2 + p \\ u_2(E + p) \end{pmatrix}. \tag{2}$$

The system is closed by an equation of state that relates the pressure p to $\mathbf{u}$. Here we assume a perfect gas equation of state,

$$p = (\gamma - 1)\left(E - \tfrac{1}{2}\rho||\vec{u}||^2\right)$$

with $\gamma = 1.4$.

---

[1] $\vec{n}$ is the inward normal.

3 Very high order residual distribution: general principles and the scalar case

3.1 Introduction : discrete unknowns and discrete equations

Let τ_h denote a tessellation of the spatial domain Ω. In this paper τ_h is assumed to be composed of triangles and quads in $2D^2$. A generic element is denoted by K. Denote by n_t the number of elements of the mesh. The mesh parameter h denotes the maximum radius of the outer circles of the elements $K \in \tau_h$. The vertices of the mesh are denoted by $\{M_i\}_{i=1,...,n_s}$. When there is no ambiguity, we denote the vertices of an element K by $1, \ldots, n_d^K$.

In our approach, the discrete unknowns are a set of local values of the solution in some mesh locations, such as *e.g.* the vertices M_i, edge mid-points etc., etc. These unknowns are referred to as the Degrees of Freedom (DOF). Denote by $\{\sigma_l\}_{l=1....,n_{dof}}$ the list of degrees of freedom. In the case of a second order RD scheme, the DOF are the vertices of the mesh, that is : $\sigma_l = M_l$, $\forall l$. To construct a higher order accurate RD scheme, there are two options :

1. The contribution to the discrete equation of a DOF σ_l in a generic element K is obtained by using information outside K. This option has been followed in [19], and in [20, 14]. In this case, the compactness of the computational stencil is reduced, with the main drawback of an increased algorithmic complexity, especially when more than third order of accuracy is sought for.

2. Discrete equations are written in an element by element fashion, without using any input outside each element. Naturally, in this case, additional DOFs need to be stored in each element, in order to be able to increase the accuracy. This is the approach followed *e.g.* in [21, 18].

Here, following our initial work [18], we use a local higher order polynomial interpolation allowing to keep the local element-by-element structure of the RD formulation. Several ways of obtaining continuous k-th degree polynomials exist. In this paper, we will focus on the case of standard $\mathbb{P}^k$ and $\mathbb{Q}^k$ Lagrange elements defined as follows :

- Quadratic interpolation : the DOFs are the solution values in the vertices and the edges mid–points. This yields $3 + 3$ points per triangle in 2D and $4 + 6$ points per tetrahedron in 3D. For a quadrangle, we need to add the centroid, leading to $4 + 5$ points per elements in 2D. The 3D case would need 27 DOFs per element.

---

[2]We have obtained results in 3D, not reported in this paper. We have not yet considered the case of hybrid meshes in 3D, even though our method should extend without problems to hex, prisms and pyramids. This will be done in a future work.

- Cubic interpolation: in the 2D case, the DOF are the vertices, 2 points per edge (which with the vertices form three segments of equal length), and the centroid, *i.e.* $3 + 2 \times 3 + 1$ DOF per element. In 3D case, the DOF are the 4 vertices, 2 DOF per edge, and the centroid of each face, i.e. $4 + 6 \times 2 + 4 = 20$ DOFs. The case of quadrangle elements leads to 16 DOF per elements, the 3D case would need 64 DOFs per elements.

- etc.

Note that the continuity of the standard Lagrange elements requires that all the DOF on element boundaries are shared by neighboring elements.

As a consequence, in the triangular/tet case, we can count the total number of DOF in terms of the number of vertices, edges, faces (in 3D) and elements,

- in the 2D case, we have

  - Quadratic : $n_s + n_{edge}$ DOFs,
  - Cubic : $n_s + 2n_{edge} + n_t$ DOFs.

- in the 3D case, if in addition n_{face} is the number of faces, we have

  - Quadratic : $n_s + n_{edge}$ DOFs,
  - Cubic : $n_s + 2n_{edge} + n_{face}$ DOFs.

  Thanks to the Euler formula, it is possible to give, for a regular triangulation, an estimate of the asymptotic behavior of the global number of DOF. It is known that in 2D, we have $n_{edge} \approx 3n_s$ and $n_t \approx 2n_s$ and in 3D, $n_{edge} \approx 7n_s$, $n_{face} \approx 10n_s$ and $n_t \approx 6n_s$. On Table 1, we have reported the asymptotic number of degrees of freedom with respect to the dimension and the degree of interpolation. For the sake of comparison, we have also given the same parameters in the case of a discontinuous approximation, as the one used in DG schemes. It is clear that the continuous approximation requires a much smaller number of DOF to yield the same polynomial representation, but the number of DOF increase more rapidly for continous approximations than discontinous ones. Both cases are asymptotically similar. The same conclusion also holds for the quad/hex.

Once we have established what our discrete unknowns are, we have to provide each of them with a discrete equation. We distinguish two cases.

1. in the case of an internal DOF σ, a residual distribution scheme for (1) reads

$$\text{for all } \sigma \in \tau_h, \qquad \sum_{T \ni \sigma} \Phi_\sigma^K = 0 , \qquad (3)$$

where the split residuals Φ_σ^K in (3) must satisfy the following conservation constraint

$$\text{for any } K, \qquad \sum_{\sigma \in K} \Phi_\sigma^K = \oint_{\partial K} \mathbf{f}^h(\mathbf{u}^h) \cdot \vec{n} dl - \int_K S^h(\mathbf{u}^h) dx := \Phi^K \quad (4)$$

where $\mathbf{f}^h(\mathbf{u}^h)$ and $S^h(\mathbf{u}^h)$ are high order accurate approximations of the flux $\mathbf{f}(\mathbf{u})$ and the source term $S(\mathbf{u})$. Natural choices are: the Lagrange interpolant of $\mathbf{f}(\mathbf{u})$ at the degrees of freedom defining $\mathbf{u}^h$, or the true flux evaluated for $\mathbf{u}^h$.

2. if σ is a DOF lying on the boundary of Ω, the equation for σ has to take into account the boundary conditions. Let Γ be any edge/face of the inflow boundary of Ω. We consider a numerical flux $\mathcal{F}$ which depends on the boundary condition $\mathbf{u}_-$, the inward normal $\vec{n}$ and the local state $\mathbf{u}^h$. Then we define boundary residuals Φ_σ^Γ which satisfy the following conservation relation

$$\text{for any } \Gamma \subset \partial\Omega, \qquad \sum_{\sigma\in\Gamma} \Phi_\sigma^\Gamma = \int_{\partial\Gamma} \left(\mathcal{F}(\mathbf{u}^h, \mathbf{u}_-, \vec{n}) - \mathbf{f}^h(\mathbf{u}^h)\cdot\vec{n} \right) dl := \Phi^\Gamma,$$
(5)

At this point we can write for an arbitrary DOF on $\partial\Omega$:

$$\text{for all } \sigma \in \partial\Omega, \qquad \sum_{K\ni\sigma} \Phi_\sigma^K + \sum_{\Gamma\subset\partial\Omega^-, \Gamma\ni\sigma} \Phi_\sigma^\Gamma = 0.$$
(6)

Then following [21], it is easy to show that if the sequence $\mathbf{u}^h$ is bounded in L^∞ when $h \to 0$, and if there exists $\mathbf{v}$ such that $\mathbf{u}^h \to \mathbf{v}$ when $h \to 0$, then $\mathbf{v}$ is a weak solution of (1). One essential ingredient of the proof is the continuity of the interpolant across edges. One can however alleviate this constraint, and define RD schemes on discontinuous elements, see [22, 6, 7] for the second order case. Additional constraints, such as the satisfaction of an entropy inequality, could be set but this will not be considered in this paper.

Remark 3.1 (Numerical quadrature). *Before going further, let us make a remark concerning the notation, and the definition of the element and boundary edge residuals Φ^K, and Φ^Γ, respectively. The definitions (4) and, (5) a priori, need exact integration of the discrete flux and source. However, in practice numerical quadrature is more often implemented. In this case, we replace (4) and (5) by*

$$\text{for any } K, \qquad \sum_{\sigma\in K} \Phi_\sigma^K = \sum_{e\in\partial K} |e| \sum_{p=1}^{G_f} \omega_p \mathbf{f}^h(\mathbf{u}^h(x_p))\cdot\vec{n}_e - |K| \sum_{p=1}^{G_v} \omega_p S^h(\mathbf{u}^h(x_p)) := \widetilde{\Phi^K}.$$
(7a)

having denoted by e the generic edge (face in 3d) of K, and

$$\text{for any } \Gamma \subset \partial\Omega^-, \qquad |\Gamma| \sum_{p=1}^{G_f} \omega_p \left(\mathcal{F}(\mathbf{u}^h(x_p), \mathbf{u}_-(x_p), \vec{n}_\Gamma) - \mathbf{f}^h(\mathbf{u}^h(x_p))\cdot\vec{n}_\Gamma \right) := \widetilde{\Phi^\Gamma}$$
(7b)

where the G_f and G_v denote the number of face and volume Gauss points used in the numerical quadrature. The choice of G_f and G_v, viz of the quadrature formulas used in practice, should not degrade the accuracy of the discretization. The constraints on the numerical quadrature are illustrated hereafter.

3.2   Accuracy constraints

In the previous section we have introduced the general abstract form of our RD discretization. This formulation involves integrals of numerical approximation of the fluxes (and of the source term) based on the $\mathbb{P}^k$ and $\mathbb{Q}^k$ Lagrange approximation of the unknown $\mathbf{u}^h$. These integrals are in practice evaluated numerically, and replaced eventually by the quadrature integrals of equations (7a) and (7b).

The accuracy obtained in practice is of course dependent on the type of quadrature used in the implementation of the schemes. In order to characterize this dependence, we follow the truncation error analysis of [21]. Following the last reference, one can show that scheme (3), (6), (4) and (5) satisfies, for any $\varphi \in C_0^1(\Omega)^3$, the following truncation error

$$
\begin{aligned}
\mathcal{E}(\mathbf{w}^h, \varphi_h) = \sum_{\sigma \in \Omega} \varphi(\sigma) &\left(\sum_{K \ni \sigma} \Phi_\sigma^K + \sum_{\Gamma \subset \partial \Omega^-, \Gamma \ni \sigma} \Phi_\sigma^\Gamma \right) \\
= \int_\Omega &\left(\operatorname{div} \mathbf{f}^h(\mathbf{w}^h) - S^h(\mathbf{w}^h) \right) \varphi_h(x)\, dx + \sum_{K \subset \Omega} \frac{1}{\#\{\sigma \in K\}} \sum_{\sigma, \sigma' \in K} \left(\varphi(\sigma) - \varphi(\sigma') \right) \left(\Phi_\sigma^K - \Phi_\sigma^{K,c} \right) \\
+ \int_{\partial \Omega} &\left(\mathcal{F}(\mathbf{w}^h, \mathbf{w}_-, \vec{n}) - \mathbf{f}^h(\mathbf{w}^h) \cdot \vec{n} \right) \varphi_h(x)\, dl + \sum_{\Gamma \subset \partial \Omega} \frac{1}{\#\{\sigma \in \Gamma\}} \sum_{\sigma, \sigma' \in \Gamma} \left(\varphi(\sigma) - \varphi(\sigma') \right) \left(\Phi_\sigma^\Gamma - \Phi_\sigma^{\Gamma,c} \right) \\
= - \int_\Omega & \nabla \varphi_h(x) \cdot \mathbf{f}^h(\mathbf{w}^h) + \int_{\partial \Omega} \varphi_h(x) \mathbf{f}^h(\mathbf{w}^h) \cdot \vec{n}\, dl - \int_\Omega \varphi_h(x) S^h(\mathbf{w}^h)\, dx \\
+ \int_{\partial \Omega} &\left(\mathcal{F}(\mathbf{w}^h, \mathbf{w}_-, \vec{n}) - \mathbf{f}^h(\mathbf{w}^h) \cdot \vec{n} \right) \varphi_h(x)\, dl \\
& + \sum_{K \subset \Omega} \frac{1}{\#\{\sigma \in T\}} \sum_{\sigma, \sigma' \in K} \left(\varphi(\sigma) - \varphi(\sigma') \right) \left(\Phi_\sigma^K - \Phi_\sigma^{K,c} \right) \\
& + \sum_{\Gamma \subset \partial \Omega} \frac{1}{\#\{\sigma \in \Gamma\}} \sum_{\sigma, \sigma' \in \Gamma} \left(\varphi(\sigma) - \varphi(\sigma') \right) \left(\Phi_\sigma^\Gamma - \Phi_\sigma^{\Gamma,c} \right).
\end{aligned}
\tag{8}
$$

where $\mathbf{w}$ is a classical solution of the problem, $\mathbf{w}^h$ being its $\mathbb{P}^k/\mathbb{Q}^k$ Lagrange approximation, φ_h is the Lagrange interpolant of $\{\varphi(\sigma)\}_\sigma$, $\Phi_\sigma^{K,c}$ and $\Phi_\sigma^{\Gamma,c}$ are the Galerkin residuals

$$
\Phi_\sigma^{K,c} = \int_K \psi_\sigma \left(\operatorname{div} \mathbf{f}(\mathbf{w}^h) - S(\mathbf{w}^h) \right) dx \quad \text{and} \quad \Phi_\sigma^{\Gamma,c} = \int_\Gamma \psi_\sigma \left(\mathcal{F}(\mathbf{w}^h, \mathbf{w}_-, \vec{n}) - \mathbf{f}(\mathbf{w}^h) \cdot \vec{n} \right) dx ,
\tag{9}
$$

and $\psi_\sigma \in \mathbb{P}^k(K)$ or $\mathbb{Q}^k(K)$ is the Lagrange basis function relative to the DOF $\sigma \in K$. Using the fact that everywhere $\operatorname{div} \mathbf{f}(\mathbf{w}) - S(\mathbf{w}) = 0$, the error can be

---

[3] $C_0^1(\Omega)$ is the set of C^1 functions on Ω with a compact support

easily decomposed as [21, 23]

$$
\begin{aligned}
\mathcal{E}(\mathbf{w}^h, \varphi_h) = &-\int_\Omega \nabla \varphi_h(x) \cdot \left(\mathbf{f}^h(\mathbf{w}^h) - \mathbf{f}(\mathbf{w}) \right) && \text{(flux approximation error)} \\
&-\int_\Omega \varphi_h(x) \left(S^h(\mathbf{w}^h) - S(\mathbf{w}) \right) dx && \text{(source approximation error)} \\
&+\int_{\partial\Omega} \left(\mathcal{F}(\mathbf{w}^h, \mathbf{w}_-, \vec{n}) - \mathbf{f}(\mathbf{w}) \cdot \vec{n} \right) \varphi_h(x) dl && \text{(BC approximation error)} \\
&+\sum_{K \subset \Omega} \frac{1}{\#\{\sigma \in K\}} \sum_{\sigma, \sigma' \in K} \left(\varphi(\sigma) - \varphi(\sigma') \right) \left(\Phi_\sigma^K - \Phi_\sigma^{K,c} \right) && \text{(distribution error - interior)} \\
&+\sum_{\Gamma \subset \partial\Omega} \frac{1}{\#\{\sigma \in \Gamma\}} \sum_{\sigma, \sigma' \in \Gamma} \left(\varphi(\sigma) - \varphi(\sigma') \right) \left(\Phi_\sigma^\Gamma - \Phi_\sigma^{\Gamma,c} \right) && \text{(distribution error - boundary).}
\end{aligned}
$$
$$(10)$$

As discussed thoroughly in [21, 23], relation (8) is a consequence of the conservation relations (4) and (5). Following again the last references, we can prove the following result:

Proposition 3.2. *Given a regular enough classical solution* $\mathbf{w}$*, if the residuals evaluated on the* $\mathbb{P}^k, \mathbb{Q}^k$ *interpolant* $\mathbf{w}^h$ *satisfy*

$$
\Phi_\sigma^K(\mathbf{w}^h) = \mathcal{O}(h^{k+d}) \tag{11a}
$$

and

$$
\Phi_\sigma^\Gamma(\mathbf{w}^h) = \mathcal{O}(h^{k+d-1}), \tag{11b}
$$

and if the approximations $\mathbf{f}^h(\mathbf{w}^h)$*, and* $S^h(\mathbf{w}^h)$ *are* $k+1$*-order accurate, then the truncation error satisfies*

$$
|\mathcal{E}(\mathbf{w}^h, \varphi^h)| \leq C(\varphi, \mathbf{f}, \mathbf{w}) \, h^{k+1}.
$$

The constant $C(\varphi, \mathbf{f}, \mathbf{w})$ *depends only on* φ*,* $\mathbf{f}$*, and* $\mathbf{w}$*.*

Proof. See appendix B.   □

A first consequence of the analysis is obtained by noting that, under the hypotheses of proposition 3.2 (see [21, 23] for details) :

$$
\Phi^K(\mathbf{w}^h) = \mathcal{O}(h^{k+d}) \quad \text{and} \quad \Phi^\Gamma(\mathbf{w}^h) = \mathcal{O}(h^{k+d-1})
$$

As a consequence, if there exists a constant (in the scalar case) or a matrix (in the system case) β_σ^T such that

$$
\Phi_\sigma^K = \beta_\sigma^K \Phi^K \tag{12a}
$$

$$
\Phi_\sigma^\Gamma = \beta_\sigma^\Gamma \Phi^\Gamma \tag{12b}
$$

then the condition (11) is fulfilled provided that β_σ^T is uniformly bounded. This gives a design criterion for high order schemes. For historical reasons, RD discretizations that can be written as in (12a)-(12b) are referred to *Linearity Preserving* even-though the interpolant is no longer linear.

In practice, as explained in remark 3.1, one uses of numerical quadrature to evaluate the cell residuals. In this case, linearity preserving RD schemes read (cf. equations (7a) and (7b))

$$\Phi_\sigma^K = \beta_\sigma^K \widetilde{\Phi^K} \tag{13a}$$

$$\Phi_\sigma^\Gamma = \beta_\sigma^\Gamma \widetilde{\Phi^\Gamma} \tag{13b}$$

To maintain the same error level, we see immediately that the constraints on the quadrature formulas used to obtain (7a) and (7b) are the following :

- In (13a), we must have

$$\sum_{e \in \partial T} |e| \sum_{p=1}^{G_f} \omega_p \mathbf{f}^h(\mathbf{u}^h(x_p)) \cdot \vec{n}_e = \oint_{\partial T} \mathbf{f}^h(\mathbf{u}^h) \cdot \vec{n} dl + \mathcal{O}(h^{k+d}) \tag{14a}$$

and

$$|T| \sum_{p=1}^{G_v} \omega_p S^h(\mathbf{u}^h(x_p)) = \int_T S^h(\mathbf{u}^h) dx + \mathcal{O}(h^{k+d}) \tag{14b}$$

- In (13b), we must have for the boundary conditions integrals

$$|\Gamma| \sum_{p=1}^{G_f} \omega_p \left(\mathcal{F}(\mathbf{u}^h(x_p), \mathbf{u}_-(x_p), \vec{n}_\Gamma) - \mathbf{f}^h(\mathbf{u}^h(x_p)) \cdot \vec{n}_\Gamma \right) =$$
$$\int_{\partial \Gamma} \left(\mathcal{F}(\mathbf{u}^h, \mathbf{u}_-, \vec{n}) - \mathbf{f}^h(\mathbf{u}^h) \cdot \vec{n} \right) dl + \mathcal{O}(h^{k+d-1}) \tag{14c}$$

There are of course numerous quadrature formulas that can be used to stay within the error bounds given above. An inferior bound to the polynomial degree that has to be integrated exactly is given by these bounds.

The practical approach used in this work is to reconstruct in each element a flux polynomial based on the Lagrange interpolation of the flux values evaluated at the degrees of freedom. The quadrature points coincide with the DOF, and the quadrature weights are easily computed once and for all. This is equivalent to a quadrature free approach (cf. [24]). We come back to this point in section 5.4 to discuss our actual implementation of the boundary conditions.

3.3 Monotonicity preservation

In this paragraph we consider the issue of guaranteeing the non-oscillatory character of the solution. We make use of the theory of positive coefficient schemes

[25] to design discretizations yielding solutions that verify a discrete maximum principle. To do this, we consider the case in which (1) is a scalar equation for the unknown u, and the homogeneous case $S = 0$. In this setting, all RD schemes can be re-written as

$$\Phi_\sigma^K = \sum_{\sigma' \in T} c_{\sigma\sigma'}(u_\sigma - u_{\sigma'}) \tag{15}$$

so that equation (3) becomes for any σ (and neglecting boundary conditions)

$$\sum_{T \ni \sigma} \sum_{\sigma' \in T} c_{\sigma\sigma'}(u_\sigma - u_{\sigma'}) = 0.$$

In general, the coefficients $c_{\sigma\sigma'}$ depend on the solution, which means that the last expression defines a set of non linear equations that needs to be solved by means of an iterative method. The simplest one is the *Jacobi–like iteration*

$$u_\sigma^{n+1} = u_\sigma^n - \omega_\sigma \left(\sum_{K \ni \sigma} \sum_{\sigma' \in T} c_{\sigma\sigma'}^K(u_\sigma - u_{\sigma'}) \right)^n \tag{16}$$

where ω_σ is a relaxation parameter.

It is easy to verify that if the scheme satisfies the *positivity conditions*

$$\sum_{K \ni \sigma, K' \ni \sigma'} c_{\sigma\sigma'}^K \geq 0 \quad \forall \sigma, \sigma' \qquad \text{and} \qquad 1 - \omega_\sigma \left(\sum_{K \ni \sigma} \sum_{\sigma' \in K} c_{\sigma\sigma'}^K \right) \geq 0 \quad \forall \sigma \tag{17}$$

then the solution verifies the following discrete maximum principle

$$\min_{K \ni \sigma} \min_{\sigma' \in K} u_{\sigma'}^0 \leq u_\sigma^n \leq \max_{K \ni \sigma} \max_{\sigma' \in K} u_{\sigma'}^0 \tag{18}$$

having denoted by u_σ^0 the values of the initial solution in the DOF locations. A more convenient approach for the design of the schemes is to replace conditions (17) by local constraints. To do this, let us define for each DOF the following median dual areas :

$$\mathcal{C}_\sigma^K = \frac{|K|}{n_d} \qquad \text{and} \qquad \mathcal{C}_\sigma = \sum_{K \ni \sigma} \mathcal{C}_\sigma^K$$

Obviously, conditions (17) are met if the following *local positivity* conditions are verified :

$$c_{\sigma\sigma'}^K \geq 0 \, \forall \sigma, \sigma' \in K \text{ and } \forall K \qquad \text{and} \qquad \omega_\sigma \max_{K \ni \sigma} \left[\frac{\mathcal{C}_\sigma}{\mathcal{C}_\sigma^K} \left(\sum_{\sigma' \in K} c_{\sigma\sigma'}^K \right) \right] \leq 1 \, \forall \sigma \tag{19}$$

These conditions do not imply that the iterative scheme (16) is convergent, but only that the discrete maximum principle (18) (*viz.* L^∞-stability) is satisfied. In the rest of the paper, a scheme that verifies conditions (19) is said to be monotonicity preserving.

3.4 Getting high order accuracy and monotonicity preservation

It is known that a scheme that is monotonicity preserving with coefficients $c_{\sigma\sigma'}^K$ independent on the solution cannot satisfy (11). This results is a variant of Godunov's theorem for RD schemes, of which a general proof is been given in [26]. As a consequence of this, a monotonicity and linearity preserving scheme must be non linear. Here we follow [21] to obtain nonlinear schemes verifying both properties.

We start from a monotone first order scheme which residuals are (for $S = 0$)

$$\Phi_\sigma^L = \sum_{\sigma' \in K} c_{\sigma\sigma'}^L (u_\sigma - u_{\sigma'})$$

the super-script ^{L} standing for Low order. By assumption, the coefficients $c_{\sigma\sigma'}^L$ are *all* positive, and, of course,

$$\sum_{\sigma \in K} \Phi_\sigma^L = \Phi^K$$

Then, let Φ_σ^H denote high order residuals, such that

$$\Phi_\sigma^H = \beta_\sigma \Phi^K \qquad \text{with} \qquad \sum_{\sigma \in K} \beta_\sigma^H = 1 \tag{20}$$

By analogy, we introduce the parameters x_σ defined by

$$x_\sigma = \frac{\Phi_\sigma^L}{\Phi^K}$$

for which, thanks to the conservation relation, we also have $\sum_\sigma x_\sigma = 1$.

The next step is to write the formal identity

$$\Phi_\sigma^H = \frac{\Phi_\sigma^H}{\Phi_\sigma^L} \Phi_\sigma^L = \sum_{\sigma'} \frac{\Phi_\sigma^H}{\Phi_\sigma^L} c_{\sigma\sigma'}^L (u_\sigma - u_{\sigma'})$$

and we can see that, starting from Φ_σ^L, we could obtain a monotonicity preserving high order scheme, provided that we satisfy the constraint

$$\frac{\Phi_\sigma^H}{\Phi_\sigma^L} \geq 0 \quad \forall \sigma \in K$$

because then we have for the high order scheme

$$c_{\sigma\sigma'}^H = \frac{\Phi_\sigma^H}{\Phi_\sigma^L} c_{\sigma\sigma'}^L \geq 0$$

All this can be rephrased in terms of x_σs and β_σs.

1. Conservation.

$$\sum_{\sigma \in K} \beta_\sigma = 1 \quad \text{and} \quad \sum_{\sigma \in K} x_\sigma = 1$$

2. Monotonicity preservation.

$$x_\sigma \beta_\sigma \geq 0 \quad \forall \sigma \in K$$

These relations can be interpreted geometrically. Since there is no ambiguity, we can assume that the degrees of freedom can be numbered from 1 to n_d, an we identify the DOF σ to its number ℓ in $[1, \ldots, n_d]$.

Let us consider in $\mathbb{R}^{n_d}$ n_d linearly independent points $\mathcal{S} = \{A_\ell\}_{\ell=1,\cdots,n_d}$. Note they do not have connections with any physical points in the mesh. We can introduce for any point $M \in \mathbb{R}^{n_d}$ its barycentric coordinates $\{\lambda_\ell(M)\}$ with respect to $\mathcal{S}$:

$$M = \sum_{\ell=1}^{n_d} \lambda_\ell(M) A_\ell$$

or equivalently, for any $O \in \mathbb{R}^{n_d}$

$$\overrightarrow{OM} = \sum_{\ell=1}^{n_d} \lambda_\ell(M) \overrightarrow{OA_\ell}.$$

We have by definition $\sum_{\ell=1}^{n_d} \lambda_\ell(M) = 1$. Thus, we can interpret $\{x_l\}$ and $\{\beta_l\}$ as the barycentric coordinates of the points L and H such that

$$L = \sum_{\ell=1}^{n_d} x_\ell A_\ell$$
$$H = \sum_{\ell=1}^{n_d} \beta_\ell A_\ell$$

and the problem becomes to define a mapping onto $\mathbb{R}^{n_d} : L \mapsto H$ such that the constraints $x_\ell \beta_\ell \geq 0$ are true : the advantage of that interpretation is that the conservation properties are automatically satisfied.

There are many solution to that problem, one particularly simple one is an extension of the PSI "limiter" of Struijs :

$$\beta_\ell = \frac{x_\ell^+}{\sum_{\ell'} x_{\ell'}^+}, \qquad x^+ = \max(x, 0). \tag{21}$$

There is no singularity in the formula since

$$\sum_{\ell'} x_{\ell'}^+ = \sum_{\ell'} x_{\ell'} - \sum_{\ell'} x_{\ell'}^- \geq 1.$$

Throughout the paper, we use (21).

3.5 Spurious modes and iterative convergence : a numerical example and a counter example

Before proceeding further with the construction, we consider two numerical examples involving the solution of the advection equation

$$\vec{\lambda} \cdot \nabla u = 0, \qquad x \in \Omega. \tag{22}$$

3.5.1 Case of triangles.

Using the $\mathbb{P}^k$ interpolant in T

$$u^h = \sum_{\sigma \in T} u_\sigma \psi_\sigma,$$

the total residual Φ^T can be written as

$$\Phi^T = \int_T \vec{\lambda} \cdot \nabla u^h \, dx = \sum_{\sigma \in T} u_\sigma \int_T \vec{\lambda} \cdot \nabla \psi_\sigma dx.$$

By analogy with what is done with second order RD schemes [11, 27], we set

$$k_\sigma = \int_T \vec{\lambda} \cdot \nabla \psi_\sigma dx,$$

so that

$$\Phi^T = \sum_{\sigma \in T} u_\sigma k_\sigma. \tag{23}$$

We note that $\sum_{\sigma \in T} k_\sigma = 0$. To construct a nonlinear scheme, we start from the following first order (local) Lax-Friedrich's (LLxF) scheme :

$$\Phi_\sigma^T = \frac{\Phi^T}{n_d} + \alpha_T (u_\sigma - \bar{u}) \tag{24a}$$

with

$$\bar{u} = \frac{\sum_{\sigma' \in T} u_{\sigma'}}{n_d}. \tag{24b}$$

Using (23), the LLxF residual (24a) can be recast in the form (15) with

$$c_{\sigma\sigma'}^T = \frac{k_{\sigma'} - \alpha_T}{n_d}$$

and $c_{\sigma\sigma'}^T \geq 0$ if

$$\alpha_T \geq \max_{\sigma \in T} |k_\sigma|. \tag{25}$$

This first order scheme is extremely dissipative, but this is the one from which we start for two reasons : it is very cheap and simple to code, even for

systems, and it easily generalizes to any order of accuracy (*viz.* polynomial interpolation).

We test the nonlinear limited LLxF scheme obtained by applying (21) to the LLxF scheme (24a) scheme on two simple linear advection problems. On the spatial domain $\Omega = [0, 1]^2$, we take in the first problem

$$\vec{\lambda} = (1, 2)^T \quad \text{and } u(x, y) = \begin{cases} 1 & \text{if } x = 0 \text{ and } y > 0 \\ 0 & \text{if } y = 0 \text{ and } x > 0 \end{cases} \tag{26}$$

The second problem is obtained by setting

$$\vec{\lambda} = (y, -x)^T \quad \text{and} \quad u(x, y) = \begin{cases} \varphi_0(x) & \text{if } y = 0 \\ 0 & \text{otherwise} \end{cases} \tag{27}$$

where

$$\varphi_0(x) = \begin{cases} \cos^2(2\pi x) & \text{if } x \in [0.25, 0.75] \\ 0 & \text{else} \end{cases}$$

The results obtained in the $\mathbb{P}^2$ case are displayed on figure 1. The behavior observed is similar to what has been found, in the second order case, in [5] : discontinuities are approximated without over- or undershoots, however, we observe the appearance of *plateaus* in the numerical results, for both smooth and non-smooth data. This is clearly visible in the plots of the outlet data on figure 1. Other symptoms are : smooth solutions often present a high frequency oscillations (spurious modes), the iterative convergence is poor : after a very quick drop of about two orders of magnitude, the iterative residual stagnates to a constant value. The behavior is the same observed in finite volume schemes when using an over-compressive limiter. As remarked in [5], this behavior is not due to an L^∞ instability : the local maximum principle is satisfied both theoretically and numerically.

As a result of the lack of iterative convergence, equation (3) is not solved exactly but within a $\mathcal{O}(h)$ error which can be easily measured experimentally [5]. Thus the overall accuracy obtained is only that of a first order scheme.

3.5.2   Case of quadrangles.

This example is maybe more illuminating since we can exhibit some of the spurious modes. Again, we consider $\Omega = [0, 1]^2$ which is discretised by uniform quads. The vertices are $x_{i,j} = (\frac{i}{N}, \frac{j}{N})$ $(0 \leq i, j \leq N)$ and the problem writes

$$\frac{\partial u}{\partial x} = 0$$

subjected to boundary conditions on the left side of Ω. Assuming a general LP scheme, we update the solution by

$$u_\sigma^{n+1} = u_\sigma^n - \omega \left(\sum_{K, \sigma \in K} \beta_\sigma^K \Phi^K \right)$$

Two things need to be precised : the boundary conditions and the initial state u^0. On the left boundary (inflow), we impose a check-board like mode, but this is not really essential as we see at the end of the paragraph), i.e.

$$u_\sigma = (-1)^{i_\sigma}$$

where i is the index such that $\mathbf{x}_\sigma = (\frac{i}{N}, 0)$ and $u_\sigma = 0$ is σ is any mid point. The initial condition is defined by

- either as on figure 2-a : we "propagate" the boundary condition along the characteristics of the PDE.

- or as on figure 2-b.

We expect to converge to the first initialization. Let us compute the total residual on $Q = [x_i, x_{i+1}] \times [y_j, y_{j+1}]$. We get

$$\Phi^Q = \int_{y_j}^{y_{j+1}} \int_{x_i}^{x_{i+1}} \frac{\partial u}{\partial x} dx dy = \int_{y_j}^{y_{j+1}} \big(u(x_{i+1}, y) - u(x_i, y)\big) dx.$$

In our case, we have, by symmetry, $u(x_{i+1}, y) = u(x_i, y)$, so that $\Phi^Q = 0$ and $u_\sigma^{n+1} = u_\sigma^n$. This shows that the scheme cannot converge in this case.... Hence, something more must be done !

3.6 Convergent nonlinear schemes

Following our previous work [5, 18], the behavior described in the last section can be corrected by adding to the residuals (13) a term of the form

$$h_K \int_K \left(\vec{\lambda} \cdot \nabla \psi_\sigma \right) \tau \left(\vec{\lambda} \cdot \nabla u^h \right) dx, \qquad \tau > 0. \tag{28}$$

This is a streamline dissipation term, used in SUPG discretizations of hyperbolic problems to suppress the spurious modes of the Galerkin scheme [28]. As its name suggests, this term has a dissipative nature. It does not destroy the formal accuracy of the original discretization. It does not destroy the conservation property (4) because the residuals are now

$$\Phi_\sigma^\star = \Phi_\sigma^H + h_K \int_K \left(\vec{\lambda} \cdot \nabla \psi_\sigma \right) \tau \left(\vec{\lambda} \cdot \nabla u^h \right) dx \tag{29}$$

with Φ_σ^H defined by (20) and clearly,

$$\sum_{\sigma \in K} \Phi_\sigma^\star = \Phi^K$$

because $\sum_{\sigma \in K} \nabla \psi_\sigma = 0$.

In fact, (28) has the good effect of removing the spurious modes that are existing, and of improving the quality of the solution (see [5, 18] for details).

The problem of integral (28) is that its exact evaluation requires the exact integration of a polynomial of degree $2(k-1)$ which is expensive. A better analysis of the structure and the role of the dissipative term helps to reduce substantially its computational cost. We start by rewriting the nonlinear limited scheme as

$$\Phi_\sigma^K = \Phi_\sigma^{K,c} + \Phi_\sigma^K - \Phi_\sigma^{K,c},$$

with $\Phi_\sigma^{K,c}$ still given by (9). Given a function φ, multiply (3) by $\varphi(\sigma)$ and add up all the equations for all the DOF of the mesh. Using the conservation relations, and neglecting boundary conditions, (3) is equivalent to

$$\int_\Omega \varphi^h \operatorname{div} \mathbf{f}^h(u^h)dx + \sum_K q_K(\varphi^h, u^h) = 0 \tag{30a}$$

with

$$q_K(\varphi^h, u^h) = \frac{1}{n_d!} \sum_{\sigma,\sigma' \in K} (\varphi(\sigma) - \varphi(\sigma')) \left(\beta_\sigma^K \Phi^K - \Phi_\sigma^{K,c} \right) \tag{30b}$$

with $\Phi_\sigma^{K,c}$ the Galerkin residuals (9). The modification introduced in [5] amounts to adding term (28) to the quadratic form q_K. The problem is to know under which conditions the resulting scheme is dissipative, keeps the original accuracy, and preserves the non oscillatory behavior of (30a).

The most natural way of proceeding is to replace (28) by an approximation obtained by means of a "quadrature" formula :

$$d_K(\varphi^h, u^h) = |K| \sum_{x_{\text{quad}}} \omega_{\text{quad}} \left[\left(\vec{\lambda} \cdot \nabla \varphi_\sigma \right)(x_{\text{quad}}) \, \tau(x_{quad}) \, \left(\vec{\lambda} \cdot \nabla u^h \right)(x_{\text{quad}}) \right] \tag{31}$$

such that

$$(\varphi^h, u^h) \mapsto \int_\Omega \varphi^h \operatorname{div} \mathbf{f}^h(u^h)dx + \sum_K \left(q_K(\varphi^h, u^h) + \theta_K h_K d_K(\varphi^h, u^h) \right)$$

is dissipative. We have put quadrature between quotes because as we see later in the text, these "quadrature" formula do not need to be consistent as approximation of integrals.

Instead of studying the overall behavior of the scheme, we can make use of the compact nature of the discretization and focus our attention on the quadratic form

$$(\varphi^h, u^h) \mapsto q_K(\varphi^h, u^h) + \theta_K h_K d_K(\varphi^h, u^h)$$

Here, h_K is a the radius of the circle/sphere circumscribed to K. The parameter θ_K has the role of activate the extra dissipation in smooth regions, while deactivating it across discontinuities, where the original scheme has no under/overshoots. Hence, we should have $\theta_K \approx 1$ in correspondence of smooth variations of the solutions, while $\theta_K \ll 1$ across discontinuities.

Concerning accuracy on smooth solutions, whatever "quadrature" formula we use, and thanks to the term $\theta_K h_K$ in front of d_K, if u^h is the interpolant of

a smooth enough function such that $\vec{\lambda} \cdot \nabla u = 0$, then one can show (see [18] for details) that

$$\left| \theta_K h_K d_K(\varphi^h, u^h) \right| \leq C(u) \|\nabla \varphi\| \, h_K^{k+d+1},$$

so that the *formal* accuracy is not spoiled. Lastly, to ensure convergence, one can ask the bilinear form

$$(\varphi^h, u^h) \mapsto d_K(\varphi^h, u^h)$$

to be positive definite whenever $\left| \vec{\lambda} \cdot \nabla u \right| > 0$. In particular, following [5, 18], we will require that $\omega_{\text{quad}} > 0$ and that $d_K(\varphi^h, u^h)$ is positive definite whenever $\vec{\lambda} \cdot \nabla u^h \neq 0$.

This amounts at requiring a number of quadrature points allowing an exact representation of the term $\vec{\lambda} \cdot \nabla u^h$ over K. Hence, one quadrature point is enough for $k = 1$, 3 points are needed for $k = 2$, 6 for $k = 3$ and so on : one has to be exact on polynomials of degree $k - 1$. In the 3D case, one quadrature point is enough for $k = 1$, 3 for $k = 2$, and so on.

There is no need for the "quadrature" formula to be consistent with the integral

$$\int_K (\vec{\lambda} \cdot \nabla \varphi) \, \tau \, (\vec{\lambda} \cdot \nabla u) dx.$$

We choose these points so that the discrete formula is independent of the numbering of the mesh points. In our examples, and for triangles and tets, we choose the vertices of T for $k = 2$; we add to these points the mid edge points for $k = 3$: since these points are degrees of freedom, the additional cost is minimized. In the case of quads, we also choose the vertices for $k = 2$. The weights ω_{quad} used in (31) are $\omega_{\text{quad}} = 1/\#\{\text{quad points}\}$.

In the following, we denote

$$\Psi_\sigma^K = |K| \sum_{x_{\text{quad}}} \omega_{\text{quad}} \left[(\vec{\lambda} \cdot \nabla \psi_\sigma)(x_{\text{quad}}) \, \tau(x_{quad}) \, (\vec{\lambda} \cdot \nabla u^h)(x_{\text{quad}}) \right]. \tag{32}$$

Concerning the choice of the parameter θ_K, we have used in practice the following definition ;

$$\theta_K = 1 - \max_{\sigma \in T} \left[\max_{T' \ni \sigma} \left(\frac{|u_\sigma - \bar{u}_K|}{|u_\sigma| + |\bar{u}_K| + \varepsilon} \right) \right] \tag{33}$$

with ε of the order of machine zero and $\bar{u}_K = \left(\sum_{\sigma \in K} u_\sigma \right)/n_d$. Typically, $\theta = \mathcal{O}(h_K)$ in a smooth region and $\theta \equiv 1$ in a discontinuity. The relation (33) depends on values of u outside K, thus it seems that the formula is not compact. Indeed this is true, but from an algorithmic point of view, what is important is that the implementation can be made compact.

3.7   Summary of the final scheme for scalar problems.

The algorithm 1 summarizes the main operations performed for an explicit implementation, and show that the compactness of the method is not destroyed. This can easily be generalized to other type of iterative schemes.

Algorithm 1 Sketch of the explicit implementation of the schemes. The evaluation of θ_K (cf. equation (33)) is kept compact by updating and swapping the monitors θ_σ and $\tilde{\theta}_\sigma$.

1: Initialize by $\theta_\sigma = 1$ for all DOFs.
2: **for** Do for $k = 1$ to k_{max} (maximum number of iterations) **do**
3:     Set $\tilde{\theta}_\sigma = 0$ for each σ and $Res_\sigma = 0$
4:     **for** For each K **do**
5:         evaluate quantities β_σ^K, Φ^K, h_K, Ψ_σ^K, and θ_K, with

$$\theta_K = 1 - \max_{\sigma \in K} \theta_\sigma$$

6:         evaluate

$$\Phi_\sigma^K = \beta_\sigma^K \Phi^K + \theta_K h_K \Psi_\sigma^K, \qquad (34a)$$

7:         evaluate

$$\xi_\sigma = \max(\tilde{\theta}_\sigma, \frac{|u_\sigma - \bar{u}_K|}{|u_\sigma| + |\bar{u}_K| + \varepsilon}) \qquad (34b)$$

8:         set $\tilde{\theta}_\sigma = \xi_\sigma$,
9:         update

$$Res_\sigma = Res_\sigma + \Phi_\sigma^K.$$

10:     **end for**
11:     Swap : $\theta_\sigma = \tilde{\theta}_\sigma$,
12:     Update : $u_\sigma^{n+1} = u_\sigma^n - \omega_\sigma Res_\sigma$
13: **end for**

For any K, the total residual is defined by (7). The split residuals Φ_σ^K are defined by (13) and β_σ^K by (21). The term Ψ_σ^K is defined by (32). The algorithm solves (3) for the interior degrees of freedom and (4) for the boundary ones.

When the local Lax Friedrichs scheme is used as a first order building block, as for all the results of this paper, the scheme is denoted by LLxFf (for Local Lax Friedrichs filtered).

4   Numerical illustrations for the scalar case.

We start again with the advection problem with initial states and advection speeds defined by (26) and (27). The results obtained for $\mathbb{P}^2$ interpolation when adding the term (32) (cf. also equation (34a)) are displayed on figure 3. The

left picture shows that, for the discontinuous solution of problem (26), we do not get any spurious oscillations. The right picture instead shows, for problem (27), the benefice effect of the extra term in smoothing the contours that now are perfectly circular. We have also run a grid refinement study on this problem using $\mathbb{P}^2$ and $\mathbb{P}^3$ approximations. The results are summarized on table 3. The least squares slopes obtained confirm the expected convergence rates.

To better visualize the improvement in the solution when going from $\mathbb{P}^1$ to $\mathbb{P}^2$ spatial interpolation, we consider, on the spatial domain $[0,2] \times [0,1]$, the solid body rotation of the inlet profile $u(x) = \sin(10\pi x)$. In this case the advection speed is set to $\vec{\lambda} = (y, 1-x)$. The contours of the numerical solutions obtained are reported on figure 4. Note that the $\mathbb{P}^1$ run has been performed on the mesh obtained by sub-triangulating the $\mathbb{P}^2$ mesh so that exactly the same number of DOF is used in the two cases. The dramatic improvement brought by the $\mathbb{P}^2$ approximation is clearly visible in the contour plots, and also in the outlet profiles reported on figure 5.

We have also run the linear advection test case (26)-(27) on hybrid mesh made of triangles and non orthogonal quadragles. The details of the meshes, in term of triangles and quad, are described in table 2.

The figure 6 shows the errors done with second, third and fourth order schemes. We recover the expected order. We have also compared these errors with the results obtained when all the quad are cut into two triangles. It appears that the hybrid results are a little bit more accurate. This is not however a definite advantage.

We test further the definition of the smoothness sensor θ_T by solving the 2D Burgers's problem

$$\frac{\partial u}{\partial y} + \frac{1}{2}\frac{\partial u^2}{\partial x} = 0 \quad \text{if } x \in [0,1]^2$$
$$u(x,y) = 1.5 - 2x \quad \text{on } y = 0.$$

The exact solution consists in a fan that merges into a shock which foot is located at $(x,y) = (3/4, 1/2)$. More precisely, the exact solution is

$$u(x,y) = \begin{cases} \text{if } y \geq 0.5 & \begin{cases} -0.5 & \text{if } -2(x-3/4) + (y-1/2) \geq 0 \\ 1.5 & \text{else} \end{cases} \\ \text{else} & \max\left(-0.5, \min\left(1, 5, \frac{x-3/4}{y-1/2}\right)\right) \end{cases}$$

The results obtained on the mesh of figure 1 are displayed on figure 7. For the sake of comparison, we give the second and third order results on the same mesh (hence the $\mathbb{P}^2$ results have more degrees of freedom). There are no spurious oscillation across the shock.

The method also works on more complex problems such as the Guckenheimer Riemann problem. This is a well-known non-convex conservation law. We provide this example for two reasons : the solution structure is more complex, in particular a fan is ended by a shock. The second reason is that, even though we do not have any analytical proof of the entropy stability of the scheme, the numerical results seem to indicate that the entropy condition is properly met.

The problem is originally time dependent, and described by

$$\frac{\partial u}{\partial t} + \frac{1}{2}\frac{\partial u^2}{\partial x} + \frac{1}{3}\frac{\partial u^3}{\partial y} = 0$$

$$u(x,y,0) = \begin{cases} 0 & \text{if } 0 < \arctan\left(\frac{y}{x}\right) < \frac{3\pi}{4} \\[2mm] 1 & \text{if } \frac{3\pi}{4} < \arctan\left(\frac{y}{x}\right) < \frac{3\pi}{2} \\[2mm] -1 & \text{if } \frac{3\pi}{2} < \arctan\left(\frac{y}{x}\right) < 2\pi \end{cases} \tag{35}$$

The solution is self similar, and it can be recast as

$$u(x,y,t) = v(\frac{x}{t}, \frac{y}{t}) = v(\xi, \nu)\,,$$

where the function v satisfies

$$-\xi v_\xi - \nu v_\nu + \frac{1}{2}\frac{\partial v^2}{\partial \xi} + \frac{1}{3}\frac{\partial v^3}{\partial \nu} = 0 \tag{36a}$$

with the boundary conditions

$$\lim_{r \to +\infty} v(r\cos\theta, r\sin\theta) = u(\cos\theta, \sin\theta, 0). \tag{36b}$$

Solving (36) amounts to solve (35) at $t = 1$. This problem has been discussed in [29] and has been drawn to our attention by M. Ben Artzi (Hebrew University of Jerusalem). The flux $g(u) = \frac{u^3}{3}$ is non convex and this induces sonic shocks. The exact solution consists in

- A shock coming out from the line $y = 0$ that moves at the speed $1/3$ in the positive direction,

- a steady shock at $x = 0$,

- A shock coming out from the line $x + y = 0$. The analysis of [29] using the self-similarity of the solution indicates that the location of this shock is $x + y - 5/6t$, with in our case, $t = 1$.

To simulate this problem, we rewrite (36a) as

$$\frac{\partial F(u)}{\partial \xi} + \frac{\partial G(u)}{\partial \nu} + 2u = 0 \tag{37}$$

with $F(u) = \frac{1}{2}u^2 - \xi u$ and $G(u) = \frac{1}{3}u^3 - \nu u$. The total residual on T takes now into account the presence of the source term :

$$\Phi^T = \int_{\partial T} \Big(F(u)n_x + G(u)n_y\Big)dxdy + 2\int_T udxdy$$

The integral on ∂T has been evaluated by means of a 3 point Gaussian quadrature formula, while the volume integral can be easily computed exactly.

The solution is displayed on figure 8. We see that even for this non convex problem, there are no oscillations close to the discontinuities. More interestingly, the correct entropy solution is obtained. This is a topic for further investigation : it is known that the upwind RD schemes may yield solutions that do not respect the entropy inequality [30]. This does not seem to be the case with our "centered" approach, most likely due to the presence of the term (32).

5 Extension to systems

In this section, we describe the scheme for the system of the steady Euler equations described by (1a) with the flux (2) and the conserved variables $\mathbf{u} = (\rho, \rho\vec{u}, E)^T$. We assume a perfect gas equation of state, and $\gamma = 1.4$ in the applications. We denote by A (resp. B) the Jacobian matrix of the flux f_1 (resp. f_2) with respect to the state $\mathbf{u}$.

The scheme is a direct extension of what is done in the scalar case, with a major modification because the natural unknown is a vector, not a scalar. We provide the details of the scheme description on a single element K since there is no ambiguity.

All the results obtained in the system case are given for quadratic interpolants. It is of course possible to use cubic (or larger degree) interpolant, but since we deal with *steady* problems, we do not expect major improvement in the solution. This statement is certainly not correct for *unsteady* problems which is the topic of a coming paper.

5.1 The first order building block

The first order scheme is constructed on the Lax–Friedrichs scheme, i.e., for any degree of freedom $\sigma \in K$,

$$\Phi_\sigma = \frac{1}{n_d^K} \oint_{\partial K} \mathbf{f}(\mathbf{u}^h) \cdot \vec{n}dl + \alpha_K(\mathbf{u}^h - \bar{\mathbf{u}}). \tag{38}$$

Here n_d^K is the number of degrees of freedom in K, hence $n_d^K = 6$ for a triangle and $n_d^K = 9$ for a quadrangle: we have run the third order version of the scheme

in the examples. The total residual $\oint_{\partial K} \mathbf{f}(\mathbf{u}^h) \cdot \vec{n} dl$ is evaluated by Simpson formula: if $\Gamma = [a, b]$ is an edge of K and $c = \frac{a+b}{2}$, we set

$$\int_\Gamma f(x) dl \equiv \frac{1}{6} \left(f(a) + 4f(c) + f(b) \right),$$

which amounts, in our case, to use a quadratic interpolant of $\mathbf{f}$ in K. The average state is

$$\bar{\mathbf{u}} = \frac{\sum_{\sigma \in T} \mathbf{u}_\sigma}{n_d^K},$$

and α_K is larger than the spectral radius of the flux Jacobians at the degrees of freedom. In practice, it is set to twice this maximum.

5.2 Controlling the oscillations

In the scalar part, the control of oscillations is achieved by "limiting" the ratios Φ_σ / Φ. In the system case, this quantity has no meaning. Hence, we adapt the procedure presented in [26]. Using the average state $\bar{\mathbf{u}}$, we compute the average flow direction, i.e.

$$\vec{n} = \frac{\vec{\bar{u}}}{||\vec{\bar{u}}||} = (n_1, n_2).$$

Then we evaluate the Jacobian matrix

$$K_{\vec{n}} = A(\bar{\mathbf{u}})n_1 + B(\bar{\mathbf{u}})n_2 \tag{39}$$

which is diagonalizable in $\mathbb{R}$. The eigenvectors are r_p for $p = 1, \cdots, 4$ associated to the eigenvalues

$$\lambda_{1,2} = \vec{\bar{u}} \cdot \vec{n} = ||\vec{\bar{u}}||, \lambda_3 = ||\vec{\bar{u}}|| - \bar{c}, \lambda_4 = ||\vec{\bar{u}}|| + \bar{c}.$$

Last, we denote by ℓ_p the right eigenvectors of the system, i.e. the linear forms such that any state vector $X \in \mathbb{R}^4$ can be decomposed as

$$X = \sum_{p=1}^4 \ell_p(X) r_p.$$

Our method is then the following:

1. We decompose the first order residuals Φ_σ into "characteristic" residuals,

$$\text{for each } \sigma, \Phi_\sigma = \sum_{p=1}^4 \ell_p(\Phi_\sigma) r_p.$$

We denote the characteristic residual by $\varphi_\sigma^p = \ell_p(\Phi_\sigma)$, they satisfy the conservation relation:

$$\text{for each } p \in \{1, \cdots, 4\}, \sum_{\sigma \in K} \varphi_\sigma^p = \ell_p(\Phi) := \varphi^p$$

2. For any $p = 1, \cdots, 4$, we "limit" the characteristic sub-residual by the same procedure as in the scalar case:

$$\beta_\sigma^p := \frac{\left(\dfrac{\varphi_\sigma^p}{\varphi^p} \right)^+}{\displaystyle\sum_{\sigma' \in T} \left(\dfrac{\varphi_{\sigma'}^p}{\varphi^p} \right)^+}$$

3. We construct the limited residual by

$$\Phi_\sigma^\star := \sum_{p=1}^4 \beta_\sigma^p \varphi^p r_p. \tag{40}$$

The property (11) is satisfied in a suitable norm. Indeed, if A_0 denotes the Hessian of the mathematical entropy evaluated at $\bar{\mathbf{u}}$, we know that we can find a set of eigenvectors r_l that are orthogonal for the metric defined by A_0. We recall this in the annex A. Indeed, if we denote by $(\, . \, , \, . \,)_{A_0}$ the scalar product associated to A_0. We have

$$\varphi_\sigma^p = (r_p, \Phi_\sigma)_{A_0}$$

so that

$$\begin{aligned}
||\Phi_\sigma^\star||_{A_0}^2 &= \sum_p |\beta_\sigma^p|^2|; |\varphi^p|^2 \\
&\leq \sum_p |\varphi^p|^2 \\
&\leq ||\Phi||_{A_0}^2
\end{aligned} \tag{41}$$

where of course we have assumed that the eigenbasis $\{r_p\}$ is orthonormal.

The matrix A_0 is not uniform, but we can nevertheless state that if the conserved state is such that the density and the pressure are bounded from above and below, all the norms defined by the $A_0(\bar{\mathbf{u}})$ are equivalent and the LP property is uniformly satisfied.

The last step is to get an explicit form of such a basis. As recalled in the annex A, the standard eigenvectors of the Euler equations are simple and good candidates for that since it can easily be shown that they are orthogonal for the quadratic form defined by the entropy. Hence, this is our practical choice. They are evaluated as the eigenvectors of (39) where $\vec{n}$ is the normalized averaged velocity. It the case of a stagnation point, we choose the x direction.

5.3 Spurious mode filtering procedure.

As for the scalar case, the scheme (7b)–(40) produces very good results in discontinuous regions, and very poor one in the smooth parts of the flow. Indeed

the iterative convergence is very poor, and the results are at most first order accurate.

To the same problem, we use the same cure. We add to $(\Phi_\sigma)^\star$ a correction of the type

$$h_K \int_K \left((A, B) \cdot \nabla \varphi_\sigma \right) \tau \left((A, B) \cdot \nabla \mathbf{u} \right) dx \tag{42}$$

where the matrix τ is a scaling matrix.

Several choices have been tested. The simplest one is a diagonal scaling,

$$\tau = \alpha \mathrm{Id}$$

where α has the dimension of the inverse of a speed. We can take α as the inverse of the largest possible eigenvalue of the system, i.e. $\alpha = (||\vec{u}|| + c)^{-1}$. A better choice seems to be

$$\tau = h_K^{-1} N$$

where the N matrix is

$$N = \left(\sum_{\vec{n} \text{ normal to } \partial K} K_{\vec{n}}^+ \right)^{-1} = 2 \left(\sum_{\vec{n} \text{ normal to } \partial K} |K_{\vec{n}}| \right)^{-1}. \tag{43}$$

In this relation, the summation is the edges of K (i.e. 3 for triangles, and 4 for quadrangles) and if $\vec{n}$ is any scaled inward vector normal to the boundary ∂K of K, we have set

$$K_{\vec{n}} = (A, B)\vec{n} = An_x + Bn_y.$$

The Jacobian matrix are evaluated at the averaged state

$$\bar{\rho} = \frac{1}{n_d^K} \sum_{\sigma \in K} \rho_\sigma, \qquad \overline{\vec{u}} = \frac{1}{n_d^K} \sum_{\sigma \in K} \vec{u}_\sigma, \qquad \bar{p} = \frac{1}{n_d^K} \sum_{\sigma \in K} p_\sigma$$

It is shown in [12] that if the velocity $\overline{\vec{u}} \neq 0$, the matrix $\sum_{\vec{n} \text{ normal to } \partial K} K_{\vec{n}}^+$ is always invertible. To avoid this situation, we slightly modify the eigenvalues λ^+ appearing in the evaluation of L_i by

$$\lambda^+ \to \widetilde{\lambda^+} = \begin{cases} \lambda^+ & \text{if } |\lambda| > \varepsilon \\[2mm] \dfrac{(\lambda + \varepsilon)^2}{4\varepsilon} & \text{else.} \end{cases}$$

This is reminiscent of Harten's entropy fix, but the role here is different. Here ε is a small number of the form $\alpha(||\vec{u}|| + c)$ with $\alpha \equiv 0.01$ in the numerical applications, but the simulations do not seem to be very sensitive to this parameter [4]. Note that we apply this to each of the eigenvalues, even though only the $\lambda = \overline{\vec{u}} \cdot \vec{n}_i$ needs to be modified.

---

[4]Indeed, one could avoid this by noting that, using again [12] where a decomposition using the entropy wave is described.

The second thing we do is to simplify the expression (42) in the spirit of (31). Namely, the residual is

$$\Phi_\sigma^{\star\star} = \Phi_\sigma^\star + \theta_K \Psi_\sigma \tag{44a}$$

with

$$\Psi_\sigma = |K| \sum_{\hat{x}_{\text{quad}}} \omega_{\text{quad}} \Bigg[\left((A, B)(\hat{x}_{\text{quad}}) J^{-1}(\hat{x}_{\text{quad}}) \nabla_{\hat{K}} \hat{\psi}_\sigma \right)(\hat{x}_{\text{quad}})$$
$$N\left((A, B)(\hat{x}_{\text{quad}}) J^{-1}(\hat{x}_{\text{quad}}) \nabla_{\hat{K}} u^h \right)(\hat{x}_{\text{quad}}) \Bigg] |\det(J^{-1}(\hat{x}_{\text{quad}}))|. \tag{44b}$$

where J is the Jacobian of the transformation between the reference element $\hat{K}$ and the element K.

As in the scalar case, we have taken the smallest set of quadrature points, i.e. the vertices of K. This also simplifies the evaluation of the Jacobian matrices. The last feature is the parameter θ_K. It has to be of the order of unity in the smooth regions, and of the order of zero when the gradient of the solution is large. In the numerical experiments, we have chosen a sensor on the density,

$$\theta_K = 1 - \max_{\sigma \in K} \left(\max_{K', \sigma \in K'} \max_{\sigma' \in K'} \frac{|\rho_{\sigma'} - \overline{\rho}_{TK}|}{|\rho_{\sigma'}| + |\overline{\rho}_{K'}|} \right). \tag{44c}$$

or on the entropy

$$\theta_K = 1 - \max_{\sigma \in K} \left(\max_{K', \sigma \in K'} \max_{\sigma' \in K'} \frac{|s_{\sigma'} - \overline{s}_{K'}|}{s_{\sigma'} + \overline{s}_{K'}} \right). \tag{44d}$$

where $s = p\rho^{-\gamma}$ (in order to have a positive quantity). Both choices give very similar results, the second one being cleaner. Conservation is guarantied automatically.

5.4 Boundary conditions

We have used a simplified version of the boundary conditions. If an element K has an edge, Γ_K, on the boundary, we need to add to the degrees of freedom on Γ_K a boundary residual. We denote it by $\Phi_\sigma^{\Gamma_K}$. These residuals should satisfy the conservation relation

$$\sum_{\sigma \in \Gamma_K} \Phi_\sigma^{\Gamma_K} = \int_{\Gamma_\sigma} \left(\mathcal{F}_n(u^h) - f(u^h) \cdot \vec{n} \right) dl$$

where $\mathcal{F}_n$ is a boundary flux. In the examples of this paper, two types of boundary are considered:

- *Wall boundary conditions* . The condition $\vec{u} \cdot \vec{n} = 0$ is weakly imposed so that

$$\mathcal{F}_n(u^h) = \begin{pmatrix} 0 \\ p(u^h)n_x \\ p(u^h)n_y \\ 0 \end{pmatrix}$$

- *Inflow/outflow boundary conditions*. The state at infinity is U_∞ and we take here the modified Steger-Warming flux

$$\mathcal{F}_n(u^h) = \big(A(u^h) \cdot \vec{n}\big)^+ u^h + \big(A(u^h) \cdot \vec{n}\big)^+ u_\infty.$$

By analogy with what is done in [12], we have chosen a 'centered' version of the boundary residuals, namely

$$\Phi_\sigma^{\Gamma_K} = \int_{\Gamma_K} \big(\mathcal{F}_n(u^h) - f(u^h) \cdot \vec{n}\big)\psi_\sigma(x)dl$$

where again ψ_σ is the Lagrange basis function defined in K for σ. This is approximated by a quadrature formula with positive weights. The quadrature formula should be of order $k + d - 1$, i.e. 3 for a third order scheme in 2D. The actual residual is

$$\Phi_\sigma^{\Gamma_K} = |\Gamma_K| \sum_{\text{quadrature points}} \omega_{quad}\big(\mathcal{F}_n(u^h) - f(u^h)\big)(x_{\text{quad}}) \cdot \vec{n}. \qquad (45)$$

In the case of interest ($\mathbb{P}^2/\mathbb{Q}^2$ interpolation), we approximate these relation with Simpson's formula : only one term appears in the sum and it corresponds to σ.

All the meshes we have used are made of triangles or quadrangles. In this paper, we have used two type of boundary representation. In the first one, see figure 9-(a), is to adopt a piecewise linear representation of the boundary. As it can be seen, we might be quite far from the true geometry. In the second representation, see Figure 9-(b), we use a quadratic representation of the geometry. In principle, the situation should be better, but one has to be aware of two difficulties. First, the "numerical" representation of the boundary is not C^1 in general, even if the boundary is C^∞. An example is provided on figure 10 where we approximate the boundary of a NACA012 airfoil near the symmetry axis. The second problem is that even very simple geometries, such as circle, will not be represented exactly.

The second drawback could be solved by using NURBS representation of the boundary, the first one is here solved as follows: instead of trying to interpolate *exactly* in each boundary segment the boundary curve as in figure 9-a, we use a Bézier representation which amount to interpolate at the boundary points and respect the tangents at these points as in figure 9-b. We get an approximate quadratic representation of the boundary.

In order to simplify the coding, we have used use an isoparametric representation of each element, even for the interior elements. The filtering operator is the adapted to this context : we need a exact evaluation of the gradient and divergence operators.

Since the local representation is governed by the location of the degrees of freedom in any single element, we can use the same algorithm to represent curved elements and triangle elements. In any case, internal elements are still triangles and quadrangles.

The modifications of the scheme we have presented so far are minimal :

- Evaluation of the total residual and boundary residuals. They amount to evaluating integrals like

$$\int_\Gamma f \cdot \vec{n} dl$$

  where Γ is a possibly curved line and $\vec{n}$ is the outward unit normal. By using the local transformation between Γ and $[0,1]$ (denote it by ϕ) we have

$$\int_\Gamma f \cdot \vec{n} dl \approx \sum_{x_q \text{quad points}} \omega_q f(\phi(x_q)) \cdot (\phi')^\perp(x_q).$$

  Here $(\phi')^\perp$ is normal to ϕ' and in the same direction as $\vec{n}$.

- Evaluation of the filtering terms using (44b).

5.5   Summary of the final scheme for the system case.

This section is a cut and past of section 3.7 for the scalar case. We can repeat algorithm 1 where the main steps are modified as follows, besides that the variables are now vectors.

- At the degrees of freedom, we store the primitive variables : the density, velocity and pressure. From that are defined the Lagrange interpolant of these variables and the conservative variables. In particular, this enable to compute the total residual and the LxF residual (38).

- The residual are "limited" according to (39) and (40).

- the filtering operator (42) is added with $\tau = N$, see (43) so that we have (44).

- The boundary conditions (45).

The conserved variables are updated thanks to a linearised implicit scheme similar to what is done in [5]. From the conserved variables are decoded the primitive variables. Curved element are handled as explained in §5.4.

6   Numerical results for systems

In all the simulations (unless specified), we have used (44d). The meshes use
triangles only unless specified.

6.1   A convection problem

Our first example is the solution of the Euler system on $[0, 1] \times [0, 1]$ with the
following inflow condition at $y = 0$, $x = 1$ and $x = 0$:

$$\rho = 2 + \sin(\pi x), u = 10, v = 0, p = 1$$

The flow is assumed to be supersonic at $y = 1$. The exact solution is

$$\rho(x, y) = 2 + \sin(\pi x), u(x, y) = 10, v = 0, p(x, y) = 1, \tag{46}$$

the problem is a simple convection one. However, we use the full Euler system
and the scheme developed above to compute the solution. The scheme is really
third order accurate as it can be seen from the L^2 errors on figure 11.

6.2   Computation of jet

In this example, the domain is a square $\Omega = [0, 1]^2$. The boundary conditions
are :

- If $y > 0.5$ and $x = 0$, the Mach number is set to $M_\infty = 4$, the density is
  $\rho_\infty = 0.5$ and the velocity is $(u_\infty = M_\infty c_\infty, 0)$ with $c_\infty = \sqrt{\gamma p_\infty / \rho_\infty}$.

- If $y \leq 0.5$ and $x = 0$, the Mach number is set to 2.4, the velocity is $(u_\infty, 0)$
  and the density set to 1

- The other boundary are assumed to be supersonic.

In such a configuration, the flow is steady and supersonic. We have a shock
wave on the bottom, followed by a slip line and then a fan, see figure 12. Since
the flow is supersonic, the $x-$ coordinate plays the role of time : if one makes a
cross–section $x = $ const, we have a self-similar solution of the same type as what
one gets for a one dimensional shock tube. It is clear that there is no oscillation
at all on the density. The same conclusion holds for the other variables (not
displayed).

6.3   Subsonic examples

6.3.1   A subsonic example : flow over a sphere

We have run the case of a flow at $M_\infty = 0.35$ over a sphere. In that case,
the flow is symmetric with respect to the x–axis of the domain, but also with
respect to the y axis. This is a well known GAMM test case. We have run this
case with a second order scheme, a third order scheme, and again the second

order scheme on the mesh that has the same degrees of freedom as those of the $\mathbb{P}^2$ scheme. In other words, we subdivide each triangle into 4 smaller triangles which vertices are those of the large triangle and the mid–edges points. The initial mesh has 2719 nodes, 5308 elements and 100 nodes on cylinder. It is displayed on figure 13.

We see on Figure 14 which displays the pressure coefficient isolines the improvement of the solution quality when the scheme is upgraded from second order to third order. More important, the same Figure indicates clearly that the second order scheme on the refined mesh gives less accurate results than the third order one. Note that we have the same degrees of freedom in both cases.

This result is confirmed by Figure 15 which displays the entropy variation along the boundary. Except at the forefront stagnation point, the entropy deviation of the third order scheme is much closer than the exact one.

We have re-run this test case on an hybrid mesh using the second order and the third order schemes. In both cases, the same degrees of freedom are used (i.e. we use the DOFs of the sub-triangulation for the second order scheme). The results are shown on figure 16. The mesh use 81 points on the sphere. We get the same conclusions as before.

6.3.2 Subsonic flow over two aligned spheres

In order to better see the influence of the order of the scheme, we have run a variation of the previous case with a more complex geometric configuration. The Mach number at infinity is still 0.35, but instead of a single sphere, we have two now. More precisely, the geometrical configuration is as follow:

- Radius of the sphere: 1

- The center of the first sphere is $(0,0)$, the second sphere is $(10,0)$.

- The outer boundary is also a circle of center $(10,0)$ and radius 35.

The geometrical setup and a zoom of the mesh are shown on figure 17. The mesh is symmetric with respect to the x axis. The half mesh has

- 50 regularly spaced points on the outer boundary,

- 30 regularly spaced points on each half circle,

- The segment between the two circle has 60 regularly spaced points,

- the segment on the left has 50 points with a geometric ratio of 0.95,

- the segment on the right has 70 points with a ratio of 1.03.

Using the `emc2`[5] mesh generator, this creates a mesh with a total of 11007 vertices and 21796 triangles, that is 43811 $\mathbb{P}^2$ degrees of freedom.

---

[5] http://www-rocq1.inria.fr/gamma/cdrom/www/emc2/fra.htm

We have used the second order and third order schemes for comparisons. This mesh has been used for the third order scheme, i.e. additional degrees of freedom sitting on the mid-edge points has been used. In the case of the second order scheme, we have used the same degrees of freedoms, except on the boundaries where they have been projected onto the correct physical boundary. See figure 18. Saying this, we see that the mesh for second order runs is "better" than for third order runs.

The figure 19 show the pressure coefficient for the two schemes. There is no clear differences except that a closer inspection at the stagnation points seems to indicate that the third order solution is more symmetric with respect to the vertical axis than the second order one (we have the same isolines). More interesting are the entropy isolines, see figure 20

Again the same isolines have been used, and we see that the third order solution is much less dissipated than the second order one.

On figure 21, we have displayed on the top, the entropy deviation over the left circle and the right circle. The bottom plots represent (i) the entropy deviation distribution on the vertical axis that is in the middle of the two circles, and (ii) the same quantity at the exit of the computational domain. It is clear that, despite we have the same numbers of degrees of freedoms and the fact that the second order mesh is better that what we have used for the third order simulations, the third order solution is of a much better quality.

6.4   A transonic NACA0012 airfoils case

Our next examples is a flow over a NACA012 airfoil. It is transonic, and has the following conditions at infinity: $M = 0.8$, angle of attack of $1.25°$ The mesh has 10959 points and 21591. This corresponds to 43509 degrees of freedom.

On figure 22, we have displayed the Mach number, the pressure coefficients en relative entropy deviation for the third order version of the scheme. The same quantities plotted on the airfoil can be seen on figure 23. The solutions are fine. Note however a non physical overshoot in the entropy across the upper shock.

We have run many other tests as the following (results not shown). If we compare the second order' solution run with a mesh constructed from the mesh we have used where the element is sub-triangulated so that we have the same number of degrees of freedom, we can see an excellent agreement between the solutions with a main difference however. In both cases, the shock with is one element, but one element for the third order solution is roughly twice as large as an element for the second order one. Hence, the shock look more diffused in the third order case. However, the entropy levels are much lower, as we have already seen in the two sphere subsonic case.

6.5   The Ringleb test case

Another case is the Ringleb flow. It has been devised by F. Ringleb [31] in 1940, see [32] for a derivation of more general solutions. This is an isentropic,

irrotational two dimensional flow. It is defined from the streamline function (θ is the velocity angle with respect to a given direction and v is the norm of the velocity) $\psi = \frac{\sin\theta}{v}$. From this, it is possible to get the explicit form of the streamlines

$$x = \frac{1}{2}\frac{1}{\rho}\left(\frac{1}{v^2} - \frac{2}{k^2}\right) + \frac{J}{2}$$
$$y = \pm\frac{1}{k\rho v}\sqrt{1 - \left(\frac{q}{k}\right)^2}$$

with

$k = \frac{1}{\phi}$ a constant on any stream line, $J = \frac{1}{c} + \frac{1}{3c^2} + \frac{1}{5c^2} - \frac{1}{2}\log\left(\frac{1+c}{1-c}\right)$
$c = \sqrt{1 - \frac{\gamma-1}{2}q^2}, \rho = c^{2/(\gamma-1)}$

The pressure is determined by the equal entropy assumption. We see that the isotach lines are the circles

$$\left(x - \frac{J}{2}\right)^2 + y^2 = \frac{1}{4\rho^2 q^4}$$

From this it is possible to determine the exact solution: given a point (x, y), we determine the speed of sound c such that (x, y) belongs to the circle of center $(J(c)/2, 0)$ and radius $1/2(\rho q^2)$. Once this is done, we can get all the other values.

We have run this case in the (symmetric) domain defined by

- the circle $q = 0.3$ on the top and the bottom,

- the extreme stream lines $k = 0.4$ and $k = 0.8$.

The simulation has been conducted with two series of meshes. The first one is made of quads cut into two triangles, always in the same direction. The mesh is then made sym-etric. In the second one, we only consider the quads. In both cases, we have $2 \times P$ points on the streamlines $k = 0.3$ and 0.8 and P points on the circles $q = 0.3$. Here we have taken $P = 15, 30, 60$ and 100. The error (in the L^2 norm for the density are shown on figure 24. We see a slope of -3 for the third order scheme and -1.5 for the second order scheme. We also note that though the formal accuracy in both case is as expected, the effective accuracy on the quad meshes is much superior to what is obtained for triangle meshes.

6.6 A more complex case

We have run the same scheme on a scramjet–like configuration using an hybrid mesh as shown on Figure 25. This example has already been run in [5]. The inflow mach number is set to 3.5. The geometry is such that many waves coexist and interact in very complex flow patterns. This situation is particularly clear on the upper part of the internal body where shocks, fans and their reflection due

to wall interact. Again, in both cases, the same number of degrees of freedom have been used. Once again, the scheme has been run starting from a uniform flow configuration. Figure 26 shows the Mach number isolines. As expected, there is no real difference between the solutions since the flow is basically made of shock, fans, slip lines and constant states : this is not an accuracy case, but a case that shows that, despite the flow complexity, the third order scheme is robust.

However, one can see a small difference between the solutions : the slip line created by the interaction of two shocks after the blade is a little bit more twisted for the third order scheme than the second order one. We also see that the resolution of the discontinuities is in both case approximately one cell width.

7   Conclusion

We have described a systematic way of construction high order Residual distribution schemes of unstructured hybrid meshes. The construction use a continuous representation of data. From this information, we evaluate a total fluctuation and a high order mechanism for distribution these residual to the degrees of freedom is proposed and evaluated.

The accuracy of the scheme is evaluated on scalar problems and a standard accuracy test case. We have shown that for a given number of degree of freedom, our third order scheme is more accurate that a second order version of the scheme. The hybrid mesh solver is more accurate when quadrangles are used instead of only triangles.

The robustness of the method has been evaluated on subsonic, transonic, supersonic flow problems, as well as on complex configurations where many flow interactions occur.

The extension to the Navier Stokes equations and to unsteady problem will be considered in separate publications.

Acknowledgments

References

[1] A. Harten, B. Engquist, S. Osher, and S. R. Chakravarthy. Uniformly high order accurate essentially non-oscillatory schemes. III. (Reprint). *J. Comput. Phys.*, 131(1):3–47, 1997.

[2] X.L. Liu, S.J. Osher, and T. Chan. Weighted essentially non-oscillatory schemes. *J. Comput. Phys.*, 115(1):200–212, 1994.

[3] Bernardo (ed.) et al. Cockburn, editor. *Discontinuous Galerkin methods. Theory, computation and applications.*, volume 11 of *Lect. Notes Comput. Sci. Eng.* Springer, 1999.

[4] T.J.R.Hughes, G.Scovazzi, P. B.Bochev, and A.Buffa. A multiscale discontinuous galerkin method with the computational structure of a continuous galerkin method. *CMAME*, 195(19-22):2761–2787, 2006.

[5] R. Abgrall. Essentially non-oscillatory residual distribution schemes for hyperbolic problems. *J. Comput. Phys.*, 214(2):773–808, 2006.

[6] R. Abgrall and C.W. Shu. Development of residual distribution schemes for the discontinuous galerkin method: The scalar case with linear elements. *Communications in computational physics*, 5(2-4):376–390, 2009.

[7] R. Abgrall. A residual distribution method using discontinuous elements for the computation of possibly non smooth flows. *Advances in Applied Mathematics and Mechanics*, to appear.

[8] R. Struijs, H. Deconinck, and P.L. Roe. Fluctuation splitting schemes for the 2D Euler equations. VKI-LS 1991-01, 1991. Computational Fluid Dynamics.

[9] H. Deconinck, P.L. Roe, and R. Struijs. A multidimensional generalization of Roe's difference splitter for the Euler equations. *Computer and Fluids*, 22(2/3):215–222, 1993.

[10] E. van der Weide and H. Deconinck. Positive matrix distribution schemes for hyperbolic systems. In *Computational Fluid Dynamics*, pages 747–753, New York, 1996. Wiley.

[11] H. Deconinck, K. Sermeus, and R. Abgrall. Status of multidimensional upwind residual distribution schemes and applications in aeronautics. AIAA paper 2000-2328, June 2000. AIAA CFD Conference, Denver (USA).

[12] R. Abgrall. Toward the ultimate conservative scheme: Following the quest. *J. Comput. Phys.*, 167(2):277–315, 2001.

[13] P. De Palma, G. Pascazio, G. Rossiello, and M. Napolitano. A second-order-accurate monotone implicit fluctuation splitting scheme for unsteady problems. *J. Comput. Phys.*, 208(1):1–33, 2005.

[14] G. Rossiello, P. De Palma, G. Pascazio, and M. Napolitano. Third-order-accurate fluctuation splitting schemes for unsteady hyperbolic problems. *J. Comput. Phys.*, 222(1):332–352, 2007.

[15] C.W. Shu and S.J. Osher. Efficient implementation of eno schemes. *J. Comput. Phys.*, 1, 1989.

[16] C. Corre, G. Hanss, and A. Lerat. A residual-based compact scheme for the unsteady compressible Navier-Stokes equations. *Comput. Fluids*, 34(4-5):561–580, 2005.

[17] M.E.Hubbard. Non-oscillatory third order fluctuation splitting schemes for steady scalar conservation laws. *J. Comput. Phys.*, 222(2):740–768, 2007.

[18] R.Abgrall, A.Larat, M.Ricchiuto, and C. Tavé. A simple construction of very high order non oscillatory compact schemes on unstructured meshes. *Computers and Fluids*, 38(7):1781–1824, 2009.

[19] M.E.Hubbard and A.L.Laird. Achieving high-order fluctuation splitting schemes by extending the stencil. *Computer and Fluids*, 34(4-5):443–459, 2005.

[20] D. Caraeni and L. Fuchs. Compact third–order multidimensional upwind scheme for Navier Stokes simulations. *Theoretical and Computational Fluid Dynamics*, 15:373–401, 2002.

[21] R. Abgrall and L. Roe, P. High-order fluctuation schemes on triangular meshes. *J. Sci. Comput.*, 19(1-3):3–36, 2003.

[22] M.E. Hubbard. A framework for discontinuous fluctuation distribution. *Int. J. Numer. Methods Fluids*, 56(8):1305–1311, 2008.

[23] Mario Ricchiuto. Application of conservative residual distribution schemes to the solution of the shallow water equations on unstructured meshes. *Journal of Computational Physics*, 222(1):287–331, 2007.

[24] H.L. Atkins and C.W. Shu. Quadrature free implementation of discontinuous galerkin method for hyperbolic equations. *AIAA Journal*, 36(5):775–782, 1998.

[25] S.P. Spekreijse. Multigrid solution of monotone second order discretisations of hyperbolic conservation laws. *Mcomp*, 49(179):135–155, 1987.

[26] R. Abgrall and M. Mezine. Construction of second-order accurate monotone and stable residual distribution schemes for steady problems. *J. Comput. Phys.*, 195(2):474–507, 2004.

[27] H. Deconinck and M. Ricchiuto. Residual distribution schemes: foundation and analysis. In E. Stein, R. de Borst, and T.J.R. Hughes, editors, *Encyclopedia of Computational Mechanics*. John Wiley & Sons, Ltd., 2007. DOI: 10.1002/0470091355.ecm054.

[28] T.J.R. Hughes and A. Brook. Streamline upwind Petrov-Galervon karman institute for fluid dynamicsn formulations for convection dominated flows with particular emphasis on the incompressible Navier-Stokes equations. *Comp. Meth. Appl. Mech. Engrg.*, 32:199–259, 1982.

[29] J. Guckenheimer. Shocks and rarefactions in two space dimensions. *Arch. Ration. Mech. Anal.*, 59:281–291, 1975.

[30] K. Sermeus and H. Deconinck. An entropy fix for multidimensional upwind residual distribution schemes. *Computer and Fluids*, 34(4):617–640, 2005.

[31] F. Ringleb. Exakte lösungen der differentialgleichungen einer abadiatischen gassströmung. *ZAMM*, 20(4):185–198, 1940.

[32] R. von Mises. *Mathematical theory of compressible fluid flows.* Dover publications inc, 2004. Unabridged republication of the work first published by Academic Press Inc, New York, 1958.

[33] P.G. Ciarlet and P.A. Raviart. General Lagrange and Hermite interpolation in R^n with applications to finite element methods. *Arch. Ration. Mech. Anal.*, 46:177–199, 1972.

A Metric properties of the left and right eigenvectors

We recall here very classical material. Consider an hyperbolic system

$$\frac{\partial \mathbf{u}}{\partial t} + \sum_{i=1}^{d} A_i \frac{\partial \mathbf{u}}{\partial x_i} = 0 \tag{47}$$

equipped with an entropy S. We denote by $\mathbf{v}$ the entropy variables,

$$\mathbf{v} = \nabla_{\mathbf{u}} S$$

and by A_0 the Hessian of S with respect to $\mathbf{u}$. The Hessian is a positive definite symmetric matrix. This entropy symetrises (47), that is

$$A_0^{-1} \frac{\partial \mathbf{v}}{\partial t} + \sum_{i=1}^{d} A_i A_0^{-1} \frac{\partial \mathbf{v}}{\partial x_i} = 0 \tag{48}$$

is a system where the matrices $A_i A_0^{-1}$ are symetric.

Take a direction $\vec{n} = (n_1, \cdots, n_d)^T$ and set $K_n = \sum_i A_i n_i$ We see that the symetric matrix

$$A_0^{1/2} \big(K_n A_0^{-1} \big) A_0^{1/2} = A_0^{1/2} K_n A_0^{-1/2}$$

is similar to K_n: one can find a set of vectors $\{R_j\}$ and $\{r_j\}$

- the eigenvectors R_j of $A_0^{1/2} \big(K_n A_0^{-1} \big) A_0^{1/2}$ are orthonormal,

- the eigenvectors r_j of K_n satisfy

$$R_j = A_0^{1/2} r_j$$

Hence

$$R_j R_k^T = \left(A_0^{1/2} r_j\right)\left(A_0^{1/2} r_k\right)^T$$
$$= r_j A_0 r_k^T = \delta_j^k.$$

If all the eigenvalues of K_n are single, such a decomposition is unambiguous. In the case of an eigenvalue with multiplicity larger than 1, this results shows that one can find such a decomposition.

In the case of the Euler equations, the eigenvectors are well known :

$$r_{u_n}^1 = \begin{pmatrix} 1 \\ u \\ v \\ \dfrac{u^2 + v^2}{2} \end{pmatrix}, \quad r_{u_n}^2 = \begin{pmatrix} 0 \\ -n_y \\ n_x \\ -n_y u + n_x v \end{pmatrix}, \quad r_{+c} = \begin{pmatrix} 1 \\ u_n + c n_x \\ u_n + c n_y \\ H + u_n c \end{pmatrix}, \quad r_{-c} = \begin{pmatrix} 1 \\ u_n - c n_x \\ u_n - c n_y \\ H - u_n c \end{pmatrix}.$$

The above analysis shows that $r_{\pm c}$ are mutually orthogonal (with respect to the quadratic form defined by the Hessian of the entropy) and also orthogonal to $r_{u_n}^1$ and $r_{u_n}^2$. It is a simple but tedious exercices to check that

$$(r_{u_n}^2)^T A_0 r_{u_n}^1 = 0$$

so that the set of eigenvectors is an orthogonal basis for this quadratic form. For that reason, the transformations from conservative to "characteristic" variables and vice–versa are well defined. This is the reason why we have chosen to decompose the residual onto characteristic variables.

B Proof of the error estimates.

Proof of proposition 3.2. This inequality is a consequence of (8) because we have

$$-\int_\Omega \nabla \varphi_h(x) \cdot \mathbf{f}^h(\mathbf{u}^h) + \int_{\partial\Omega} \varphi_h(x) \mathbf{f}^h(\mathbf{u}^h) \cdot \vec{n} dl + \int_\Omega \varphi^h(x) S^h(\mathbf{u}^h) dx =$$
$$\left(-\int_\Omega \nabla \varphi_h(x) \cdot \mathbf{f}(\mathbf{u}) + \int_{\partial\Omega} \varphi_h(x) \mathbf{f}(\mathbf{u}) \cdot \vec{n} dl + \int_\Omega \varphi^h(x) S^h(\mathbf{u}) dx \right)$$
$$+ \left(-\int_\Omega \nabla \varphi_h(x) \cdot \left(\mathbf{f}(\mathbf{u}) - \mathbf{f}^h(\mathbf{u}^h) \right) \right.$$
$$\left. + \int_{\partial\Omega} \varphi_h(x) \left(\mathbf{f}(\mathbf{u}) - \mathbf{f}^h(\mathbf{u}^h) \right) \cdot \vec{n} dl + \int_\Omega \varphi^h(x) \left(S^h(\mathbf{u}) - S^h(\mathbf{u}^h) \right) dx \right)$$
$$\tag{49}$$

where $\mathbf{u}^h$ is the Lagrange interpolant of $\mathbf{u}$. From standard interpolation results [33], we have $|\varphi^h| \leq C$ and $|\nabla \varphi^h| \leq C'$, $|\mathbf{f}^h(\mathbf{u}^h) - \mathbf{f}(\mathbf{u})| \leq C(\mathbf{u}, \mathbf{f}) h^{k+1}$

and $|S^h(\mathbf{u}^h) - S(\mathbf{u})| \leq C(\mathbf{u}, S)h^{k+1}$. So that (49) is in norm smaller that $C(\mathbf{u}, \mathbf{f}, S)h^{k+1}$ for a suitable constant $C(\mathbf{u}, \mathbf{f}, S)$.

The second point is to notice that we have, for any K and Γ, $|\Phi_\sigma^{K,c}| \leq C(\mathbf{u}, \mathbf{f}, S)h^{k+d}$ and $|\Phi_\sigma^{\Gamma,c}| \leq C(\mathbf{u}, \mathbf{f}, S)h^{k+d-1}$ where d is the space dimension, see lemma B.1.

Then, for any K,

$$|\sum_{\sigma,\sigma' \in K} (\varphi(\sigma) - \varphi(\sigma')) (\Phi_\sigma^K - \Phi_\sigma^{K,c})| \leq \sum_{\sigma,\sigma' \in K} (|\varphi(\sigma) - \varphi(\sigma')|) (|\Phi_\sigma^K| + |\Phi_\sigma^{K,c}|)|$$

$$\leq \# \text{ of elements } \times N \times ||\nabla\varphi||_\infty h \times C(\varphi, \mathbf{f}, S)h^{k+d}$$

where N is the number of degree of freedom in each element. In a regular mesh for a bounded domain, the number ($\#$) of elements sizes like h^{-d} so that in the end, we can find a constant (again denoted by C) which depends on $\mathbf{u}$, $\mathbf{f}$, S and Ω such that

$$|\sum_{\sigma,\sigma' \in K} (\varphi(\sigma) - \varphi(\sigma')) (\Phi_\sigma^K - \Phi_\sigma^{K,c})| \leq C(\mathbf{u}, \mathbf{f}, S, \Omega)h^{k+1}.$$

The last estimation is to be done for the boundary terms. Using the consistency of the numerical flux, we first have

$$\left| \int_{\partial\Omega} \left(\mathcal{F}(\mathbf{u}^h, \mathbf{u}_-, \vec{n}) - \mathbf{f}^h(\mathbf{u}^h, \vec{n}) \right) \varphi_h(x)dl \right| \leq \int_{\partial\Omega} \left(\left| \mathcal{F}(\mathbf{u}^h, u_-, \vec{n}) - \mathcal{F}(\mathbf{u}^h, \mathbf{u}^h, \vec{n}) \right| \right) \varphi_h(x)dl$$

$$\leq L \int_{\partial\Omega} |\mathbf{u}^h - \mathbf{u}_-| \leq C(\mathbf{u}, \mathbf{f}, \partial\Omega)h^{k+1}$$

Similarly, we have, for any boundary edge, $|\Phi_\sigma^{\Gamma,c}| \leq C(\mathbf{u}, \mathbf{f})h^{k+d}$, (see lemma B.1. If the boundary of Ω is regular, the number of boundary faces is of the order of $h^{-(d-1)}$.

Thus, we get, using again the same arguments,

$$\left| \sum_{\Gamma \subset \partial\Omega} \sum_{\sigma,\sigma' \in \Gamma} (\varphi(\sigma) - \varphi(\sigma')) (\Phi_\sigma^\Gamma - \Phi_\sigma^{\Gamma,c}) \right| \leq C(\mathbf{u}, \mathbf{f}, \partial\Omega)h^{-d+1}h^{k+d} = C(\mathbf{u}, \mathbf{f}, \partial\Omega)h^{k+1}.$$

This completes the proof. $\qquad\square$

Lemma B.1. *Under the assumptions of proposition 3.2, we have*

- $|\Phi_\sigma^{K,c}| \leq C(\mathbf{u}, \mathbf{f})h^{k+d}$

- $|\Phi_\sigma^{\Gamma,c}| \leq C(\mathbf{u}, \mathbf{f}, S)h^{k+d-1}$

Proof. We only show the first result.

$$\Phi_\sigma^{K,c} = \int_K \psi_\sigma \left(\text{div } \mathbf{f}(\mathbf{w}^h) - S(\mathbf{w}^h) \right) dx$$

$$= \int_K \psi_\sigma \text{div} \left(\mathbf{f}(\mathbf{w}^h) - \mathbf{f}(\mathbf{w}) \right) dx$$

$$= -\int_K \nabla\psi_\sigma \cdot \left(\mathbf{f}(\mathbf{w}^h) - \mathbf{f}(\mathbf{w}) \right) + \int_{\partial K} \psi_\sigma \left(\mathbf{f}(\mathbf{w}^h) - \mathbf{f}(\mathbf{w}) \right) \cdot \vec{n}dl.$$

Since $\mathbf{f}(\mathbf{w}^h) - \mathbf{f}(\mathbf{w}) = O(h^{k+1})$, $\nabla\psi_\sigma = O(h^{-1})$, $|K| = O(h^d)$ and $|\partial K| = O(h^{d-1}$, we see that each integral scales like $O(h^{k+d})$ for a smooth mesh. $\qquad\square$
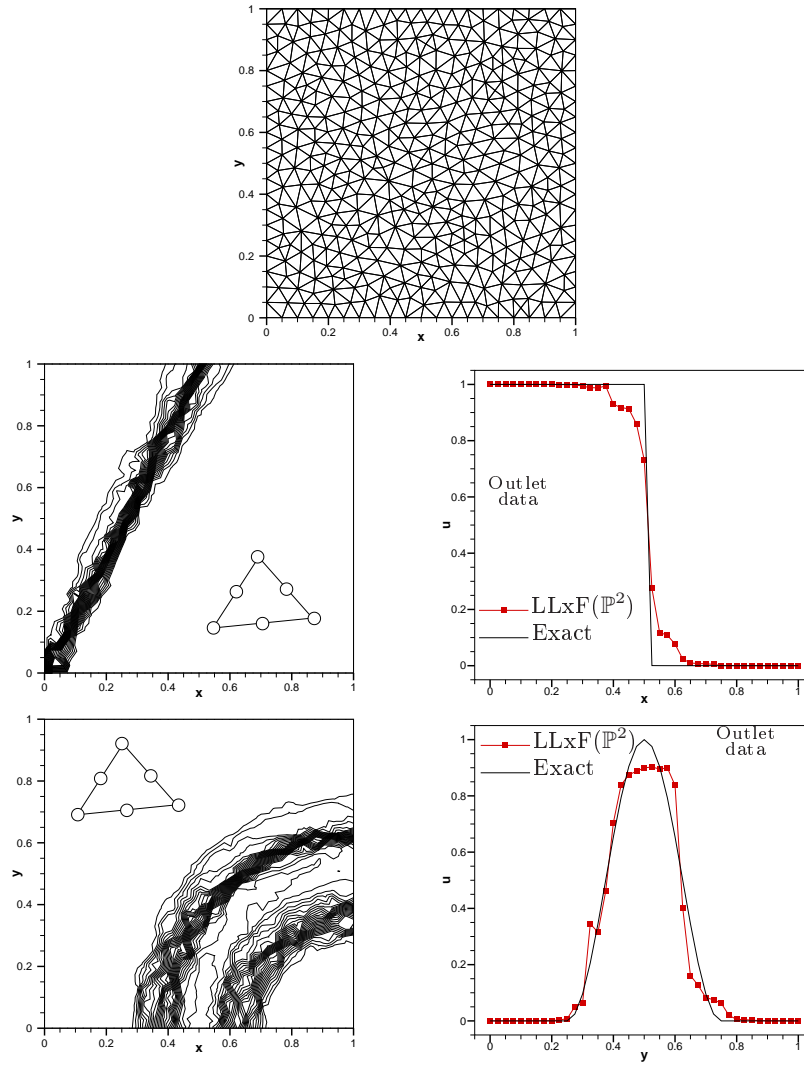
Figure 1: Convection problem : Results obtained with scheme (20)–(21) for $\mathbb{P}^2$ interpolation. Top : mesh. Middle : result for problem (26). Bottom : results for problem (27). The first order scheme is (24a).
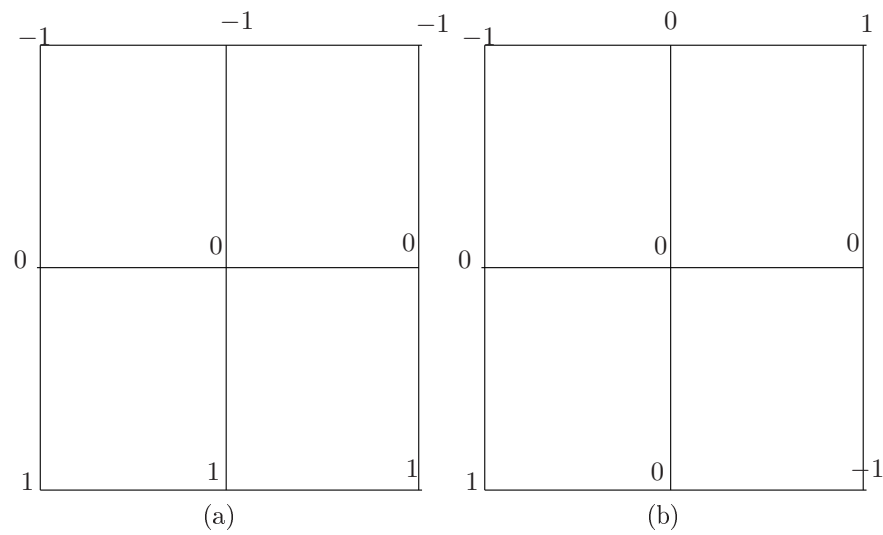
Figure 2: Two initialisations showing the creation of spurious modes. We show an elementary quad. The global initialisation is obtained by reproducing periodicaly the pattern.
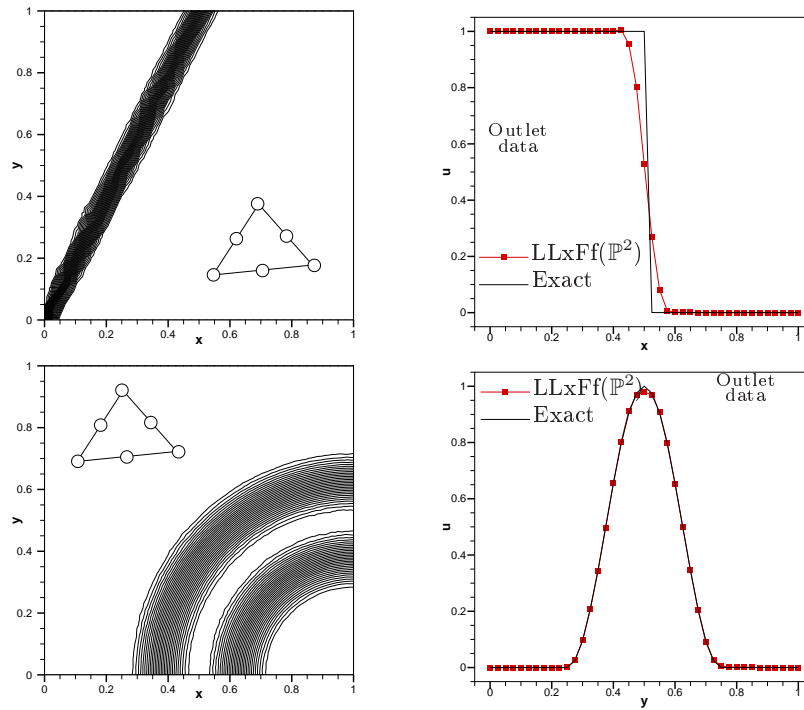
Figure 3: Rotation problem : Results obtained with the scheme (20)–(21)–(34) for $\mathbb{P}^2$ interpolation. Top : result for problem (26) (min $= -1.0094$, max $=$ 1.01). Bottom : results for problem (27) (min $= -0.1735 \, 10^{-4}$). The first order scheme is (24a).
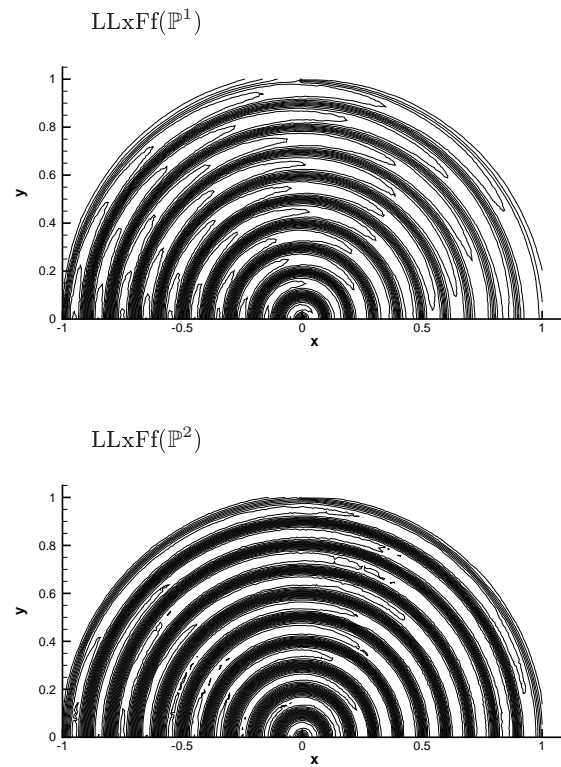
LLxFf($\mathbb{P}^1$)

LLxFf($\mathbb{P}^2$)

Figure 4: Rotation of the smooth profile: $u_{\mathrm{in}} = \sin(10x)$. Top: limited LLxF scheme, $\mathbb{P}^1$ approximation (LLxFf($\mathbb{P}^1$)). Bottom: limited LLxF scheme, P^2 approximation (LLxFf($\mathbb{P}^2$)). Computations run on the same number of degrees of freedom. Reference mesh size $h = 1/80$
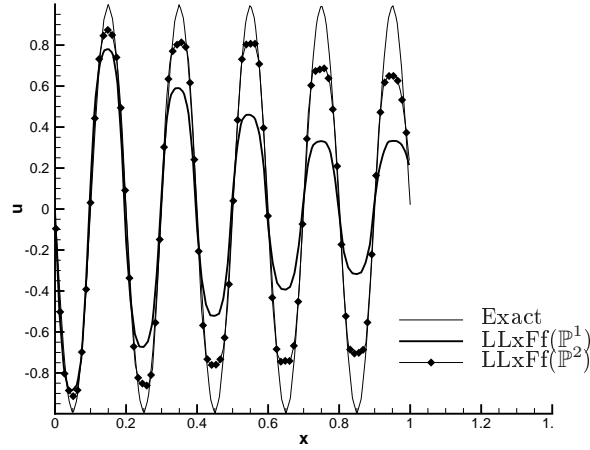
Figure 5: Rotation of the smooth profile: $u_{\text{in}} = \sin(10\pi x)$. Computed outlet profile. All computations run on the same number of degrees of freedom. Reference mesh size $h = 1/80$.
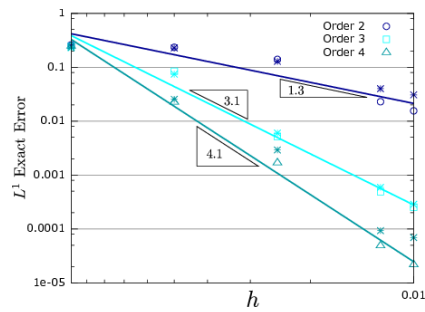
Figure 6: Mesh convergence for the constant advection problem (22) with $\vec{\lambda} = (0,1)^T$. The mean square slope are calculated with the errors measured on the hybrid meshes (represented by circles, squares and triangles). The star points correspond to the same simulations on triangular grids (same problem, same number of vertices).

LLxFf($\mathbb{P}^1$)   LLxFf($\mathbb{P}^2$)

cut at $y = 0.3$ at $y = 0.6$
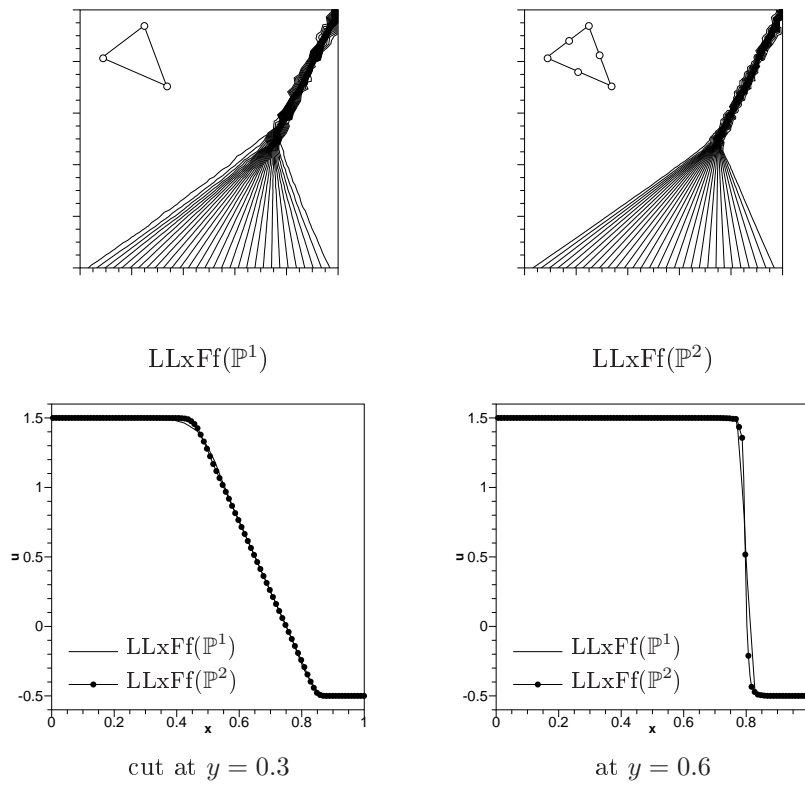
Figure 7: Burger equation, solution obtained with a $\mathbb{P}^1$ and $\mathbb{P}^2$ lagrange inter-polant and the LLxFf scheme
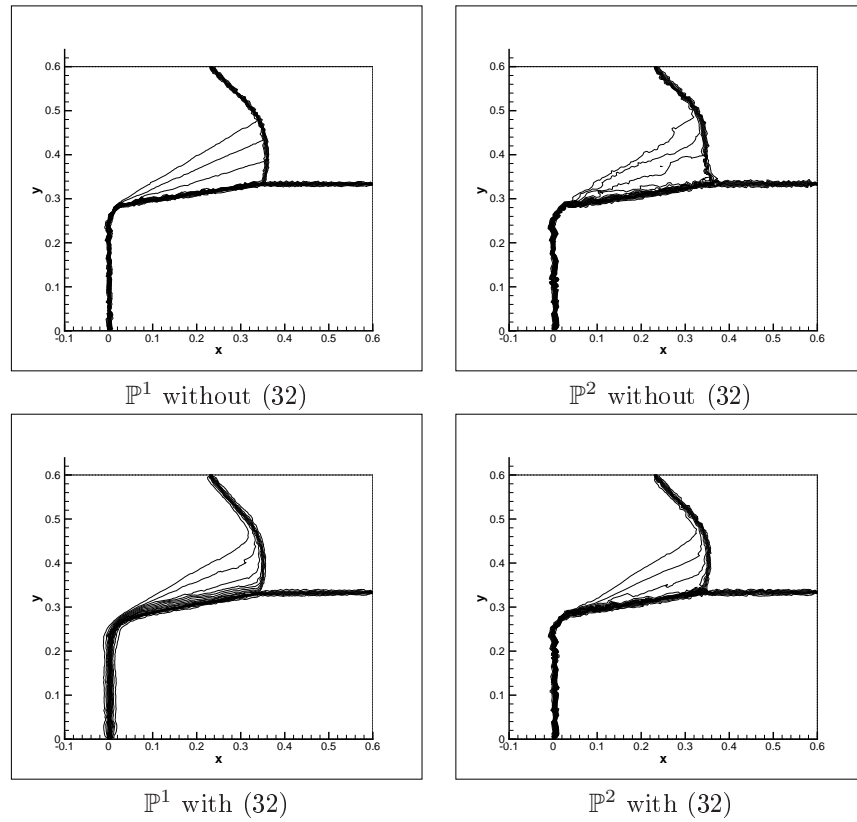
Figure 8: Burger equation, solution obtained with a $\mathbb{P}^1$ and $\mathbb{P}^2$ lagrange interpolant and the LLxFf scheme
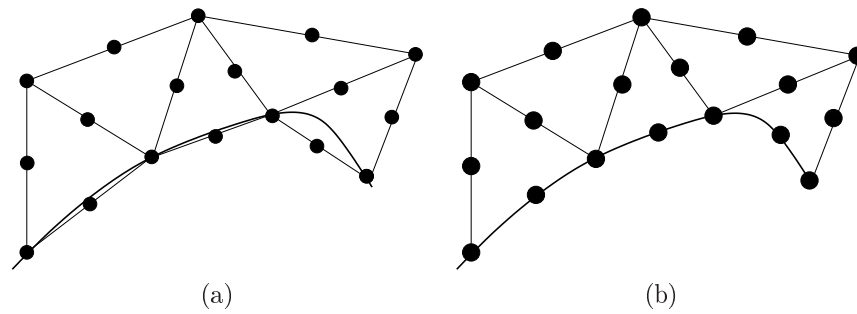
Figure 9: Boundary representation. The locations of the degrees of freedom are represented by the black circles. (a): piecewise linear representation, (b) P^2 isoparametric representation.
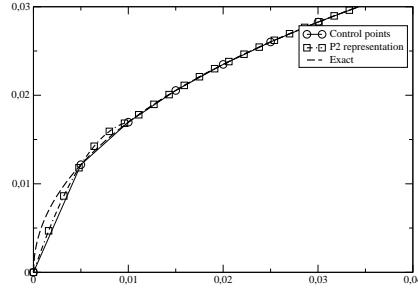
Figure 10: Comparison with the true geometry between the two boundary representation methods used in this paper. The degrees of freedom are represented by circles.
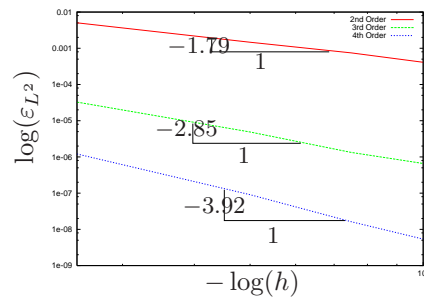
Figure 11: Transport equation : L^2 error for the second, third order and fourth order version of the LLxFf scheme.

second order   third order
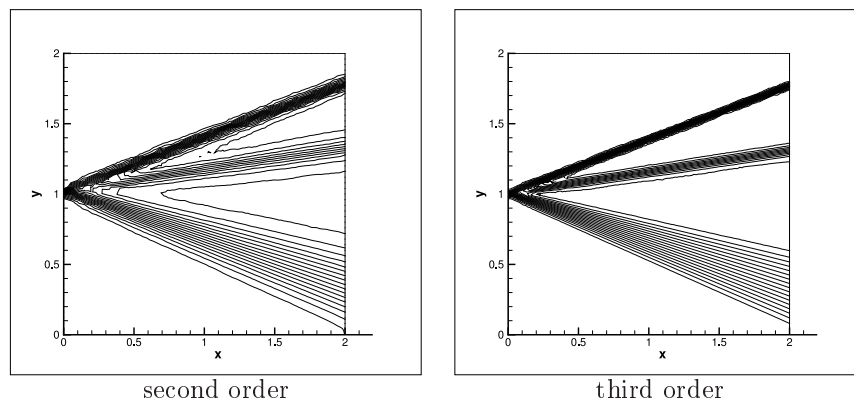
Figure 12: Jet problem : isolines of the density, second and third order LLxFf scheme. All the degrees of freedom are plotted. and the same isolines are plotted
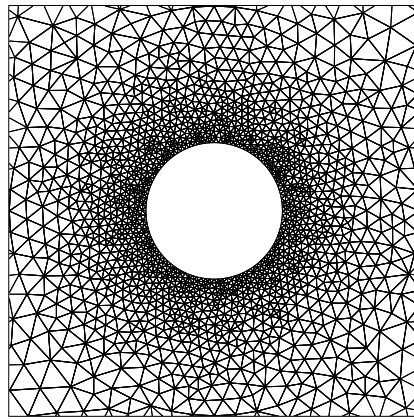
Figure 13: Subsonic sphere problem : Zoom of the mesh for the sphere problem. The mesh has no symetry.

Second order   Second order using the P^2 dofs

third order scheme
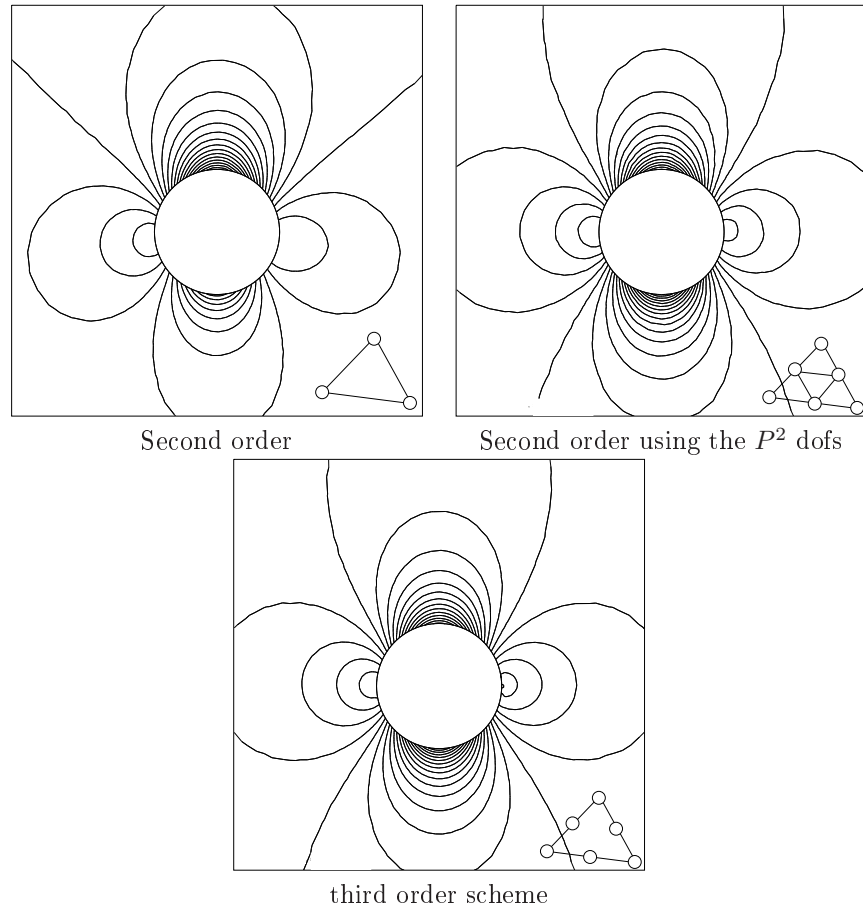
Figure 14: Subsonic sphere problem : Isolines of the pressure coefficient. We have the same isolines on each figure.
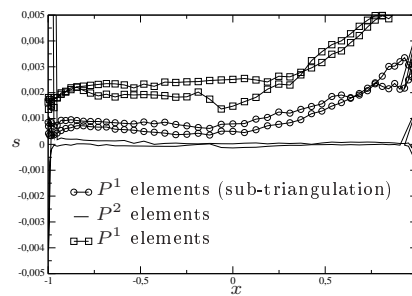
Figure 15: Subsonic sphere problem : Entropy variation along the boundary.

Pressure coefficient

Entropy variatinon

second order   third order
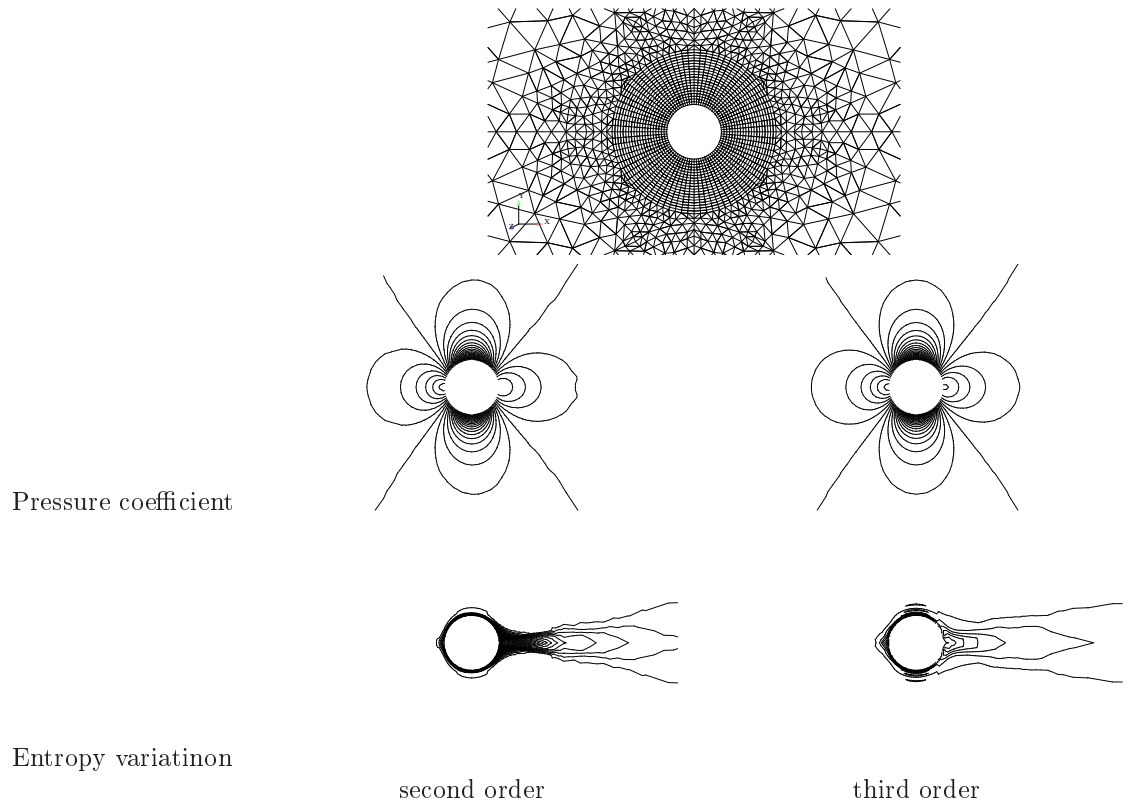
Figure 16: Subsonic sphere problem, hybrid mesh : Pressure coefficient and entropy variation on an hybrid mesh, $M_\infty = 0.35$.
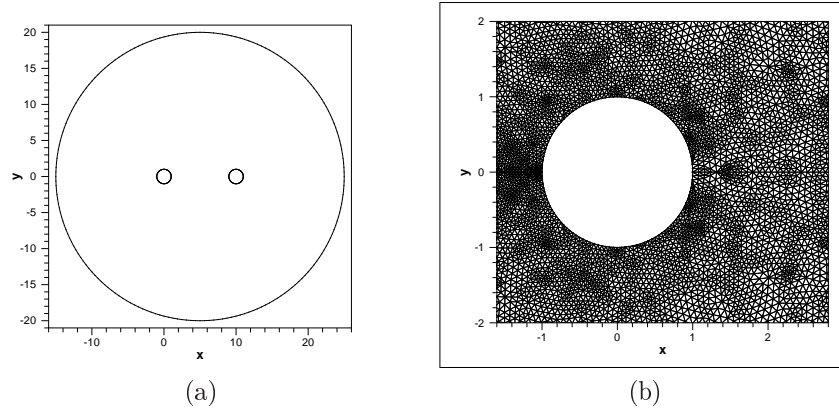
Figure 17: Subsonic two sphere problem. (a): Geometrical setup, (b): zoom of the mesh.
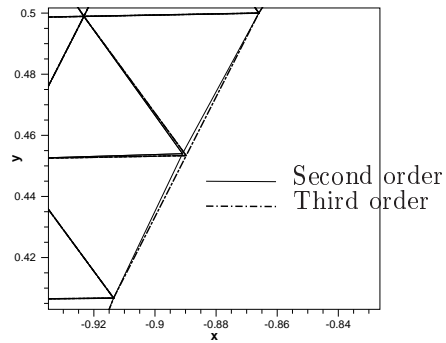
Figure 18: Subsonic two sphere problem. Location of the degrees of freedom on the boundary for the second and third order schemes.

Second order   Third order
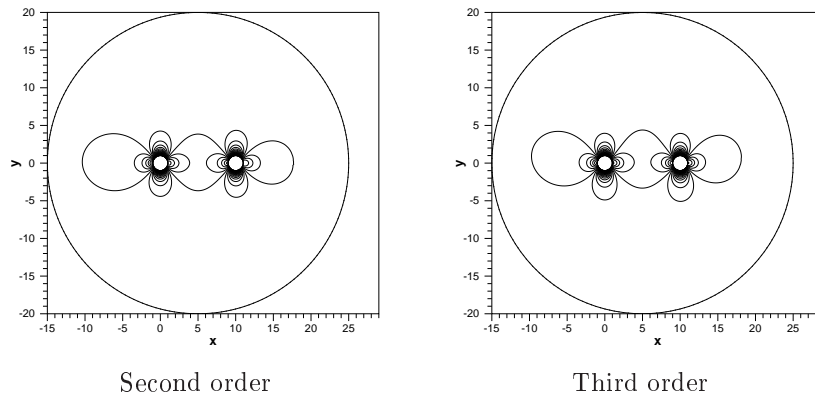
Figure 19: Subsonic two sphere problem. Pressure coefficient isolines. The same isolines have been used in the two cases.

Second order   Third order

Zoom   Zoom
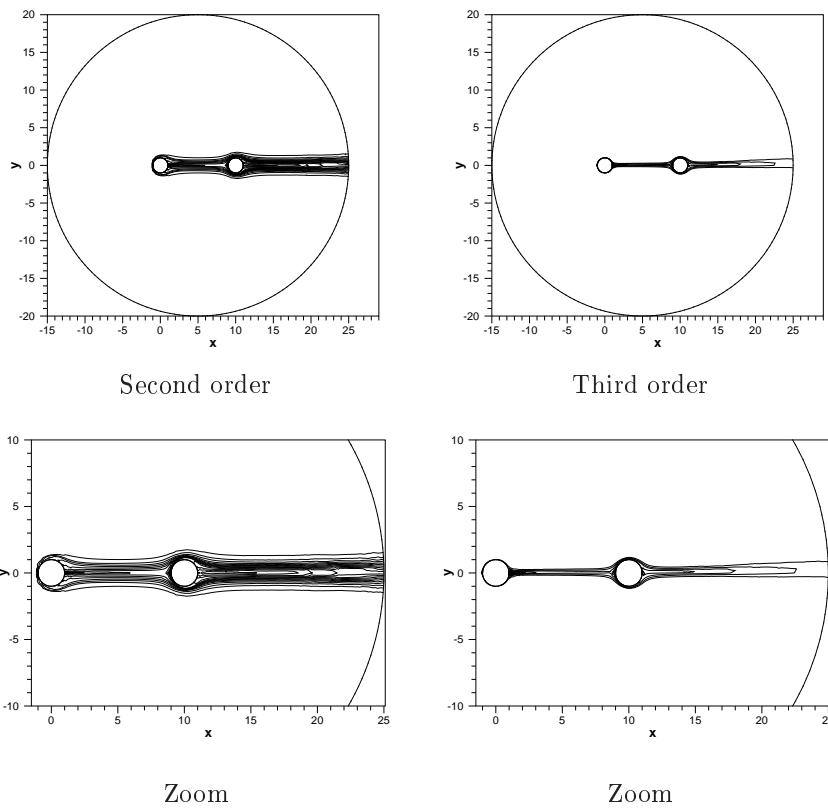
Figure 20: Subsonic two sphere problem, entropy isolines. The same isolines have been used in the two cases.

Abgrall, Larat & Ricchiuto

over the left circle   over the right circle

between the circle   at the outlet
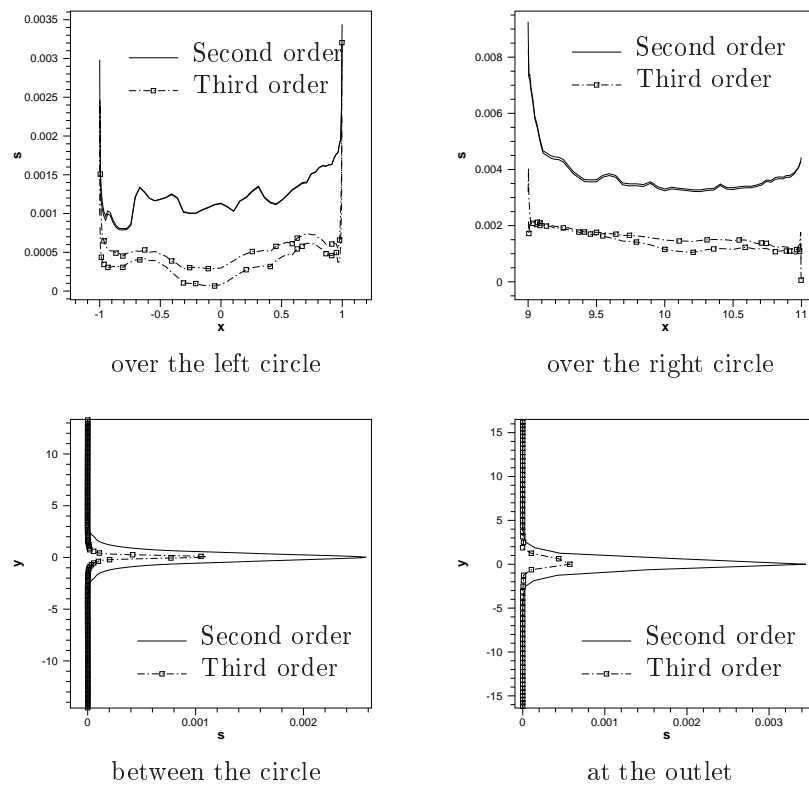
Figure 21: Subsonic two sphere problem, entropy on the boundaries of the discs, on the vertical axis in the middle between the two circles, at the outlet.

Mach number

Pressure
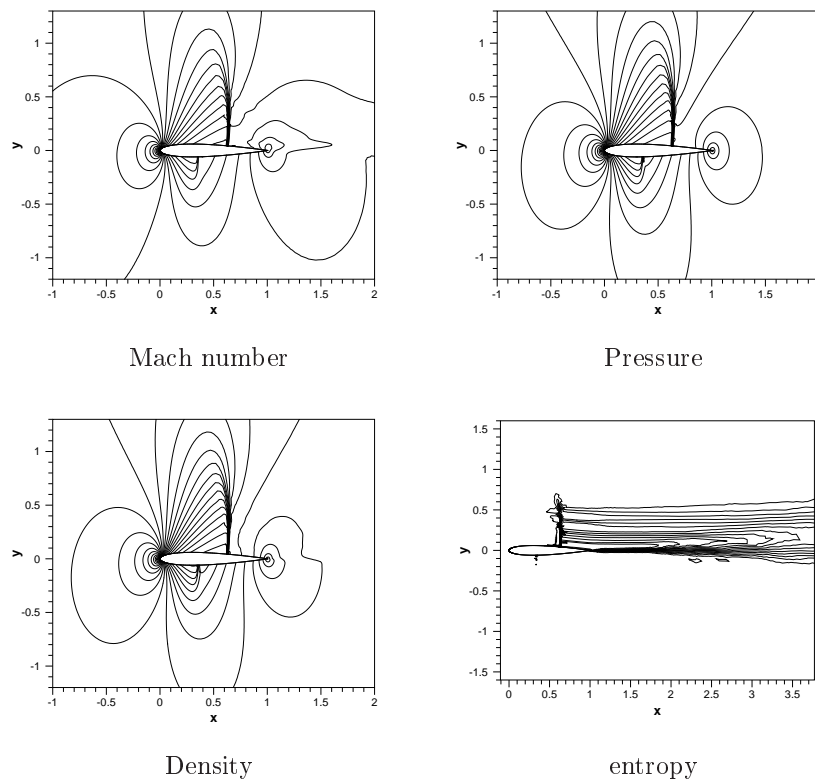
Density

entropy

Figure 22: Transonic NACA012 problem. Isolines of the Mach number, pressure, density and entropy for the NACA012 case.

Density

pressure coefficient

Density

pressure coefficient
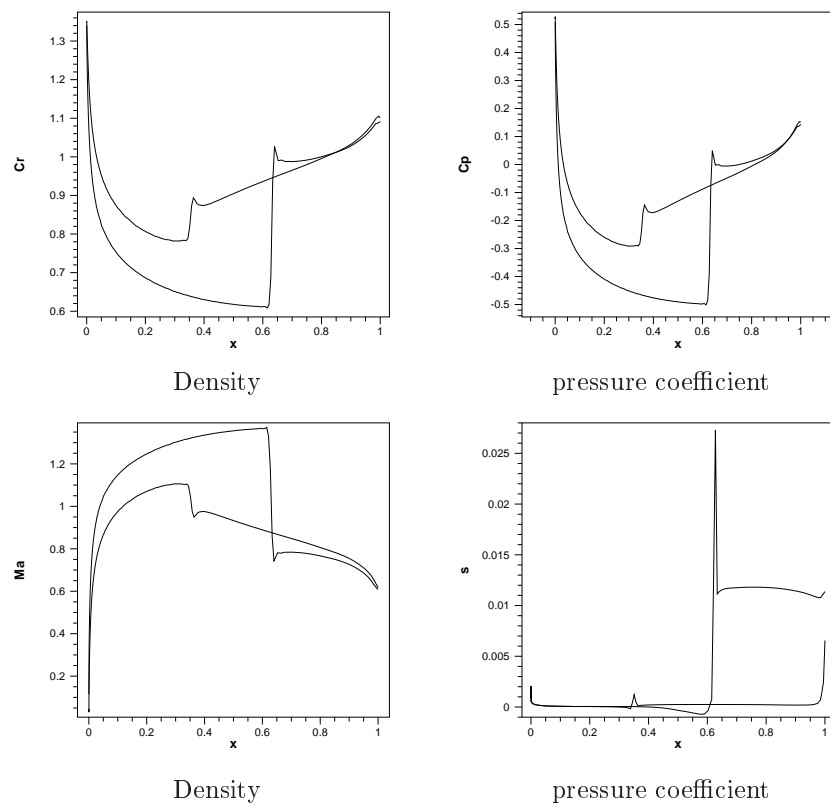
Figure 23: Transonic NACA012 problem. Plots on the profile, subsonic test case.
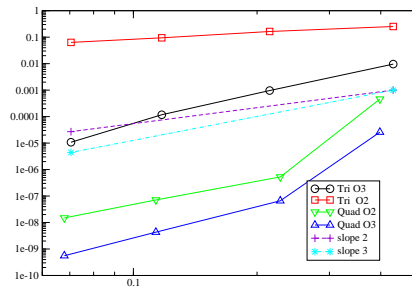
Figure 24: Ringleb flow problem. L^2 error on the density for the Ringleb flow. Tri stands for triangle, Quad for quadrangle. O2 stands for second order, O3 for third order.
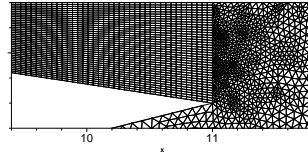
Figure 25: Zoom of the mesh for the scramjet problem.

zoom
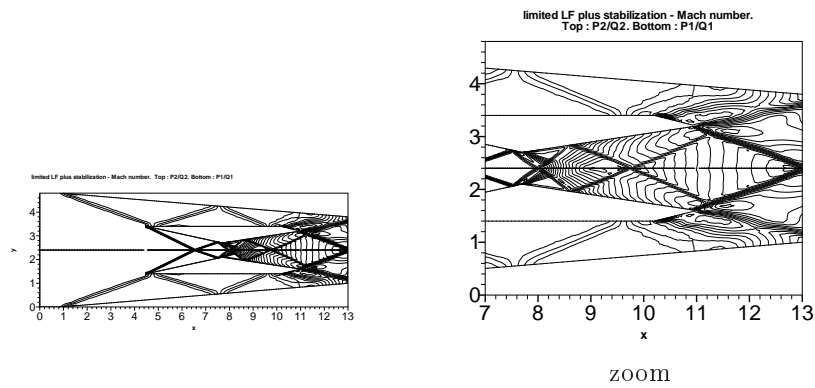
Figure 26: Scramjet problem. Mach number distribution. Top : the third order solution, bottom the second order solution. The same isolines are plotted.

	2D		3D	
Order	DG	RDS	DG	RD
2	$6n_s$	n_s	$24n_s$	n_s
3	$12n_s$	$4n_s$	$40n_s$	$8n_s$
4	$20n_s$	$9n_s$	$80n_s$	$27n_s$

Table 1: Number of degrees of freedom for third and fourth order approximation for triangular meshes.

h	Vertices	Triangles		Quadrangles
0.1	114	190	36	77
0.05	468	858	128	365
0.025	1784	3410	480	1465
0.0125	7777	15236	1982	6627
0.01	11454	22510	2858	9826

Table 2: Number of vertices, triangles and quadrangles for the different meshes used for the grid convergence. The left number in the column Triangles corresponds to the number of triangles in the triangular mesh, while the right one is the number of triangles in the hybrid grid. Hybrid grids have then about two times less elements than the triangular twin ones.

h	$\epsilon_{L^2}(P^1)$	$\epsilon_{L^2}(P^2)$	$\epsilon_{L^2}(P^3)$
1/25	0.50493E-02	0.32612E-04	0.12071E-05
1/50	0.14684E-02	0.48741E-05	0.90642E-07
1/75	0.74684E-03	0.13334E-05	0.16245E-07
1/100	0.41019E-03	0.66019E-06	0.53860E-08
	$\mathcal{O}^{\mathsf{ls}}_{L^2} = \mathbf{1.790}$	$\mathcal{O}^{\mathsf{ls}}_{L^2} = \mathbf{2.848}$	$\mathcal{O}^{\mathsf{ls}}_{L^2} = \mathbf{3.920}$

Table 3: L^2 errors for (22)–(26) with $u(x) = \varphi_0(x)$ on the inflow.

List of Figures

List of Tables