An almost fail-safe a-posteriori limited high-order CAT scheme

Emanuele Macca, Raphaël Loubère, Carlos Parés, Giovanni Russo

An almost fail-safe a-posteriori limited high-order CAT scheme

Emanuele Macca[c*]

Raphaël Loubère[a]

Carlos Parés[b]

Giovanni Russo[c]

[a] *Université de Bordeaux, CNRS, Bordeaux INP, IMB, UMR 5251, F-33400 Talence, France*
[b] *Departamento de Análisis Matemático, Estadística e Investigación Operativa, y Matemática aplicada, Universidad de Málaga, Bulevar Louis Pasteur, 31, 29010, Málaga, Spain*
[c] *Dipartimento di Matematica ed Informatica Universitá di Catania, Viale Andrea Doria 6, 95125, Catania, Italy*

Abstract

In this paper we blend the high order Compact Approximate Taylor (CAT) numerical schemes with an *a posteriori* Multi-dimensional Optimal Order Detection (MOOD) paradigm to solve hyperbolic systems of conservation laws in 2D. The resulting scheme presents high accuracy on smooth solutions, essentially non-oscillatory behavior on irregular ones, and almost fail-safe property concerning positivity issues. The numerical results on a set of sanity test cases and demanding ones are presented assessing the appropriate behavior of the CAT-MOOD scheme.

Key words: High-order scheme, CAT, MOOD, HLL/HLLC, Rusanov, Hyperbolic system of conservation laws, Hydrodynamics.

1. Introduction

Peter Lax and Burton Wendroff have presented their seminal finite difference numerical method more than 60 years ago in [19]. This scheme was designed to solve generic hyperbolic systems of conservation laws. At the core of the Lax-Wendroff (LW) scheme lays the so-called LW procedure which relies on Taylor expanding the solution in time up to second-order of accuracy, then replacing the time derivative by the space derivative according to the governing equations, and, finally, approximating the space derivatives by finite differences. This procedure revealed extremely fruitful.

Two main difficulties arise in the extension of this strategy to construct high order methods for nonlinear problems, and come from the transformation of time derivatives into spatial derivatives through the Cauchy-Kovaleskaya (CK) procedure: first, because of the nonlinearity one has to deal with discontinuous solution, so exact or approximate Riemann problems have to be taken into account in the procedure; second, this approach may indeed be impractical from the computational point of view because it often requires extended symbolic calculus, ended up into inefficient codes.

*Corresponding author

Email addresses: emanuele.macca@unict.it (Emanuele Macca[c*]), raphael.loubere@math.u-bordeaux.fr (Raphaël Loubère[a]), pares@uma.es (Carlos Parés[b]), giovanni.russo1@unict.it (Giovanni Russo[c])

In the context of ADER methods introduced by Toro and collaborators the first difficulty was solved by high order genralization of the Riemann problem at each call interface, (see [39, 37, 31]), while the second one has been circumvented by replacing the CK procedure by local space-time problems that are solved with a Galerkin method: see [12], [11]. The strategy introduced in [42] to avoid the CK procedure consisted in computing time derivatives in a recursive way using high-order centered differentiation formulas combined with Taylor expansions in time. This strategy leads to high-order Lax-Wendroff Approximated methods (LAT) that are oscillatory close to discontinuities: in [42] they were combined with WENO reconstructions to compute the first time derivatives. The resulting methods give non-oscillatory and accurate results. Compact Approximated Taylor methods (CAT) introduced in [5] circumvent the CK procedure using the same strategy as LAT methods. These methods are compact in the sense that the length of the stencils is minimal: $(2P + 1)$-point stencils are used to get order $2P$ compared to $4P + 1$ -point stencils in LAT methods with P an arbitrary natural number. Unlike LAT methods, CAT methods reduce to the standard high-order Lax-Wendroff methods when applied to linear problems and, due to this, they have better stability properties than LAT: see [5].

When dealing with discontinuous solutions which may occur for any hyperbolic system of Partial Differential Equations (PDEs), the key point of most numerical methods is their ability to dissipate appropriately. In other words extra dissipation must be added. The questions about where, when and much dissipation has to be added are of paramount importance to ensure that the numerical method can handle smooth flows and discontinuous solutions equally well. Second-order shock-capturing schemes have been obtained with slope or flux limiters relying on maximum principle preservation, or alternative related procedures. Beyond second-order accuracy, the limiting is not anymore a well-agreed subject of research. The most known technique is presumably the ENO/WENO procedure [34] for finite volume (FV) or finite differences (FD) schemes. Most of the limiting techniques rely on blending the first-order scheme/flux/reconstruction with a high-order one, using some *a priori* sensor to determine where this blending would be appropriate.

The limiting entirely depends on the quality of the *a priori* sensor which must determine where to act and the amount of blending, i.e. how much dissipation is needed to ensure that the numerical solution is physically and numerically acceptable. Based on this philosophy, CAT methods have been combined with WENO in [5] and [6] to avoid oscillations near discontinuities. Nevertheless this combination is not optimal: while the best CAT methods are those of even order, WENO methods have odd accuracy order. Moreover, the restriction on the time step imposed by WENO methods may spoil the advantages of the better stability property of CAT methods. To avoid this, ACAT methods [4] were introduced in which the oscillations near discontinuities are cured by adapting the order of accuracy – and thus the width of the stencils – to the smoothness of the solution. To do this, a new class of smoothness indicators was introduced.

Contrarily, in this work we operate a change of paradigm, by coupling the high-order CAT schemes with the *a posteriori* MOOD limiting procedure [8, 9, 10]. The fundamental idea at the heart of Multidimensional Optimal Order Detection (MOOD) revolves around the belief that it's more practical to evaluate and understand the adverse consequences resulting from the application of a high-order scheme, rather than trying to predict them in advance using predefined measures. A MOOD procedure is a sort of a 'try and error' loop. Consequently, within the MOOD loop, we utilize a comprehensive high-order explicit scheme to compute a preliminary numerical solution for the current time-step. This solution is then meticulously examined against specific criteria, determining the validity of different cells. The valid cells are preserved, while the invalid ones are reanalyzed using a less precise yet resilient approach – potentially a second-order limited scheme or, in more extreme situations, a robust first-order scheme. The central objective of this paper is to demonstrate a proof of concept that emphasizes the development and validation of a CAT-MOOD scheme tailored for systems governed by conservation laws, spotlighting the approach of evaluating

outcomes after the fact.

The remaining sections of this paper are structured as follows. In the second section, we provide an introduction to the governing equations. Following that, we delve into the LW procedure and introduce CAT schemes. Specifically, we revisit and derive the second-, fourth-, and sixth-order accurate versions of CAT scheme, in addition to outlining the generic CAT2P scheme of order $2P$, with P an arbitrary natural number along with its adaptive limiter (ACAT). Moving on, the fourth section elaborates on the integration of CAT schemes with the *a posteriori* MOOD procedure, effectively replacing the adaptive limiter. We detail the process of blending CAT schemes with the MOOD procedure to enhance numerical accuracy. In the subsequent fifth section, we present the numerical results obtained through simulations. These results are critical in evaluating and showcasing the robust performance of the CAT-MOOD sixth-order scheme, particularly in scenarios involving both smooth and discontinuous solutions. Finally, in the last section we summarize our findings, draw conclusions, and outline potential future directions and perspectives.

2. Governing equations

By essence this paper studies the application of CATMOOD6 scheme to the 2D Euler equations. However, to avoid complex 2D notation, it was decided to introduce the scheme in the simpler-to-read 1D case.

2.1. 1D linear and non-linear scalar conservation laws

To simplify the description of the numerical methods we also consider the non-linear scalar conservation law on the Oxt-Cartesian frame

$$u_t + \partial_x f(u) = 0, \tag{1}$$

where $u = u(x,t) : \mathbb{R} \times \mathbb{R}^+ \to \mathbb{R}$ denotes the scalar variable, and $f(u) = f(u(x,t))$ the non-linear flux depending on u. $u(x,0) = u_0(x)$ denotes the initial condition (IC), while the boundary condition (BC) are prescribed depending on the test case; for instance periodic ones, Dirichlet or Neumann ones.

(1) represents the generic model of non-linear scalar equation, the simplest one being probably Burgers' equation for which the flux is given by: $f(u) = u^2/2$. (1) also represents the generic model of linear scalar advection equation if $f(u) = au$ with $a \in \mathbb{R}$ being the advection velocity.
In the following we denote the partial derivative in time and space with under-script letters as $u_t \equiv \partial_t u$ and $u_x \equiv \partial_x u$.

2.2. 2D gas-dynamics system of conservation laws

In this paper we focus on hyperbolic systems of conservation laws (Partial Differential Equations, PDEs) in 1D and 2D of the form

$$\partial_t \mathbf{U} + \nabla \cdot \mathbb{F}(\mathbf{U}) = 0, \tag{2}$$

where $t \in \mathbb{R}^+$ represents the time variable, $\mathbf{x} = (x,y) \in \mathbb{R}^2$ the space variable in 2 dimensions. $\mathbf{U} = \mathbf{U}(\mathbf{x},t)$ is the vector of conserved variables while $\mathbb{F}(\mathbf{U}) = (\mathbf{F}(\mathbf{U}(\mathbf{x},t)), \mathbf{G}(\mathbf{U}(\mathbf{x},t)))^t$ is the flux vector. $\nabla\cdot$ is the divergence operator which allows us to rewrite (2) as

$$\partial_t \mathbf{U} + \partial_x \mathbf{F}(\mathbf{U}) + \partial_y \mathbf{G}(\mathbf{U}) = \mathbf{0}. \tag{3}$$

In this work we mainly focus on the gas-dynamics system of PDEs (Euler equations) where $\mathbf{U} = (\rho, \rho u, \rho v, \rho e)^\top$ with ρ the density, $\mathbf{u} = (u, v)$ the velocity vector, $e = \varepsilon + \frac{1}{2}\|\mathbf{u}\|^2$ the total energy per unit mass, and ε the internal one. The flux tensor is given by

$$\mathbb{F}(\mathbf{U}) = \begin{pmatrix} \mathbf{F}(\mathbf{U}) & \mathbf{G}(\mathbf{U}) \end{pmatrix}, \quad \text{with} \quad \mathbf{F}(\mathbf{U}) = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ (\rho e + p)u \end{pmatrix}, \quad \mathbf{G}(\mathbf{U}) = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ (\rho e + p)v \end{pmatrix}. \tag{4}$$

The system is closed by an Equation Of State (EOS) which specifies the value of the pressure p as a function of two thermodynamics variables, for instance of the form $p = p(\rho, \varepsilon)$. For a polytropic gas we have $p = (\Gamma - 1)\rho\varepsilon$ with Γ the polytropic constant characterizing the type of gas considered. The sound-speed is given by $a^2 = \Gamma p/\rho$. With the above mentioned choice, in this work, equations (2) express the conservation of mass, momentum and total energy. An entropy inequality is supplemented to the system of PDEs to ensure that the weak solutions are the entropic ones. This system is hyperbolic with eigenvalues $\lambda^- = \mathbf{u}\cdot\mathbf{n} - a$, $\lambda^0 = a$ (multiplicity 2), $\lambda^+ = \mathbf{u}\cdot\mathbf{n} + a$ where $\mathbf{n}$ is a generic unit vector indicating a direction in 2D. It is well known that the physical states all belong to

$$\mathcal{A} = \left\{ \mathbf{U} \in \mathbb{R}^4, \text{ such that } \rho > 0,\ p > 0 \right\}. \tag{5}$$

The primitive variables are the components of vector $\mathbf{W} = (\rho, u, v, p)^\top$ and are computed from the conservative ones as

$$u = (\rho u)/\rho, \qquad v = (\rho v)/\rho, \qquad p = (\Gamma - 1)\left((\rho E) - \frac{1}{2}\left((\rho u)^2 + (\rho v)^2 \right)/\rho \right). \tag{6}$$

System (2) is further equipped with Initial Conditions (IC) and Boundary Conditions (BC). This system of PDEs is the target one, but a simpler one is also considered in the next section for the sake of simplicity.

2.3. Mesh

In this article we consider logical rectangular meshes in 1D and 2D. The time domain $\mathcal{T} = [0, T]$ with final time $T > 0$ is split into time intervals $[t^n, t^{n+1}]$, $n \in \mathbb{N}$, and time-steps $\Delta t = t^{n+1} - t^n$ subject to a CFL (Courant-Friedrichs-Lewy) condition [28]. The computational domain denoted Ω is a segment/rectangle in 1D/2D. In 1D, Ω is paved with N_x cells. The generic cell is denoted ω_i and indexed by a unique label $1 \le i \le N_x$. Classically we identify the cell end-points by half indexes so that $\omega_i = [x_{i-1/2}, x_{i+1/2}]$ and the cell center is given by $x_i = \frac{1}{2}(x_{i+1/2} + x_{i-1/2})$. The size of a cell is given by $\Delta x_i = x_{i+1/2} - x_{i-1/2}$, that we simply denote as Δx, since we assume, for simplicity, that the grid is uniform. In 2D, Ω is paved with $N_x \times N_y$ cells. The generic cell is denoted $\omega_{i,\ell}$ and indexed by a double label $1 \le i \le N_x$ and $1 \le \ell \le N_y$. The four vertices of any cell $\omega_{i,\ell}$ are $\boldsymbol{x}_{i\pm1/2,\ell\pm1/2} = (x_{i\pm1/2}, y_{\ell\pm1/2})$. $x_{i+1/2}$ represents a vertical mesh line, while $y_{\ell+1/2}$ a horizontal one. Accordingly, $\omega_{i,\ell} = [x_{i-1/2}, x_{i+1/2}] \times [y_{\ell-1/2}, y_{\ell+1/2}]$, and, the cell center is given by $\boldsymbol{x}_{i,\ell} = (x_i, y_\ell) = \left(\frac{x_{i+1/2}+x_{i-1/2}}{2}, \frac{y_{\ell+1/2}+y_{\ell-1/2}}{2} \right)$. The size of a cell is given by $\Delta x_i \times \Delta y_\ell$ with $\Delta y_\ell = y_{\ell+1/2} - y_{\ell-1/2}$, that we simply denote as Δx and Δy, since we shall adopt a uniform mesh throughout the paper. We call an interface or face, the intersection between two neighbor cells, that is a point in 1D and a segment in 2D. The neighbor cells of a generic one are those with a non-empty intersection. A generic cell has two/eight neighbors in 1D/2D on logical rectangular grids. In 2D we make the difference between the four face-neighbors and the four corner-neighbors. A "stencil" in 1D is a collection of $K > 0$ cells surrounding and including the current one, onto which derivatives are approximated, see Figure 1 for a graphical view.

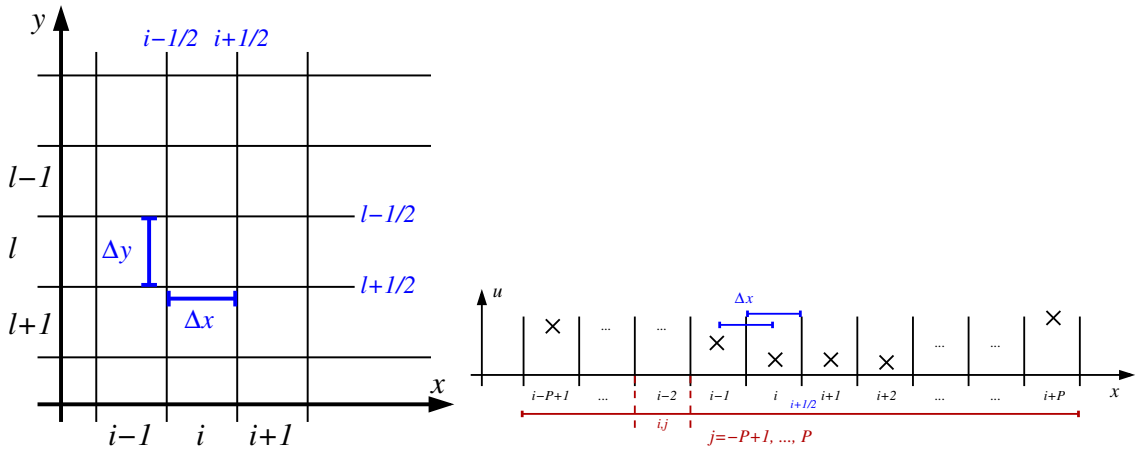


Figure 1: Left: Logical rectangular grid used in this paper. Right: illustration of spatial stencil in 1D around cell i with stencils i, j with $-P+1 \leq j \leq P$.

2.4. Notation

In this article the scheme description is made mostly in 1D for the sake of clarity. The following notations refer to the different type of derivatives or approximations.

- $u_{i,j}^{(k)}$ is the *local*[1] approximation of the k-th time derivative of u at time t^n in position x_{i+j}, where i refers to the stencil $S_{i+\frac{1}{2}}^P$ (19) centered at cell x_i, and, j to the position in the stencil. In general, for a scheme of order $2P$, $k = 1, \ldots, 2P-1$ and $j = -P+1, \ldots, P$.
- $f_{i,j}^{(k)}$ is the k-th time derivative of $f(u)$ at time t^n in position x_{i+j}, likewise for u. In general, for a scheme of order $2P$, $k = 1, \ldots, 2P-1$ and $j = -P+1, \ldots, P$. The notation $f_{i,j}^{(0)} := f(u_{i+j}^n)$ will also be used.
- $u_{i,j}^{k,n+r}$ is the explicit Taylor expansion of function u in time truncated to order k, centered at time t^n at distance $r\Delta t$ in time and at spatial location x_{i+j}. Again i refers to the cell and j to the position in the stencil. In general, for a scheme of order $2P$, it is $k = 1, \ldots, 2P-1$, while $j, r = -P+1, \ldots, P$.
- $f_{i,j}^{k,n+r}$ refers to $f\left(u_{i,j}^{k,n+r}\right)$.
- The symbol $A_P^{k,q}(\varphi, \Delta x)$ is used for the interpolatory numerical differentiation formula that approximates the k-th space derivative of an arbitrary function $\varphi(x, y)$ at $x = x_i + q\Delta x$ with $q \in \mathbb{R}$ and $t = t_n$, using its values at the $2P$ points x_{i+j}, $j = -P+1, \ldots, P$. This formula can be written in the form:

$$\partial_x^k \varphi(x_i + q\Delta x, t_n) \approx A_P^{k,q}(\varphi, \Delta x) = \frac{1}{\Delta x^k} \sum_{j=-P+1}^{P} \gamma_{P,j}^{k,q} \varphi(x_{i+j}, t_n),$$

[1] In this context, by local we mean that if we fix the stencil i_1 and i_2, with $i_1 \neq i_2$, and also fix j_1 and j_2 such that $i_1 + j_1 = i_2 + j_2$, then generally it holds that $u_{i_1+j_1}^{(k)} \neq u_{i_2+j_2}^{(k)}$, see Remark 2 on page 10.

where $\gamma_{P,j}^{k,p}$ are the coefficients of the formula. For $k = 0$, it represents Lagrange interpolation. A similar notation will be used for the approximation of time derivatives:

$$\partial_t^k \varphi(x_i, t_n) \approx A_P^{k,0}(\varphi, \Delta t) = \frac{1}{\Delta t^k} \sum_{r=-P+1}^{P} \gamma_{P,r}^{k,0} \varphi(x_i, t_{n+r}).$$

Observe that, unlike for space derivatives, only centered formulas will be used for time derivatives, i.e. $q = 0$.

- For functions that depends on both x and t, the symbol $*$ will be used to indicate to which index (space or time) the differentiation is applied as illustrated in the equations below for the approximation of space and time derivatives, respectively (see also (21))

$$\partial_x^k f(x_i + q\Delta x, t^n) \approx A_P^{k,q}(f_{i,*}^{(0)}, \Delta x) = \frac{1}{\Delta x^k} \sum_{j=-P+1}^{P} \gamma_{P,j}^{k,q} f_{i,j}^{(0)}, \tag{7}$$

$$\partial_t^k f(x_i, t^n) \approx A_P^{k,0}(f_i^{k,*}, \Delta t) = \frac{1}{\Delta t^k} \sum_{r=-P+1}^{P} \gamma_{P,r}^{k,0} f_i^{k,n+r}. \tag{8}$$

From now on, since all the formulas are computed at time $t = t^n$, we avoid the extra index n on equation like (7). A similar notation is adopted for 2D problems with the necessary precautions.

3. Compact Approximate Taylor (CAT) schemes

The focus of this chapter is the presentation of a family of numerical methods for non-linear systems of conservation laws, named Compact Approximate Tayor (CAT) schemes[2]. This family is based on an approximate Taylor procedure that constitutes a proper generalization of Lax-Wendroff (LW) method, in the sense that it reduces to the standard high-order LW method when the flux is linear. In this section we recall the LW and CAT procedures of second and fourth order.

3.1. Lax Wendroff procedure

A scheme of historical as well as practical importance is the celebrated **Lax Wendroff** scheme introduced by Peter Lax and Burton Wendroff in 1960 in [19], and [21, 40, 20, 17, 38]. It has been the most widely adopted scheme for aeronautical applications, up to the end of the 1980s under various forms. Probably the most widely used variant is the two-step Mac Cormack scheme, which has similar properties than LW schemes, and avoids the computation of the second derivative. Published originally at a conference in 1969, the paper has been reproduced in [29].

The original derivation of Lax and Wendroff was based on a Taylor expansion in time of function u at point (x_i, t) up to second order of accuracy, thus

$$u(x_i, t + \Delta t) = u(x_i, t) + \Delta t\, u_t(x_i, t) + \frac{\Delta t^2}{2} u_{tt}(x_i, t) + O(\Delta t^3), \tag{9}$$

where $\Delta t > 0$ is a small increment in time. The numerical scheme is then obtained by neglecting the higher order term in Δt, using the governing equation to replace time derivatives by spatial ones,

[2]For simplicity we will summarize the numerical deduction of the method in the scalar case.

and then substituting the obtained space derivatives with their finite difference approximations. For the *linear case*, $u_t = -au_x$ with $f(u) = au$, we obtain

$$u(x_i, t^n + \Delta t) = u(x_i, t^n) - \Delta t\, a u_x(x_i, t^n) + \frac{\Delta t^2}{2} a^2 u_{xx}(x_i, t) + O(\Delta t^3). \tag{10}$$

Using centred finite differences to approximate spatial derivatives, the numerical scheme follows as:

$$u_i^{n+1} = u_i^n - \Delta t\, a\, \frac{u_{i+1}^n - u_{i-1}^n}{2\Delta x} + \frac{\Delta t^2}{2} a^2 \frac{u_{i+1}^n - 2u_i^n + u_{i-1}^n}{\Delta x^2}, \tag{11}$$

where u_i^n is an approximation of the point value of the solution at position x_i at the time t^n. A useful alternative formulation written in conservative form yields

$$u_i^{n+1} = u_i^n - \frac{\Delta t}{\Delta x}\left(F_{i+\frac{1}{2}}^{\mathrm{LW}} - F_{i-\frac{1}{2}}^{\mathrm{LW}}\right), \tag{12}$$

where the so-called LW numerical flux, $F_{i+\frac{1}{2}}^{\mathrm{LW}}$, is given by

$$F_{i+\frac{1}{2}}^{\mathrm{LW}} = \frac{a}{2}\left(u_{i+1}^n + u_i^n\right) - \frac{a^2 \Delta t}{2\Delta x}\left(u_{i+1}^n - u_i^n\right). \tag{13}$$

The *non-linear case*, $u_t = -f_x(u)$, yields

$$u_i^{n+1} = u_i^n - \frac{\Delta t}{2\Delta x}\left(f_{i+1}^n - f_{i-1}^n\right) + \frac{\Delta t^2}{2\Delta x^2}\left(\mathrm{A}_{i+\frac{1}{2}}\left(f_{i+1}^n - f_i^n\right) - \mathrm{A}_{i-\frac{1}{2}}\left(f_i^n - f_{i-1}^n\right)\right), \tag{14}$$

where $f_{i+l}^n = f(u_{i+l}^n)$ for $l = -1, 0, 1$ and A is an approximation of the derivative of f, i.e $A = \partial f / \partial u$. Hence, $\mathrm{A}_{i\pm\frac{1}{2}}$ is the approximation derivative of f evaluated at $u_{i+\frac{1}{2}}^n = \frac{1}{2}(u_i^n + u_{i\pm1}^n)$, that is $\mathrm{A}_{i\pm\frac{1}{2}} \equiv \mathrm{A}(u_{i+\frac{1}{2}}^n)$, or, the average between the cell-based derivative, that is $\mathrm{A}_{i\pm\frac{1}{2}} \equiv \frac{1}{2}\left(\mathrm{A}(u_i^n) + \mathrm{A}(u_{i\pm1}^n)\right)$. Notice that they depend non-linearly on variable u. Moreover they are always evaluated at time t^n, so we can omit this time dependency. The alternative conservative formulation of the LW scheme is expressed as:

$$u_i^{n+1} = u_i^n - \frac{\Delta t}{\Delta x}\left(F_{i+\frac{1}{2}}^{\mathrm{LW}} - F_{i-\frac{1}{2}}^{\mathrm{LW}}\right), \tag{15}$$

where the numerical flux

$$F_{i+\frac{1}{2}}^{\mathrm{LW}} = \underbrace{\frac{1}{2}\left(f_{i+1}^n + f_i^n\right)}_{\text{Physical flux}} \underbrace{- \frac{\Delta t}{2\Delta x} A_{i+\frac{1}{2}}\left(f_{i+1}^n - f_i^n\right)}_{\text{Dissipation}}, \tag{16}$$

is composed of two parts: the average of the physical fluxes at cells i and $i+1$, and, the numerical dissipation.

3.2. Compact Approximate Taylor (CAT) procedure

The generalized Lax-Wendroff method is used to update the numerical solution:

$$u_i^{n+1} = u_i^n + \sum_{k=1}^{2P} \frac{(\Delta t)^k}{k!} u_i^{(k)}, \tag{17}$$

where we recall that u_i^n is an approximation of the value of the exact solution $u(x,t)$ at time t^n at position x_i [16], and, $u_i^{(k)}$ is an approximation of $\partial_t^k u(x_i, t^n)$. The k-th derivative in time of u is

computed with a compact and numerical version of the Cauchy-Kovaleskaya procedure introduced by Carrillo and Parés in [5].
The final expression of the $2P$-order CAT method in conservative form is:

$$u_i^{n+1} = u_i^n + \frac{\Delta t}{\Delta x}\left(F^P_{i-\frac{1}{2}} - F^P_{i+\frac{1}{2}}\right). \tag{18}$$

The above equation is a finite difference-type scheme: u_i^n is a high order approximation of the point-wise value of the exact solution $u(x_i, t^n)$, at variance with finite volume schemes where the numerical solution is an approximation to cell averages (see [33]). Let us introduce the sets $\mathcal{S}^P_{i\pm\frac{1}{2}}$ of values u_i^n on stencils centered around interface $i \pm \frac{1}{2}$ of size $2P$, that is

$$\mathcal{S}^P_{i+\frac{1}{2}} = \left\{u^n_{i-P+1}, \dots, u^n_i, u^n_{i+1}, \dots u^n_{i+P}\right\}. \tag{19}$$

The flux functions $F^P_{i\pm\frac{1}{2}}$ are then computed, respectively, on the sets $\mathcal{S}^P_{i\pm\frac{1}{2}}$, as

$$F^P_{i+\frac{1}{2}} = \sum_{k=1}^{2P} \frac{\Delta t^{k-1}}{k!} f^{(k-1)}_{i+\frac{1}{2}}, \tag{20}$$

and

$$f^{(k-1)}_{i+\frac{1}{2}} = A^{0,\frac{1}{2}}_P\left(f^{(k-1)}_{i,*}, \Delta x\right), \quad \text{with} \quad A^{0,\frac{1}{2}}_P\left(f^{(k-1)}_{i,*}, \Delta x\right) = \sum_{p=-P+1}^{P} \gamma^{0,\frac{1}{2}}_{P,p}\, f^{(k-1)}_{i+p}, \tag{21}$$

where $A^{0,\frac{1}{2}}_P$ is an interpolation formula of order $2P-1$ based on $2P$-point stencil. In the following we use the index i for the cell global index, j for the local position inside the stencil, r for the Taylor expansion in time, and, $k/(k)$ to refer to the k-th time step/(k)th time derivative. For the sake of clarity, we detail the description of the second order ($P = 1$) CAT2, in the next sub-section, and CAT4 in the Appendix 7.1.

3.2.1. Second order version – CAT2

Let $p \in \mathbb{N}$ denote an integer such that $0 \le p \le P$. In the case $P = 1$ then the relative stencil is simply $\mathcal{S}^1_{i+\frac{1}{2}} = \{u^n_i, u^n_{i+1}\}$ for interface $i + 1/2$, while the flux reconstruction is:

$$F^1_{i+\frac{1}{2}} = f^{(0)}_{i+\frac{1}{2}} + \frac{\Delta t}{2} f^{(1)}_{i+\frac{1}{2}}, \tag{22}$$

where $f^{(0)}_{i+\frac{1}{2}} = \frac{1}{2}\left(f^n_i + f^n_{i+1}\right)$ is the interpolation of the flux at time t^n, while $f^{(1)}_{i+\frac{1}{2}}$ is the interpolation of the first time-derivative of the flux for any interface $i + 1/2$. This implies that $\gamma^{0,\frac{1}{2}}_{1,p} = \frac{1}{2}$ in (21) for all p. These are computed over stencils $\mathcal{S}^1_{i+\frac{1}{2}}$ for $p \in \{0, 1\}$ as

$$\mathcal{S}^1_{i+\frac{1}{2}}: \qquad f^{(1)}_{i+\frac{1}{2}} = \frac{f^{(1)}_{i,0} + f^{(1)}_{i,1}}{2}, \quad f^{(1)}_{i,p} = \frac{f\left(u^n_{i+p} + \Delta t\, u^{(1)}_{i,p}\right) - f^n_{i+p}}{\Delta t}, \qquad u^{(1)}_{i,p} = -\frac{f^n_{i+1} - f^n_i}{\Delta x}.$$

Finally, the expanded form of the fluxes (22) is given by

$$F^1_{i+\frac{1}{2}} = \frac{1}{4}\left(f^n_i + f^n_{i+1} + f\left(u^n_i + \Delta t\, u^{(1)}_{i,0}\right) + f\left(u^n_{i+1} + \Delta t\, u^{(1)}_{i,1}\right)\right), \tag{23}$$

and the solution is obtained by substituting these fluxes in formula (18).
The computation of the interfacial flux $F^1_{i+\frac{1}{2}}$ can be recast into the algorithm:

Step 1: Compute $f^{(0)}_{i+\frac{1}{2}}$ adopting an interpolation formula over stencil $\mathcal{S}^1_{i+\frac{1}{2}}$ at time t^n.

Step 2: Compute the first derivatives $u^{(1)}_{i,p}$ in time through the numerical compact Cauchy-Kovalesky procedure, using $\partial_t u = -\partial_x f$, with data at time t^n.

Step 3: Compute the truncated Taylor expansions: $u^{1,n+1}_{i,p} = u^n_{i+p} + \Delta t\, u^{(1)}_{i,p}$ for $p = 0$ and 1.

Step 4: Compute the first time derivatives of the flux using the first difference formulas:

$$f^{(1)}_{i,p} = \frac{f\left(u^{1,n+1}_{i,p}\right) - f^n_{i+p}}{\Delta t}.$$

Step 5: Compute $f^{(1)}_{i+\frac{1}{2}}$ through $f^{(1)}_{i,j}$ adopting an interpolation formula on stencil $\mathcal{S}^1_{i+\frac{1}{2}}$;

Step 6: Compute $F^1_{i+\frac{1}{2}}$ as a Taylor expansion: $F^1_{i+\frac{1}{2}} = f^{(0)}_{i+\frac{1}{2}} + \dfrac{\Delta t}{2} f^{(1)}_{i+\frac{1}{2}}$ with (22).

3.2.2. CAT2P

Generically, the $2P$ CAT scheme follows formulation (18) with the interface fluxes given by (20). The expression of the right numerical flux of order $2P$ is obtained with formula:

$$F^P_{i+\frac{1}{2}} = \sum_{k=1}^{2P} \frac{\Delta t^{k-1}}{k!} A_P^{0,\frac{1}{2}}\left(f^{(k-1)}_{i,*}, \Delta x\right) = \sum_{k=1}^{2P} \frac{\Delta t^{k-1}}{k!} \sum_{j=-P+1}^{P} \gamma^{0,\frac{1}{2}}_{P,j} f^{(k-1)}_{i,j}, \tag{24}$$

where the high order time derivatives of the flux are computed following and extending the iterative algorithm presented for CAT2 in the previous section (see also [5, 4, 3] for more details):

1. Define $f^{(0)}_{i,j} := f(u^n_{i+j})$ for all $j = -P+1, \ldots, P$;
2. For every $k = 1, \ldots, 2P-1$:
   (a) Compute the k-th derivative of u at time step t^n for each position x_{i+j} with $j = -P+1, \ldots, P$ through the numerical compact version of the Cauchy-Kovalesky identity Cauchy-Kovaleskaya identity
   $$\partial^k_t u = -\partial_x f^{(k-1)}(u), \quad k = 1, 2, \ldots$$
   as:
   $$u^{(k)}_{i,j} = -A^{1,j}_P\left(f^{(k-1)}_{i,*}, \Delta x\right);$$
   (b) Compute the Taylor expansion of u in time truncated to term k for all positions x_{i+j} with $j = -P+1, \ldots, P$ at time t^{n+r} with $r = -P+1, \ldots, P$ as:
   $$u(x_{i+j}, t^{n+r}) \approx u^{k,n+r}_{i,j} = u^n_{i+j} + \sum_{m=1}^{k} \frac{(r\Delta t)^m}{m!} u^{(m)}_{i,j};$$
   (c) Compute the $k-$th time derivative of flux for each position x_{i+j} with $j = -P+1, \ldots, P$ at time t^n as:
   $$f^{(k)}_{i,j} = A^{k,j}_P\left(f^{k,*}_{i,j}, \Delta t\right),$$
   where $f^{k,*}_{i,j}$ means that we are applying the A operator in time and in particular we apply the differentiation formula to the set of flux approximations
   $$f^{k,n-P+1}_{i,j}, \ldots, f^{k,n+P}_{i,j},$$
   in which $f^{k,n+r}_{i,j} = f\left(u^{k,n+r}_{i,j}\right)$ for all $j, r = -P+1, \ldots, P$.

Remark 1. *Observe that the computation of the numerical flux $F^P_{i+1/2}$ requires the approximation of u at the nodes of a space-time grid of size $2P\times 2P$, represented by $u^{k,n+r}_{i,j}$, for $-P+1 \leq j, r =\leq P$ (see Figure 19). Please note that the only information used to compute these values is given by u^n_{i+j}, $j = -P+1, \ldots, P$, so that only $2P$ points are used to compute $F^P_{i+1/2}$ and thus the stencil size is $2P+1$. The approximations of the solution u at successive times $(n-P+1)\Delta t$, $\ldots$, $(n-1)\Delta t$ are different from the ones already computed in the previous time steps $t^{n-P}, \ldots, t^{n-1}$ which are u^{n-P}_{i+j}, $\ldots$, u^{n-1}_{i+j}. In other words, the discretization in time is not based on a multi-step method but on a one-step one. In fact, it can be re-interpreted as a Runge-Kutta method whose stages are $u^{n+r}_{i,j}$, $r = -P+1, \ldots, P$, see [5, 4, 3].*

Remark 2. *These approximations are local. Indeed, suppose that $i_1+j_1 = i_2+j_2 = \ell$, i.e. $x_\ell > 0$ belongs to $\mathcal{S}^P_{i_1+1/2}$ and $\mathcal{S}^P_{i_2+1/2}$ with local coordinates j_1 and j_2 respectively. Then, $f^{(k)}_{i_1,j_1}$ and $f^{(k)}_{i_2,j_2}$ are, in general, two different approximations of $\partial^k_t f(u)(x_\ell, t^n)$.*

3.2.3. Computational complexity

In this paragraph we estimate the computational complexity for the CAT2P scheme. The details about the complexity of the algorithm can be found in Appendix 7.2. As a summary, the operation count per cell per time step for CAT2P, $P > 1$, applied to the scalar case is $3.5(2P)^3 - 1.5(2P)^2 + (2P)$ flop plus $(2P)^3 - 2(2P)^2 + (2P) + 1$ function evaluations. For CAT2 this gives 24 flop and 3 function evaluations, while for CAT4 we get 204 flop and 37 function evaluations.

3.3. Adaptive limiter for CAT schemes - ACAT schemes

Although the Compact Approximate Taylor (CAT) schemes are linearly stable in the L^2-sense under the usual CFL-1 condition, they may produce bounded oscillations close to the discontinuity of the solution. To avoid these spurious phenomena, an Adaptive *a priori* shock-capturing technique has been developed in [4] and called ACAT. There, the order of the method is locally adapted to the smoothness of the numerical solution by means of indicators which check the regularity of the data for each temporal step. More specifically, once the approximations of the solution u at time t^n have been computed, the stencil of the data adopted to actually compute the right flux $F^P_{i+1/2}$ are set to belong to

$$\mathcal{S}^p_{i+1/2} = \{u^n_{i-p+1}, \ldots, u^n_{i+p}\}, \quad p = 1, \ldots, P.$$

The selected stencil is the one with maximal length among those in which the solution at time t^n is 'smooth'. The smoothness is assessed according to some smoothness indicators: $\psi^p_{i+1/2}$, for any $p = 1, \ldots, P$, which are defined as:

$$\psi^p_{i+1/2} \approx \begin{cases} 1 & \text{if } u \text{ is 'smooth' in } \mathcal{S}^p_{i+1/2}, \\ 0 & \text{otherwise.} \end{cases} \tag{25}$$

For this strategy one needs to define a robust first-order flux reconstruction, for instance Rusanov-, HLL- or HLLC-based flux reconstruction [38, 22]. Next, one can employ a TVD flux-limiter for the second-order flux reconstruction to be combined with CAT2, such as, for instance minmod, van Alabada or superbee [22].

ACAT2. The expression of the ACAT2 numerical method (for $P = 1$) based on a flux limiter (see [22, 23, 38]) is given by

$$u^{n+1}_i = u^n_i + \frac{\Delta t}{\Delta x}\left(F^*_{i-1/2} - F^*_{i+1/2}\right), \tag{26}$$

where the fluxes $F^*_{i+1/2}$ are blended as

$$F^*_{i\pm1/2} \quad = \quad \varphi^1_{i\pm1/2}\, F^1_{i\pm1/2} + (1-\varphi^1_{i\pm1/2})\, F^{\text{low}}_{i\pm1/2}. \tag{27}$$

$F^1_{i+1/2}$ is the CAT2 flux given by (23), while $F^{\text{low}}_{i\pm1/2}$ is the first order flux reconstruction, and, $\varphi^1_{i\pm1/2}$ is a switch computed by a flux limiter which verifies

$$\varphi^1_{i-1/2} \approx \begin{cases} 1 & \text{if } \{u^n_{i-2},\dots,u^n_{i+1}\} \text{ is 'smooth'},\\ 0 & \text{otherwise,}\end{cases} \qquad \varphi^1_{i+1/2} \approx \begin{cases} 1 & \text{if } \{u^n_{i-1},\dots,u^n_{i+2}\} \text{ is 'smooth'},\\ 0 & \text{otherwise.}\end{cases}$$

For scalar problems, standard flux limiter functions, $\varphi^1(r)$, such as minmod, superbee, van Leer [35, 18], may be used:

$$\varphi^1_{i+1/2} = \varphi^1(r_{i+1/2}), \tag{28}$$

where

$$r_{i+1/2} = \begin{cases} r^L_{i+1/2} = \dfrac{u^n_i - u^n_{i-1}}{u^n_{i+1} - u^n_i} & \text{if } a_{i+1/2} > 0,\\ r^R_{i+1/2} = \dfrac{u^n_{i+2} - u^n_{i+1}}{u^n_{i+1} - u^n_i} & \text{if } a_{i+1/2} \le 0,\end{cases} \qquad a_{i+1/2} = \begin{cases} \dfrac{f(u^n_{i+1}) - f(u^n_i)}{u^n_{i+1} - u^n_i} & \text{if } |u^n_i - u^n_{i+1}| > \varepsilon,\\ f'(u^n_i) & \text{otherwise,}\end{cases} \tag{29}$$

where $a_{i+1/2}$ is an approximation of the wave speed such as Roe's intermediate speed and ε is a small number (but not too small to avoid numerical cancellation). An alternative procedure introduced in [38] avoids the computation of an intermediate speed by defining $\varphi^1_{i+1/2} = \min(\varphi^1(r^R_{i+1/2}), \varphi^1(r^L_{i+1/2}))$. This strategy is easily extended to systems by computing the flux limiter component by component.

Smoothness indicators. The smoothness indicators used in this work have been introduced in [4]. Nonetheless we briefly recall their construction for the sake of completeness. Given a set of the point values f_j of a function f in the nodes of the stencil $\mathcal{S}^p_{i+1/2}$, $p \ge 2$ we define $\psi^p_{i+1/2}$ as follows. First define the lateral left (^{L}) and right (^{R}) weights $w^{p,L/R}_{i+1/2}$ as

$$w^{p,L}_{i+1/2} := \sum_{j=-p+1}^{-1} (f_{i+1+j} - f_{i+j})^2 + \varepsilon, \quad w^{p,R}_{i+1/2} := \sum_{j=1}^{p-1} (f_{i+1+j} - f_{i+j})^2 + \varepsilon, \tag{30}$$

where $\varepsilon = 10^{-8}$ is a small quantity only used to prevent the weights to vanish. Next, using the half harmonic mean, $w^p_{i+1/2} = \frac{w^{p,L}_{i+1/2} w^{p,R}_{i+1/2}}{w^{p,L}_{i+1/2} + w^{p,R}_{i+1/2}}$, one defines the high order smoothness indicator over stencil $\mathcal{S}^p_{i+1/2}$ by

$$\psi^p_{i+1/2} := \left(\frac{w^p_{i+1/2}}{w^p_{i+1/2} + \tau^p_{i+1/2}} \right), \tag{31}$$

where $\tau^p_{i+1/2} = (\Delta^{2p-1}_{i-p+1} f)^2$, and $\Delta^{2p-1}_{i-p+1} f$ is the undivided difference or order $2p-1$ of the $2p$ values $\{f_{i-p+1},\dots,f_{i+p}\}$ which is related to the γ coefficients by the formula

$$(2p-1)!\Delta^{2p-1}_{i-p+1} f = \sum_{j=-p+1}^{p} \gamma^{2p-1,1/2}_{p,j}\, f^n_{i+j}.$$

These indicators are such that

$$\psi^p_{i+1/2} \approx \begin{cases} 1 & \text{if } \{f_j\} \text{ are 'smooth' in } \mathcal{S}^p_{i+1/2}, \\ 0 & \text{otherwise,} \end{cases} \tag{32}$$

see [4] for a precise statement of this property and its proof.

Remark 3. *Observe that, if data in the stencil $\mathcal{S}^p_{i+1/2}$ are smooth, then $w^{p,L}_{i+1/2} = O(\Delta x^2)$, $w^{p,R}_{i+1/2} = O(\Delta x^2)$, and $\tau^p_{i+1/2} = O(\Delta x^{4p-2})$. Since*

$$\frac{1}{w^p_{i+1/2}} = \frac{1}{w^{p,L}_{i+1/2}} + \frac{1}{w^{p,R}_{i+1/2}},$$

then $w^p_{i+1/2} = O(\Delta x^2)$. As such

$$\psi^p_{i+1/2} = \frac{w^p_{i+1/2}}{w^p_{i+1/2} + \tau^p_{i+1/2}} = \frac{O(\Delta x^2)}{O(\Delta x^2) + O(\Delta x^{4p-2})},$$

so that $\psi^p_{i+1/2}$ is expected to be close to 1. On the other hand, if there is an isolated discontinuity in the stencil then $\tau^p_{i+1/2} = O(1)$, therefore one of the lateral weights is $O(1)$ and the other $O(\Delta x^2)$ so that the harmonic mean implies that $w^p_{i+1/2} = O(\Delta x^2)$ and thus:

$$\psi^p_{i+1/2} = \frac{w^p_{i+1/2}}{w^p_{i+1/2} + \tau^p_{i+1/2}} = \frac{O(\Delta x^2)}{O(\Delta x^2) + O(1)},$$

so that $\psi^p_{i+1/2}$ is expected to be close to 0.

ACAT2P schemes. Using these ingredients, the final expression of the ACAT2P for $P > 1$ scheme is of the form

$$u_i^{n+1} = u_i^n + \frac{\Delta t}{\Delta x}\left(F_{i-\frac{1}{2}}^{\mathcal{P}_{i-1/2}} - F_{i+\frac{1}{2}}^{\mathcal{P}_{i+1/2}}\right), \tag{33}$$

where

$$F_{i\pm1/2}^{\mathcal{P}_{i\pm1/2}} = \begin{cases} F^*_{i\pm1/2} & \text{if } \mathcal{P}_{i\pm1/2} = \emptyset; \\ F^{p_{\max}}_{i\pm1/2} & \text{otherwise.} \end{cases} \tag{34}$$

Here, $\mathcal{P}_{i+1/2}$ is the set of consecutive indices

$$\mathcal{P}_{i+1/2} = \{p \in \{2, \dots, P\} \text{ s.t. } \psi^p_{i+1/2} \approx 1\}, \quad \text{and} \quad p_{\max} = \max(\mathcal{P}_{i+1/2}). \tag{35}$$

Moreover $F^*_{i+1/2}$ is the ACAT2 second-order numerical flux given by (27), $F^{p_{\max}}_{i+1/2}$ is the ACAT2$p_{\max}$ numerical fluxes defined in (24). If $\mathcal{P}_{i+1/2} \neq \emptyset$, the order of the flux $F^{p_{\max}}_{i+1/2}$ can range from $2P$ to 4. Throughout the paper we clip ψ to 1 if $\psi \geq 0.95$.

Remark 4. *Notice that in (35) index p starts from 2 since it is not possible to determine the smoothness of the data in the two-point stencil $\mathcal{S}^1_{i+1/2}$.*

3.4. Extension to 2D

In this section we focus on the extension of CAT methods to non-linear two-dimensional systems of hyperbolic conservation laws

$$u_t + f(u)_x + g(u)_y = 0. \tag{36}$$

The following multi-index notation will be used:

$$\mathbf{i} = (i_1, i_2) \in \mathbb{Z} \times \mathbb{Z},$$

and

$$\mathbf{0} = (0,0), \quad \mathbf{1} = (1,1), \quad \frac{\mathbf{1}}{\mathbf{2}} = (\frac{1}{2}, \frac{1}{2}), \quad \mathbf{e}_1 = (1,0), \quad \mathbf{e}_2 = (0,1).$$

We consider Cartesian meshes with nodes

$$\mathbf{x}_\mathbf{i} = (i_1 \Delta x, i_2 \Delta y).$$

Using this notation, the general form of the CAT2P method will be as follows:

$$u_\mathbf{i}^{n+1} = u_\mathbf{i}^n + \frac{\Delta t}{\Delta x}\left[F^P_{\mathbf{i}-\frac{1}{2}\mathbf{e}_1} - F^P_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1}\right] + \frac{\Delta t}{\Delta y}\left[G^P_{\mathbf{i}-\frac{1}{2}\mathbf{e}_2} - G^P_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2}\right], \tag{37}$$

where the numerical fluxes $F^P_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1}$, $G^P_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2}$ will be computed using the values of the numerical solution $U^n_\mathbf{i}$ in the P^2-point stencil centered at $\mathbf{x}_{\mathbf{i}+\frac{\mathbf{1}}{\mathbf{2}}} = ((i_1 + \frac{1}{2})\Delta x, (i_2 + \frac{1}{2})\Delta y)$

$$S^P_{\mathbf{i}+\frac{\mathbf{1}}{\mathbf{2}}} = \{\mathbf{x}_{\mathbf{i}+\mathbf{j}}, \quad \mathbf{j} \in \mathcal{I}_P\},$$

where

$$\mathcal{I}_P = \{\mathbf{j} = (j_1, j_2) \in \mathbb{Z} \times \mathbb{Z}, \quad -P+1 \le j_k \le P, \quad k = 1,2\}.$$

3.4.1. 2D CAT2

In order to show the extension of CAT2P procedure let us start with the expression of the CAT2. The numerical fluxes are constructed as follows:

$$F^1_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1} = \frac{1}{4}\left(f^{1,n+1}_{\mathbf{i},\mathbf{0}} + f^{1,n+1}_{\mathbf{i},\mathbf{e}_1} + f^n_\mathbf{i} + f^n_{\mathbf{i}+\mathbf{e}_1}\right), \tag{38}$$

$$G^1_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2} = \frac{1}{4}\left(g^{1,n+1}_{\mathbf{i},\mathbf{0}} + g^{1,n+1}_{\mathbf{i},\mathbf{e}_2} + g^n_\mathbf{i} + g^n_{\mathbf{i}+\mathbf{e}_2}\right), \tag{39}$$

where

$$f^{1,n+1}_{\mathbf{i},\mathbf{j}} = f\left(u^n_{\mathbf{i}+\mathbf{j}} + \Delta t\, u^{(1)}_{\mathbf{i},\mathbf{j}}\right),$$
$$g^{1,n+1}_{\mathbf{i},\mathbf{j}} = g\left(u^n_{\mathbf{i}+\mathbf{j}} + \Delta t\, u^{(1)}_{\mathbf{i},\mathbf{j}}\right),$$

for $\mathbf{j} = \mathbf{0}, \mathbf{e}_1$ in the x direction, and $\mathbf{j} = \mathbf{0}, \mathbf{e}_2$ in the y direction.

Remark 5. *Despite what happens for the 1D reconstruction, the first time derivative of u, $u^{(1)}_{\mathbf{i},\mathbf{j}}$, does not coincide in the 2D-grid points. Indeed, observe that $u^{(1)}_{\mathbf{i},\mathbf{0}} \neq u^{(1)}_{\mathbf{i},\mathbf{e}_1}$ and $u^{(1)}_{\mathbf{i},\mathbf{0}} \neq u^{(1)}_{\mathbf{i},\mathbf{e}_2}$.*

Note that, in the 1D case, $u^{(1)}_{i,0} = u^{(1)}_{i,1}$.

Hence, the first time derivatives $u^{(1)}_{\mathbf{i},\mathbf{j}}$ are so defined:

$$u^{(1)}_{\mathbf{i},\mathbf{0}} = -\frac{1}{\Delta x}\left(f^n_{\mathbf{i}+\mathbf{e}_1} - f^n_{\mathbf{i}}\right) - \frac{1}{\Delta y}\left(g^n_{\mathbf{i}+\mathbf{e}_2} - g^n_{\mathbf{i}}\right),$$
$$u^{(1)}_{\mathbf{i},\mathbf{e}_1} = -\frac{1}{\Delta x}\left(f^n_{\mathbf{i}+\mathbf{e}_1} - f^n_{\mathbf{i}}\right) - \frac{1}{\Delta y}\left(g^n_{\mathbf{i}+\mathbf{1}} - g^n_{\mathbf{i}+\mathbf{e}_1}\right),$$
$$u^{(1)}_{\mathbf{i},\mathbf{e}_2} = -\frac{1}{\Delta x}\left(f^n_{\mathbf{i}+\mathbf{1}} - f^n_{\mathbf{i}+\mathbf{e}_2}\right) - \frac{1}{\Delta y}\left(g^n_{\mathbf{i}+\mathbf{e}_2} - g^n_{\mathbf{i}}\right),$$

where

$$f^n_{\mathbf{i}+\mathbf{j}} = f(u^n_{\mathbf{i}+\mathbf{j}}), \quad g^n_{\mathbf{i}+\mathbf{j}} = g(u^n_{\mathbf{i}+\mathbf{j}}), \quad \forall \mathbf{j}.$$

Finally, the 2D CAT2 method is so defined:

$$u^{n+1}_{\mathbf{i}} = u^n_{\mathbf{i}} + \frac{\Delta t}{\Delta x}\left[F^1_{\mathbf{i}-\frac{1}{2}\mathbf{e}_1} - F^1_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1}\right] + \frac{\Delta t}{\Delta y}\left[G^1_{\mathbf{i}-\frac{1}{2}\mathbf{e}_2} - G^1_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2}\right], \tag{40}$$

3.4.2. 2D CAT2P

The high order CAT2P iterative procedure are computed as follows:

1. Define
$$f^{(0)}_{\mathbf{i},\mathbf{j}} = f^n_{\mathbf{i}+\mathbf{j}}, \quad g^{(0)}_{\mathbf{i},\mathbf{j}} = g^n_{\mathbf{i}+\mathbf{j}}, \quad \mathbf{j} \in \mathcal{I}_P.$$
2. For $k = 2, \ldots, 2P$:
   (a) Compute
   $$u^{(k-1)}_{\mathbf{i},\mathbf{j}} = -A^{1,j_1}_P(f^{(k-2)}_{\mathbf{i},(*,j_2)}, \Delta x) - A^{1,j_2}_P(g^{(k-2)}_{\mathbf{i},(j_1,*)}, \Delta y), \quad \mathbf{j} \in \mathcal{I}_P.$$
   (b) Compute
   $$f^{k-1,n+r}_{\mathbf{i},\mathbf{j}} = f\left(u^n_{\mathbf{i}+\mathbf{j}} + \sum_{l=1}^{k-1} \frac{(r\Delta t)^l}{l!} u^{(l)}_{\mathbf{i},\mathbf{j}}\right), \quad \mathbf{j} \in \mathcal{I}_P, \ r = -P+1, \ldots, P.$$
   (c) Compute
   $$f^{(k-1)}_{\mathbf{i},\mathbf{j}} = A^{k-1,0}_P(f^{k-1,*}_{\mathbf{i},\mathbf{j}}, \Delta t), \quad \mathbf{j} \in \mathcal{I}_P.$$
3. Compute

$$F^P_{\mathbf{i}+\frac{1}{2}\mathbf{e}_1} = \sum_{k=1}^{2P} \frac{\Delta t^{k-1}}{k!} A^{0,1/2}_P(\tilde{f}^{(k-1)}_{\mathbf{i},(*,0)}, \Delta x), \tag{41}$$

$$G^p_{\mathbf{i}+\frac{1}{2}\mathbf{e}_2} = \sum_{k=1}^{2P} \frac{\Delta t^{k-1}}{k!} A^{0,1/2}_P(\tilde{g}^{(k-1)}_{\mathbf{i},(0,*)}, \Delta y). \tag{42}$$

The notation used for the approximation of the spatial partial derivatives is the following:

$$A^{k,q}_P(f_{\mathbf{i},(*,j_2)}, \Delta x) = \frac{1}{\Delta x^k} \sum_{l=-P+1}^{P} \gamma^{k,q}_{P,l} f_{\mathbf{i},(l,j_2)}$$
$$A^{k,q}_P(g_{\mathbf{i},(j_1,*)}, \Delta y) = \frac{1}{\Delta y^k} \sum_{l=-P+1}^{P} \gamma^{k,q}_{P,l} g_{\mathbf{i},(j_1,l)}$$

Remark 6. *In the last step of the algorithm above the set $\mathcal{I}_P$ can be replaced by its $(2P-1)$-point subset*

$$\mathcal{I}^0_P = \{\mathbf{j} = (j_1, j_2) \ \text{ such that } \ j_1 = 0 \ \text{ or } \ j_2 = 0\}$$

since only the corresponding values of $\tilde{f}^{(k-1)}_{\mathbf{i},\mathbf{j}}$ are used to compute the numerical fluxes (41) and (42).

3.5. Discussion

The ACAT numerical method presented in previous sections is exhaustively described in [4, 3] where numerical results are provided. The extension to systems of conservation laws is done component-by-component, and the extension to multi-dimensions relies on a direction-by-direction-type splitting. However we can point out several defects in the design of such an approach. First of all, the impact of the *a priori* smoothness indicators is tremendous. Indeed, they must effectively detect the presence of a discontinuity, but also the occurrence of new ones. While the former is achievable with *a priori* limiters, the latter is usually more difficult. Moreover the smoothness indicators can detect discontinuity only if the mesh is fine enough, otherwise it is hard to distinguish between a discontinuity and a smooth region with a large gradient. Secondly, the smoothness indicator cost drastically increases with P. Moreover, it becomes increasingly more difficult to determine the smoothness of a numerical solution when higher and higher order of accuracy is required. Thirdly, just like any *a priori* limiter, the current one has a natural tendency to over-estimate and over-react to possible spurious troubles. As such, the nominal order of accuracy on smooth solution is not always achieved, unless the grid is really fine. Furthermore, the *a priori* method focuses solely on the smoothness of the approximation and does not evaluate the physical aspects, such as positivity preservation or entropy creation.
These facts have restricted the effective use of ACAT schemes for orders not greater than $2P = 6$. Also the cost of the current version of ACAT2P constitutes a genuine limitation in 2D.
In this work we present an alternative way to couple CAT scheme with an *a posteriori* limiting technique, called MOOD, and remove some of the previously described defects.

4. CAT-MOOD

The main objective of this work is to combine the *a posteriori* shock capturing technique (MOOD) [8] to this family of one-step spatial and temporal reconstructions of high order CAT schemes [26].

The MOOD algorithm evaluates *a posteriori* the solution of the high-order numerical method using a class of criteria that detect a variety of oscillations, even very small ones. It is therefore possible to ensure that the numerical solution preserves some properties of the exact solutions of the PDE system, such as, for example, positivity, monotonicity and increase of the physical entropy, even in complex cases. In addition, this procedure allows to reduce and eliminate the numerical oscillations introduced by the high order methods in the presence of shocks or large gradients as is the case for the CAT methods.

The basic idea of the MOOD procedure is to apply a high-order method over the entire domain for a time step, then check locally, for each cell i, the behavior of the solution via admissibility criteria. If the solution computed in cell i at time t^{n+1} is in accordance with the criteria considered, it is kept, otherwise, it is recomputed with a new numerical method of order lower than the previous one. This operation is repeated until acceptability, or, when the (last) first order scheme is used. This last case occurs when the admissibility criteria fail with any previous reconstruction.

Therefore, the object of this work is to design a cascade of CAT methods in which the order is locally adapted according to *a posteriori* admissibility criteria thus creating a new family of adaptive CAT methods called CAT-MOOD schemes.

4.1. MOOD admissibility criteria

In this work we select 3 different admissibility criteria [9, 10] which are invoked onto the candidate numerical solution $\left\{u_i^{n+1}\right\}_{1\leq i\leq I}$:

1. *Computer Admissible Detector (CAD)*: This criterion is responsible to detect undefined or unrepresentable quantities, usually not-a-number `NaN` or infinity quantity due to some division by zero for instance.
2. *Physical Admissible Detector (PAD)*: The second detector is responsible for ensuring the physical validity of the candidate solution. The detector reacts to every negative pressure p or density ρ in the computational domain, in compliance with (5), since otherwise the solution will create non-physical sound speeds, imaginary time steps and so on. The physicality here is assessed from the point of view of a fluid flow, which limits the generality of the criteria as far as predicted pressures are concerned. Said differently, this physical admissibility criteria must be adapted to the model of PDEs which is solved.
3. *Numerical Admissible Detector (NAD)*: This criterion corresponds to a relaxed variant of the Discrete Maximum Principle (see [7, 8])

$$\min_{c\in\mathcal{C}_i^P}(w_c^n) - \delta_i \leq w_i^* \leq \max_{c\in\mathcal{C}_i^P}(w_c^n) + \delta_i,$$

where $\mathcal{C}_i^P = \{-P, \dots, P\}$ is the local centered stencil of order $2P$ and δ_i is a relaxation term to avoid problems in flat region. w_i^* is the numerical solution obtained with the scheme of order $2P$ and δ_i is set as:

$$\delta_i = \max\left(\varepsilon_1, \varepsilon_2 \left[\max_{c\in\mathcal{C}_i^P} w_c^n - \min_{c\in\mathcal{C}_i^P} w_c^n\right]\right). \tag{43}$$

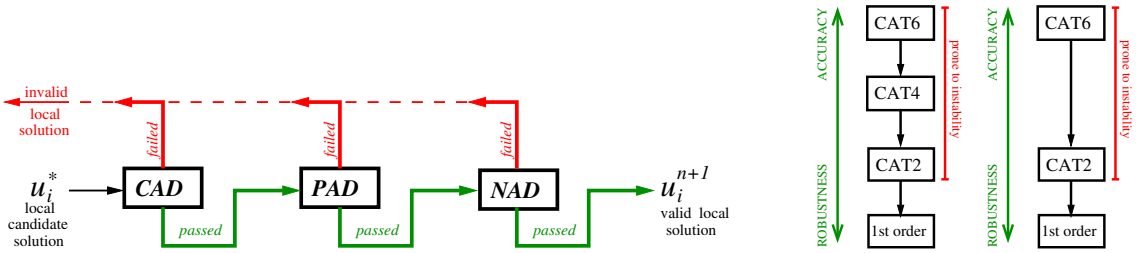
Here ε_1 and ε_2 are small dimensional constants. The values of ε_1 and ε_2 are reported in Sec. 5. The optimal choice of such constants depends on several factors, such as the order of magnitude of the field variables, grid spacing, and so on (see e.g. [2]). This criterion is responsible for guaranteeing the essentially non oscillatory (ENO) character of the solution; that is, no large and spurious minima or maxima are introduced locally in the solution[3].

If a `NaN` number is detected by CAD in the candidate solution w_i^*, then this cell is sent back for re-computation right away. Next, if the candidate solution has some positivity issues, then the PAD test is not passed, and, the cell is also invalid. At last the cell enters the NAD criteria to test for possible numerical oscillation. If spurious oscillations have contaminated the candidate solution, w_i^*, then it will fail NAD. The candidate solution is then recomputed if at least one of the previous criteria ordered into a chain, see Figure 2a, has failed. As a consequence, the MOOD loop drives the code to locally downgrade the order of accuracy by using an auxiliary scheme of lower accuracy.

4.2. CAT scheme with MOOD limiting

In this work we target to reach a maximal 6th order of accuracy on the part of the domain where a smooth solution is present. This can be achieved with CAT6 scheme that we would like to employ as often as possible. On the contrary, for cells presenting a discontinuous solution we plan to rely on a 1st order low accurate but robust scheme, for instance using Rusanov or HLL fluxes, that we would employ only when and where necessary. In between these two extremes, the CAT4 schemes of 4th order of accuracy is inserted and tried when CAT6 fails. If CAT4 scheme also fails, then CAT2 scheme (2nd order of accuracy) is further tried. As such we build several cascades of schemes of decreasing orders but possibly increasing robustness, see Figure 2b. Notice

[3] In the numerical tests of Sec. 5 we compute the relaxed discrete maximum principle only for density ρ and pressure p. No limiting, on the velocity components, u, v, has been considered.



(a) Detection criteria ordered list. (b) Detection accuracy order.

Figure 2: Sinopsis of CATMOOD. (a) Detection criteria of the MOOD procedure for a candidate solution u_i^*. Criteria: *Computer Admissible Detector (CAD)*, *Physical Admissible Detector (PAD)* and *Numerical Admissible Detector (NAD)*. (b) Cascades of CAT schemes used in the MOOD procedure. Starting from the most accurate one, CAT6, when the solution does not appear to be smooth, the scheme selects lower order methods, down to a robust 1st order one: two possible strategies are shown. The cascade we found to be more effective is the one on the far right.

that, apart from CAT6 and the 1st order parachute scheme, the user can decide if the intermediate schemes (CAT4 and CAT2 here) should be included or not. In the numerical results in section 5 we skip CAT4 and follow the second cascade illustrated in Figure 2b in order to spare computing resources. This scheme is referred to as CATMOOD6.

4.3. CATMOOD algorithm

Practically, for the time-step $[t^n, t^{n+1}]$ and for each cell i, we define a 'mask', $M_i^n \in \{-1, 0, 1\}$, such that

$$M_i^n = \begin{cases} 1 & \text{if } u_i^* \text{ fails at least one criterion,} \\ -1 & \text{if } \exists j \in \mathcal{N}_i \text{ s.t } M_j^n = 1, \\ 0 & \text{otherwise.} \end{cases} \tag{44}$$

where $\mathcal{N}_i$ is the set of direct neighbor cells of cell ω_i. $M_i^n = -1$ means that cell i is the neighbor of an invalid cell.
The algorithm designed for CATMOOD6 scheme is:

Init Let $\{u_i^n\}_{1 \leq c \leq N_c}$ be the numerical solution at $t = t^n$ over the whole domain Ω.

CAT6 Let $\{u_i^*\}_{1 \leq c \leq N_c}$ be the numerical solution at $t = t^{n+1}$ of order 6 obtained from CAT6 scheme. Set $M_i^n = 0$.
For all cell i, check if u_i^* satisfies all detection criteria. In this case, then $u_i^{n+1} = u_i^*$ and $M_i^n = 0$. Otherwise $M_i^n = 1$ and, if $M_j^n = 0$ then set $M_j^n = -1$ for all $j \in \mathcal{N}_i$.

CAT4 Only for the troubled cell i, i.e $M_i^n \neq 0$, recompute $\{u_i^*\}$ the numerical solution at time $t = t^{n+1}$ of order 4 obtained with CAT4 scheme.
Check if u_i^* satisfies all detection criteria. In this case, then $u_i^{n+1} = u_i^*$ and $M_i^n = 0$. Otherwise $M_i^n = 1$ and, if $M_j^n = 0$ then set $M_j^n = -1$ for all $j \in \mathcal{N}_i$.

CAT2 Only for the troubled cell i, i.e $M_i^n \neq 0$, recompute $\{u_i^*\}$ the numerical solution at time $t = t^{n+1}$ of order 2 obtained with CAT2 scheme.
Check if u_i^* satisfies all detection criteria. In this case, then $u_i^{n+1} = u_i^*$ and $M_i^n = 0$. Otherwise $M_i^n = 1$ and, if $M_j^n = 0$ then set $M_j^n = -1$ for all $j \in \mathcal{N}_i$.

1st-ord Only for the remaining troubled cell i, i.e $M_i^n \neq 0$, recompute $\{u_i^*\}$ the numerical solution at time $t = t^{n+1}$ of order 1 obtained with a first order scheme, and set $u_i^{n+1} = u_i^*$.

For efficiency purposes we may remove the CAT4 step in the cascade.

4.4. Complexity, cost, convergence, implementation

The unavoidable extra-cost when using a MOOD procedure is the detection of troubled cells with the admissible criteria from section 4.1. However the detection criteria are particularly inexpensive to compute compared to smoothness indicators or sensors used for instance in ACAT scheme. Usually these admissibility criteria have a negligible cost.
The MOOD limiting procedure always converges because the number of cells and schemes are finite. In the worst case scenario the entire solution is computed successively with all CAT2P schemes up to the 1st order accurate solution. In the best case scenario the solution from the unlimited CAT6 scheme is accepted without any correction. Any situation in-between is possible, and generally only few cells need to be recomputed.
The detection criteria are fundamental, they must be designed to ensure that, if the mesh is fine enough, a smooth solution computed by CAT6 scheme does not produce any troubled cell. They must also ensure that, in the vicinity of strong discontinuity, the robust 1st order scheme is regularly employed to avoid spurious oscillations.
Concerning the complexity of CATMOOD, it becomes impossible to estimate *a priori* the cost because the limiting adapts to the underlying flow and the computed numerical solution. In the best case scenario mentioned above, the cost of CATMOOD6 is the one of CAT6, in the worst, the cost of CATMOOD6 is the sum of the cost of all schemes in the cascade. Generally one observes that the amount of troubled cells is of the order of $0-20\%$ of the total number of cells, which makes the MOOD procedure genuinely competitive compared to existing *a priori* limiters, see [9, 25] and the numerical section in this paper.

4.5. CATMOOD vs LATMOOD

In the context of the Lax-Wendroff Approximated Taylor (LAT) methods proposed by D. Zorío et al. [42], it is noteworthy that the number of operations per time step is significantly reduced compared to CAT2P methods. This reduction is achieved by performing the Taylor expansions only once per point, while CAT2P requires $2P$ expansions per point. However, it is important to acknowledge that CAT2P exhibits superior stability properties compared to LAT.

Consequently, an intriguing question arises: can LATMOOD yield a more efficient approach than CATMOOD? To explore this question, it is important to consider two crucial points:

1. The local computation of numerical fluxes makes CAT more suitable for implementing MOOD. Indeed, unlike CAT schemes, LAT methods compute temporal derivatives in a global manner, which contradicts the local approach of MOOD. Specifically, in CAT methods, once the solution u^* has been computed, if any of the detectors fail, it is sufficient to recompute all local time derivatives of the fluxes with a lower order of accuracy, without causing any issues in the MOOD approach. Contrarily in the case of LAT, since all the approximations are computed in a non local sense (see Remark 2) and since LAT does not use a compact stencil, $4P$ computations have to be updated for each bad cell and approximation $u^{(1)}_{i+p}$, $u^{1,n+1}_{i+p}$, etc., making the scheme computationally inefficient.
2. From our preliminaries tests we notice that the proportion of low order cells substantially increases when LAT is employed instead of CAT.

5. Numerical test cases

In this paper only numerical tests related to the two-dimensional Euler equations are considered. Obviously, CATMOOD schemes could be applied to any systems of conservation laws without any restriction. In this section, unless otherwise stated, we consider the international system of units (SI) and therefore omit the units in the following. Our methodology of testing relies on several classical and demanding test cases:

1. *Isentropic vortex in motion.* This test measures the ability of the MOOD procedure combined with CAT schemes to achieve the optimal high order for a smooth solution. We also compared CATMOOD6 with unlimited CAT schemes, limited ACAT ones and some first order scheme using Rusanov, HLL and HLLC fluxes.
2. *Sedov Blast wave.* This problem has an exact solution presenting a single cylindrical shock wave followed by an exponential decay [32]. This test is used to check the behaviour of the CATMOOD6 scheme against shocks and to compare CATMOOD6 versus ACAT6.
3. *2D Riemann problems.* We simulate four versions of the four-state 2D Riemann problems [30]. These problems present large smooth regions, unstable shear layers, unaligned contact discontinuities and shock waves, along with complex interaction patches for which no exact solution has yet been derived.
4. *Astrophysical jet.* This intense and demanding test case challenges the positivity and robustness of the CATMOOD6 scheme as it presents an extremely violent jet at Mach 2000 generating a bow shock and unstable shear layers.

The time step is chosen according to

$$\Delta t^n = \text{CFL} \min\left(\frac{\Delta x}{\lambda^n_{\max_x}}, \frac{\Delta y}{\lambda^n_{\max_y}}\right),$$

where $\lambda^n_{\max_x}$ and $\lambda^n_{\max_y}$ represent the maximum, over cells, of the spectral radius of the Jacobian matrices $\partial\mathbf{F}/\partial u$ and $\partial\mathbf{G}/\partial u$ [1]. Empirically we found that the CFL number should be less or equal to 0.5. In our calculations we choose CFL= 0.4. In all the numerical examples below (except the one in Section 5.4), the units are chosen in such a way that the order of magnitude of all field quantities is one, therefore we always use the same value for the constants appearing in (43), i.e. $\varepsilon_1 = 10^{-4}$ and $\varepsilon_2 = 10^{-3}$, while the value $\varepsilon = 10^{-8}$ has been used in (29). With the choice of the (NAD) criteria depending on the chosen ε_1 and ε_2 (43), the mesh needs to be fine enough, otherwise the numerical admissible criteria might trigger too much smoothing of the solution. The user of this method might have not only to choose "a fine enough mesh" but possibly slightly adapt the parameters composing (43). This issue had been analyzed in [2], where it was shown how the accuracy depends on the choice of ε_1 and ε_2, as well as on the mesh refinement and an enhanced (NAD) criteria was proposed to overcome the issue.

5.1. Isentropic vortex in motion

The isentropic vortex problem [34] challenges the accuracy of numerical methods since an exact, smooth and analytic solution exists. The same units than in [34] are adopted. The computational domain is set to $\Omega = [-10, 10] \times [-10, 10]$. The ambient flow is characterized by $\rho_\infty = 1.0$, $u_\infty = 1.0$, $v_\infty = 1.0$ and $p_\infty = 1.0$, with a normalized ambient temperature $T^*_\infty = 1.0$. At the initial time, $t = 0$, onto this ambient flow is superimposed a vortex centered at $(0, 0)$ with the following state: $u = u_\infty + \delta u$, $v = v_\infty + \delta v$, $T^* = T^*_\infty + \delta T^*$, where the increments are given by

$$\delta u = -y' \frac{\beta}{2\pi} \exp\left(\frac{1-r^2}{2}\right), \quad \delta v = x' \frac{\beta}{2\pi} \exp\left(\frac{1-r^2}{2}\right), \quad \delta T = -\frac{(\Gamma-1)\beta^2}{8\Gamma\pi^2} \exp\left(1-r^2\right),$$

with $r = \sqrt{x^2+y^2}$. The so-called strength of the vortex is set to $\beta = 5.0$ and the initial density is given by $\rho = \rho_\infty (T/T_\infty)^{\frac{1}{\Gamma-1}}$. Periodic boundary conditions are prescribed. At final time $t = t_{\text{final}} = 20$ the vortex is back to its original position, and, the final exact solution matches the initial one. Since the solution is smooth, it should be simulated with optimal high accuracy, in other words, the limiting/stabilization procedure employed in the scheme should not have any effect.

2D Isentropic Vortex in motion - Rate of convergence								
	Rusanov-flux		**HLL**		**HLLC**		**CATMOOD6**	
N	L^1 error	order	L^1 error	order	L^1 error	order	L^1 error	order
50 × 50	8.44×10^{-3}	—	8.44×10^{-3}	—	7.91×10^{-3}	-	8.48×10^{-3}	—
100 × 100	8.04×10^{-3}	0.07	8.04×10^{-3}	0.07	6.86×10^{-3}	0.21	3.77×10^{-3}	1.17
200 × 200	6.68×10^{-3}	0.27	6.67×10^{-3}	0.27	5.31×10^{-3}	0.37	2.40×10^{-7}	13.94
300 × 300	5.71×10^{-3}	0.36	5.71×10^{-3}	0.36	4.53×10^{-3}	0.39	2.06×10^{-8}	6.05
400 × 400	4.98×10^{-3}	0.47	4.98×10^{-3}	0.47	3.86×10^{-3}	0.55	3.52×10^{-9}	6.14
	Expected	1	Expected	1	Expected	1	Expected	6
	CAT2		**CAT4**		**CAT6**		**ACAT6**	
N	L^1 error	order	L^1 error	order	L^1 error	order	L^1 error	order
50 × 50	7.94×10^{-3}	—	2.03×10^{-3}	—	8.46×10^{-4}	-	8.95×10^{-3}	—
100 × 100	2.55×10^{-3}	1.64	1.42×10^{-4}	3.83	1.56×10^{-5}	5.76	8.28×10^{-3}	0.11
200 × 200	6.12×10^{-4}	2.06	8.34×10^{-6}	4.09	2.41×10^{-7}	6.02	8.34×10^{-5}	9.95
300 × 300	2.69×10^{-4}	2.02	1.64×10^{-6}	4.02	2.09×10^{-8}	6.03	1.05×10^{-5}	5.14
400 × 400	1.52×10^{-4}	1.99	5.16×10^{-7}	4.01	3.68×10^{-9}	6.03	2.48×10^{-6}	4.93
	Expected	2	Expected	4	Expected	6	Expected	6

Table 1: Isentropic vortex in motion L^1-norm errors on density ρ between the numerical solution and the exact solution of the isentropic vortex in motion problem at $t_{\text{final}} = 20$ on uniform Cartesian mesh.

	1st order methods			2nd ord.	4th ord.	6th order methods		
	Rusanov	**HLL**	**HLLC**	**CAT2**	**CAT4**	**CAT6**	**CATMOOD6**	**ACAT6**
CPU(s)	48.22	60.58	98.41	424.72	4375.68	16112.38	20848.26	23597.67
Ratio	1	1.26	2.04	8.81	90.74	334.14	432.36	489.38

Table 2: Isentropic vortex in motion with 300 × 300 cells. First row: computational costs expressed in seconds. Second row: ratio with respect to the Rusanov cost.

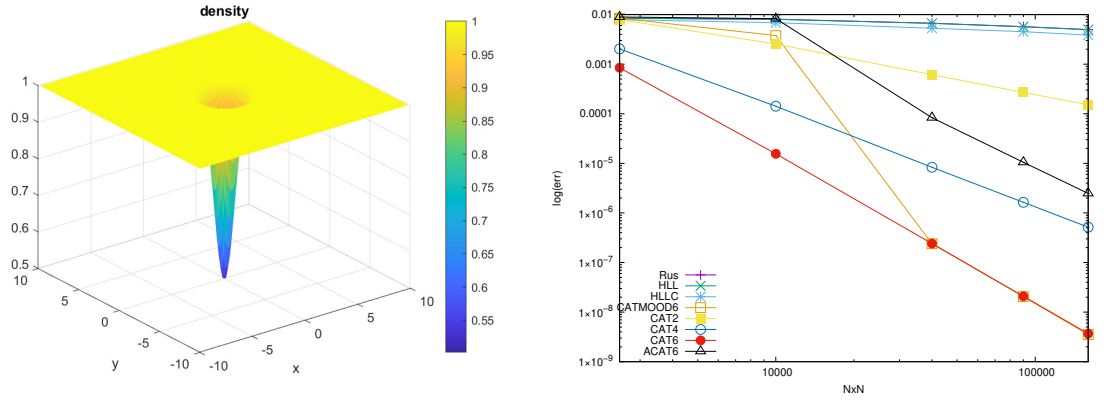
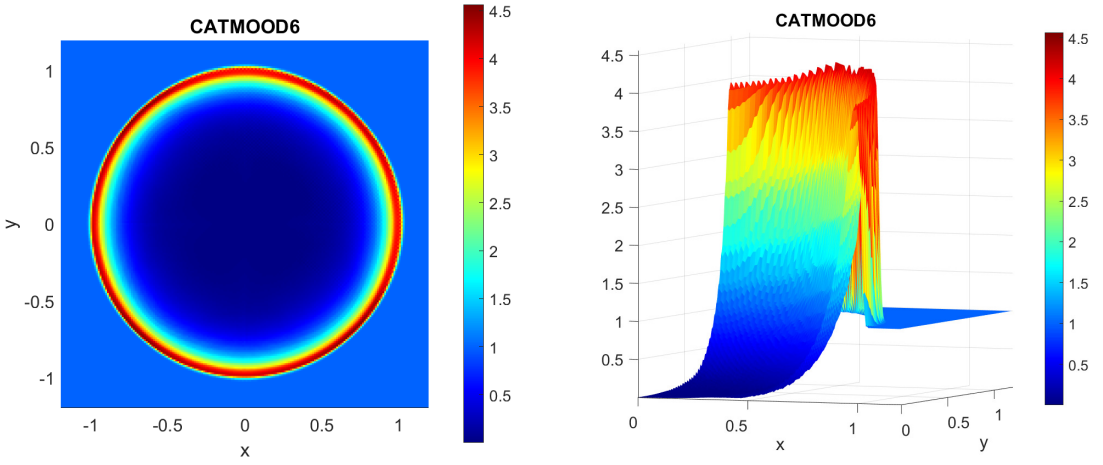


Figure 3: 2D isentropic vortex in motion — Left: Numerical solution at final time for density with CATMOOD6 with HLLC flux scheme used for the first order method on 100 × 100 uniform mesh and CFL= 0.4 — Right: Errors in L_1 norms given by all schemes.

We run the isentropic vortex test case with first order Rusanov-flux, HLL, HLLC and sixth order CATMOOD method on successive refined Cartesian meshes going from 50 × 50 up to 400 × 400 cells. The density at final time is plotted in Figure 3-left. The results of the convergence analysis

for all the schemes are displayed in table 1 and in Figure 3-right. The errors reported in the table are L^1 relative errors computed as $\text{err}_N = \|u_N - u_N^{\text{exa}}\|_1 / \|u_N^{\text{num}}\|_1$, where u_N^{num} is the numerical solution on the $N \times N$ grid, and u_N^{exa} is the exact solution computed on the same grid. The order of convergence is the log of the error ratio $\text{err}_N/\text{err}_{2N}$ divided by $\log(2)$. The expected rate of convergence is reached for CAT schemes, while for the first-order schemes the convergence is below the expected order. Ideally the limited 6th-order ACAT6 scheme should produce the same errors as the ones given by CAT6 scheme, but ACAT6 errors are two orders of magnitude greater. Contrarily, starting at mesh 200×200, CATMOOD6 errors match the optimal ones from CAT6, and, the nominal 6th order is retrieved. This proves that the *a posteriori* MOOD limiting avoids spurious intervention if the mesh is fine enough. Table 2 presents the computational costs of various methods for the isentropic vortex test case on a uniform grid with 300×300 cells, CFL$= 0.4$, final time $t_{\text{final}} = 20$, and periodic boundary conditions. The first row displays the CPU computational costs in seconds for the eight methods, while the second row shows the ratio between the computational cost of each scheme and the one of the Rusanov method. This table emphasizes the superiority of the *a posteriori* approach over the *a priori* one. Indeed ACAT6 scheme appears to be not only less accurate than CATMOOD6, but also its execution time is higher[4].

5.2. Sedov blast wave



(a) Projection of the numerical density for CATMOOD6 with HLLC as first order scheme

(b) Zoom on $[0, 1.2] \times [0, 1.2]$ of the numerical density for CATMOOD6.

Figure 4: Sedov blast wave 5.2. Numerical density obtained with CATMOOD6 with HLLC as first order scheme on the interval $[-1.2, 1.2] \times [-1.2, 1.2]$ adopting a 200×200 mesh and CFL$= 0.4$ at time $t = 1$. Projection on the plane O-xy (left); zoom on the interval $[0, 1.2] \times [0, 1.2]$ (right).

The 2D Sedov problem [32] is a cylindrical symmetric explosion. The domain is given by $(x, y) \in [-1.2, 1.2]^2$ initially filled with perfect gas at rest such as $(\rho^0, u^0, v^0, p^0, \Gamma) = (1, 0, 0, 10^{-13}, 1.4)$.

[4] The schemes have been implemented in a vectorized manner in MATLAB. However, the local nature of MOOD cannot be implemented in MATLAB in a vectorized sense. Therefore, the slight difference in the performance between the two strategies can be attributed to this phenomenon.

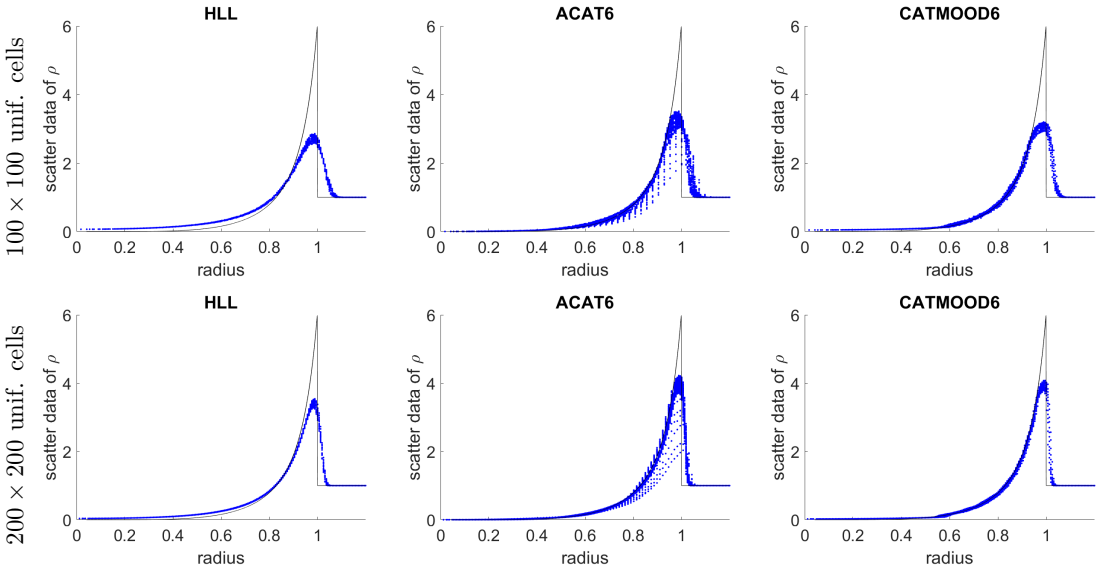


Figure 5: Sedov blast wave from section 5.2 — Scatter plots of the numerical density obtained with a 1st order scheme HLL, and 6th order ACAT6 and CATMOOD6 with HLL flux on the interval $[-1.2, 1.2] \times [-1.2, 1.2]$. Top: results for 100×100 uniform cells. Bottom: results for 200×200 uniform cells. In black is the exact solution. x-label represents the radius of the scatter crows; while y-label represents the scatter data of the numerical density.

A total energy of $E_{total} = 0.244816$ is concentrated at the origin [27]. This configuration corresponds to a point-wise symmetric explosion, for which a cylindrical shock front reaches radius $r = \sqrt{x^2 + y^2} = 1$ at $t_{\text{final}} = 1$ with a density peak of $\rho = 6$, see Figure 4b for an example of a quasi symmetric numerical solution.
Figure 5 presents the scatter plots of numerical density as a function of cell radius using, respectively for the first and second columns either 100×100 or 200×200 uniform cells. The tested schemes are the 1st order, ACAT6 and CATMOOD6 using HLL numerical flux. Notice that the unlimited CAT schemes fails in producing a numerical solution due to the presence of a strong shock wave. Because the exact solution has a cylindrical symmetry (see the black line in Figure 5), all cells at the same radius should share the same numerical density, if the scheme exactly preserves the cylindrical symmetry. The width of the spread/variance of numerical data somehow measures how good the scheme can preserve this symmetry. As is visible in the figures, the refinement of the grid implies a reduction of the variance for any of the methods. Moreover, the 6th order schemes (ACAT6 and CATMOOD6) seem to produce a sharper shock wave and more accurate results. Importantly CATMOOD6 has a better cylindrical symmetry than ACAT6. Of course no positivity issue is reported for any of these schemes.

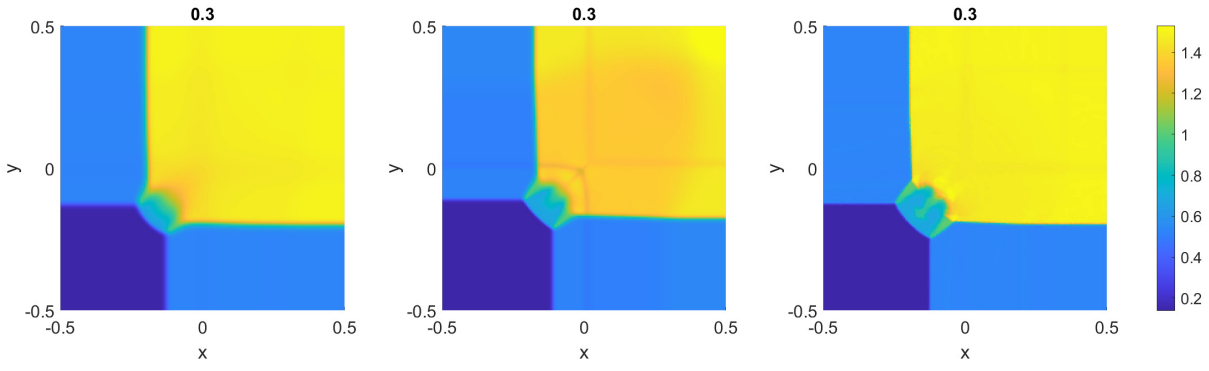
5.3. 2D Riemann problems

2D Riemann problems: four 2D Riemann problems are considered that correspond to the configurations 3, 6, 11, and 17 in [30]. The units from [30] are adopted. These problems present large smooth regions, unstable shear layers, unaligned contact discontinuities and shock waves, along with complex interaction patches for which no exact solution has yet been derived. The computational domain is $[-1, 1] \times [-1, 1]$ where zero Neumann conditions are applied, and the four initial conditions are set respectively as in Table 3. A 400×400 mesh is adopted for all schemes and configurations. The final time is set to $t_{\text{final}} = 0.3$ and the CFL to 0.4. We run four simulations using CATMOOD6 and the three first order schemes respectively with Rusanov, HLL

and HLLC flux functions. The 1st order scheme with Rusanov flux has been adopted as parachute scheme in the CATMOOD6 cascade. In these 2D figures the schemes are respectively ordered from top-left to bottom-right. In addition, in a different set of figures for CATMOOD6 schemes, we plot the percentage of cells updated with CAT6 (top), CAT2 (center) or 1st order Rusanov (bottom) schemes as a function of time-step.

Configuration 3			
$\rho_2 = 0.5323$	$u_2 = 1.206$	$\rho_1 = 1.5$	$u_1 = 0$
$v_2 = 0$	$p_2 = 0.3$	$v_1 = 0$	$p_1 = 1.5$
$\rho_3 = 0.138$	$u_2 = 1.206$	$\rho_4 = 0.5323$	$u_4 = 0$
$v_3 = 1.206$	$p_2 = 0.029$	$v_4 = 1.206$	$p_4 = 0.3$
Configuration 6			
$\rho_2 = 2$	$u_2 = 0.75$	$\rho_1 = 1.5$	$u_1 = 0.75$
$v_2 = 0.5$	$p_2 = 1$	$v_1 = -0.5$	$p_1 = 1$
$\rho_3 = 1$	$u_2 = -0.75$	$\rho_4 = 3$	$u_4 = -0.75$
$v_3 = 0.5$	$p_2 = 1$	$v_4 = -0.5$	$p_4 = 1$
Configuration 11			
$\rho_2 = 0.5313$	$u_2 = 0.8276$	$\rho_1 = 1$	$u_1 = 0.1$
$v_2 = 0$	$p_2 = 0.4$	$v_1 = 0$	$p_1 = 1$
$\rho_3 = 0.8$	$u_2 = 0.1$	$\rho_4 = 0.5313$	$u_4 = 0.1$
$v_3 = 0$	$p_2 = 0.4$	$v_4 = 0$	$p_4 = 0.4$
Configuration 17			
$\rho_2 = 2$	$u_2 = 0.$	$\rho_1 = 1$	$u_1 = 0$
$v_2 = -0.3$	$p_2 = 1$	$v_1 = -0.4$	$p_1 = 1$
$\rho_3 = 1.0625$	$u_2 = 0$	$\rho_4 = 0.5197$	$u_4 = 0$
$v_3 = 0.2145$	$p_2 = 0.4$	$v_4 = -1.1259$	$p_4 = 0.4$

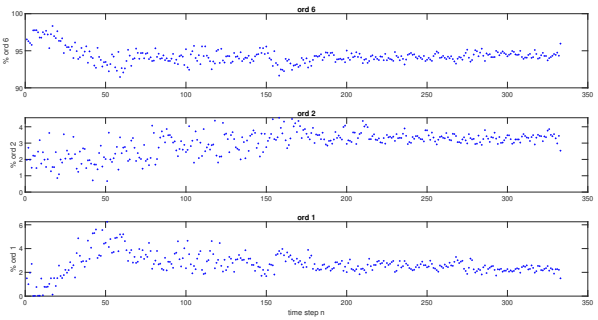
Table 3: 2D Riemann problem initial conditions.



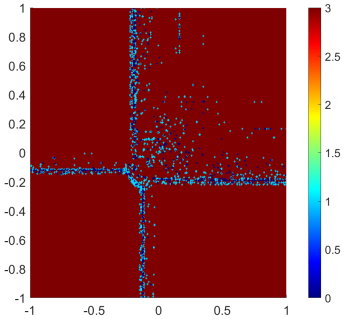
(a) Rusanov flux (b) HLLC (c) CATMOOD6

Figure 6: 2D Riemann problem from section 5.3 with initial condition configuration 3. Zoom of the numerical solution for density on the interval $[-1, 1] \times [-1, 1]$ adopting a mesh of $400 \times 400-$cells $t_{\text{final}} = 0.3$ and CFL$= 0.4$. Rusanov-flux (a); HLLC (b); and CATMOOD6 with Rusanov flux for the first order method (c).

Configuration 3. Figure 6 shows a zoom of the numerical densities for the configuration 3 (Table 3). All the schemes capture the same global solution. The first order methods are diffusive even if



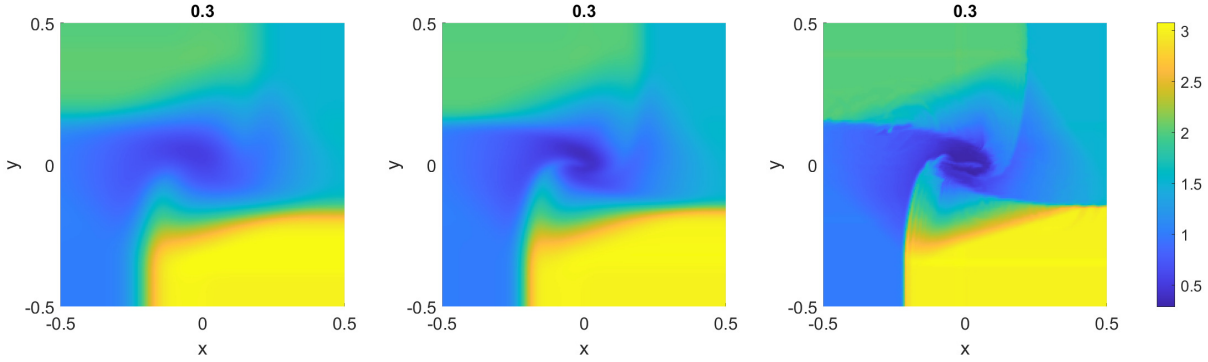
(a) Percentage of cells updated by CAT6 (top), by CAT2 (center), and by Rusanov (bottom).



(b) Local order of the density with CATMOOD6 at final time $t_{\text{final}} = 0.3$.

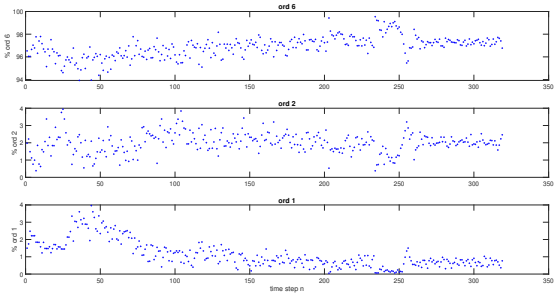
Figure 7: 2D Riemann problem from section 5.3 with initial condition configuration 3. Percentage of cells updated by CAT6, by CAT2, and by Rusanov (left); local orders used by CATMOOD6 (right). The numerical solution has been computed on the interval $[-1, 1] \times [-1, 1]$ adopting a mesh of 400×400 cells, $t_{\text{final}} = 0.3$ and CFL$= 0.4$. The cells are colored by the value of P from the CAT2P scheme used to update it, where $P \in \{0, 1, 3\}$ and $P = 0$ represents the first order Rusanov flux.

HLLC scheme performs better. Contrarily, the high order CATMOOD6 clearly improves the sharpness of the shear layers and contacts. Figure 7a exhibits the percentage of cells using CAT6 (top), CAT2 (center) and Rusanov scheme (bottom). We can observe that on average 95% of the cells are updated with 6th order of accuracy, about $3 - 4\%$ with 2nd order or 1st order. One observes that only few cells demand some limiting (see Figure 7b), and are sent back to t^n for re-computation by the MOOD approach. Consequently, the cost of CATMOOD6 is mainly the one of CAT6. The cells in Figure 7b is colored by the value of P from the CAT2P scheme used, where $P = 0$ (dark blue) represents the first order Rusanov flux, $P = 1$ (light blue) CAT2 scheme and $P = 3$ (dark red) for CAT6. Since CATMOOD6 does not include CAT4, cells updated with $P = 2$ (yellow color) do not exist. We observe that the vast majority of cells updated with a low order schemes are mainly located along the main shock waves.

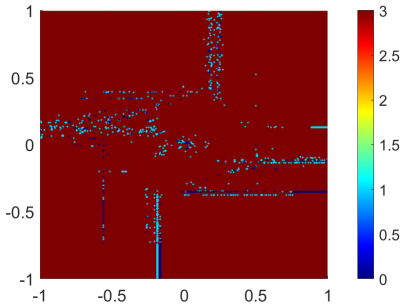


(a) Rusanov flux (b) HLLC (c) CATMOOD6

Figure 8: 2D Riemann problem from section 5.3 with initial condition configuration 6. Zoom of the numerical solution for density on the interval $[-1, 1] \times [-1, 1]$ adopting a mesh of $400 \times 400-$cells at time $t_{\text{final}} = 0.3$ and CFL$= 0.4$. Rusanov-flux (a); HLLC (b); and CATMOOD6 with HLLC for the first order method (c).

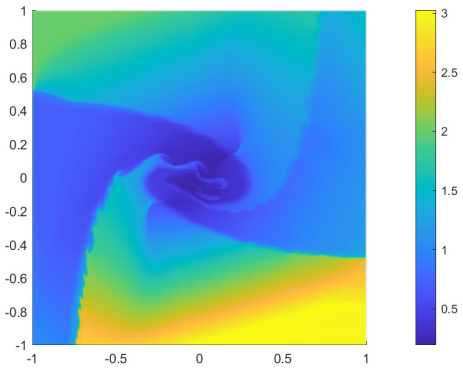


(a) Percentage of cells updated by CAT6 (top), by CAT2 (center), and by HLLC (bottom).

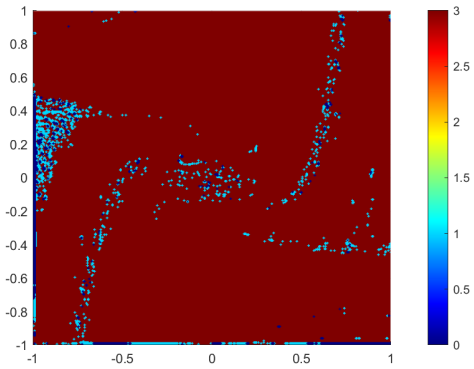


(b) Local order of the density with CATMOOD6 at final time $t_{\text{final}} = 0.3$.

Figure 9: 2D Riemann problem from section 5.3 with initial condition configuration 6. Percentage of cells updated by CAT6, by CAT2, and by Rusanov (left); local orders used by CATMOOD6 (right). The numerical solution has been computed on the interval $[-1, 1] \times [-1, 1]$ adopting a mesh of 400×400 cells, $t_{\text{final}} = 0.3$ and CFL$= 0.4$. The cells are colored by the value of P from the CAT2P scheme used to update it, where $P \in \{0, 1, 3\}$ and $P = 0$ represents the first order HLLC scheme.



(a) Numerical solution for density obtained with CATMOOD6 at final time $t_{\text{final}} = 1$.

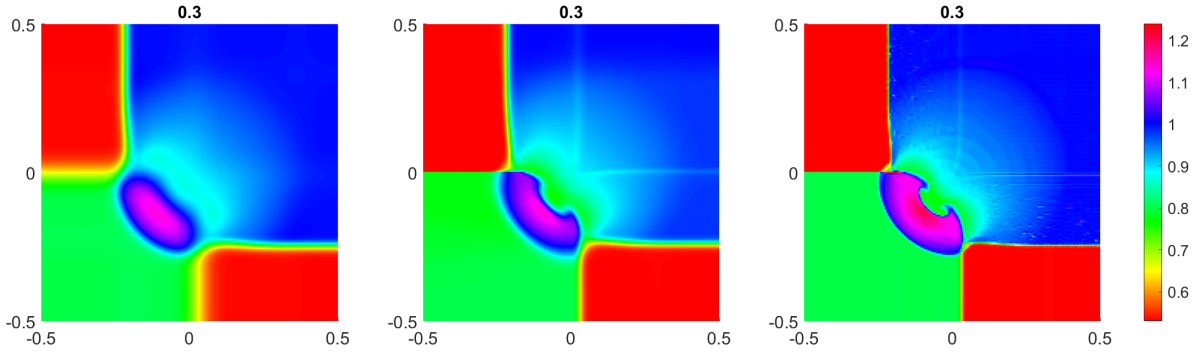


(b) Local order of the density with CATMOOD6 at final time $t_{\text{final}} = 1$.

Figure 10: 2D Riemann problem from section 5.3 with initial condition configuration 6. The numerical solution has been computed on the interval $[-1, 1] \times [-1, 1]$ adopting a mesh of 400×400 cells, $t_{\text{final}} = 1$ and CFL$= 0.4$. The numerical density is shown (left). The cells are colored by the value of P from the CAT2P scheme used to update it, where $P \in \{0, 1, 3\}$ and $P = 0$ represents the first order HLLC scheme (right).

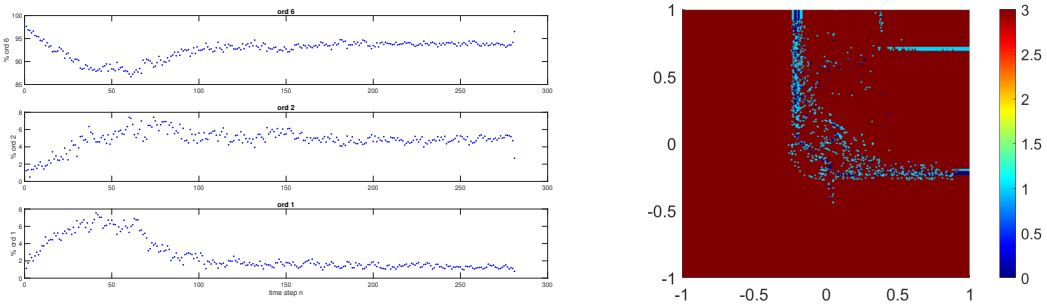
Configuration 6. Likewise for configuration 3, we plot the results in Figure 8 and 9. We again observe that the CATMOOD6 results are far more sharper than the 1st order schemes, with HLLC being the less dissipative scheme of the 1st order ones. Notice that the shear layers along the curves are Kelvin-Helmholtz unstable, so that it seems normal to see the occurrences of small vortices. This is a numerical evidence of low dissipation of CATMOOD6. The percentage of troubled cells is of the order of $3 - 4\%$ on averaged, again showing that CATMOOD6 has globally the same cost than CAT6 plus 10% of the cost of a first order scheme, see Figure 9a. Similarly to configuration 3, the cells in Figure 9b are colored by the value of P from the CAT2P scheme used, where $P = 0$ (dark blue) represents the first order HLLC scheme, $P = 1$ (light blue) CAT2 scheme and $P = 3$ (dark red) for CAT6. Since CATMOOD6 does not include CAT4, cells updated with $P = 2$ (yellow color) do not exist. We observe that the vast majority of cells updated with a low order schemes

are mainly located along the main shock waves. Then, the long time behavior of CATMOOD6 is displayed in Figure 10 for $t_{\text{final}} = 1$. The left panel presents the colored density, while the right one shows the local orders. One observes that the Kelvin-Helmholtz instabilities develop further as it is expected for a low dissipative scheme such as CATMOOD6.



(a) Rusanov flux (b) HLLC (c) CATMOOD6

Figure 11: 2D Riemann problem from section 5.3 with initial condition configuration 11. Zoom of the numerical solution for density on the interval $[-1, 1] \times [-1, 1]$ adopting a mesh of $400 \times 400-$cells $t_{\text{final}} = 0.3$ and CFL$= 0.4$. Rusanov-flux (a); HLLC (b); and CATMOOD6 with HLLC for the first order method (c).

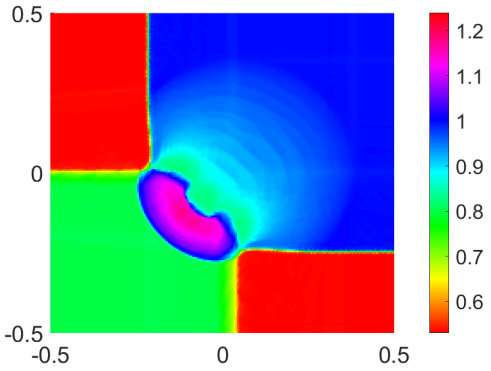


(a) Percentage of cells updated by CAT6 (top), by CAT2 (center), and by HLLC (bottom).

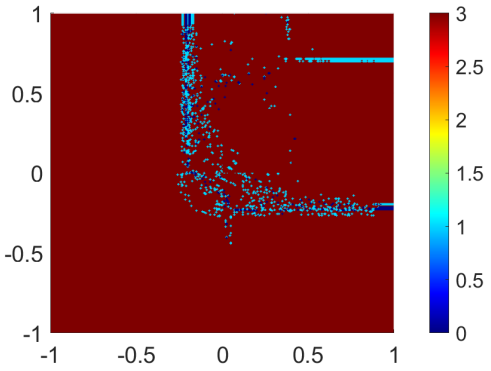
(b) Local order of the density with CATMOOD6 at final time $t_{\text{final}} = 0.3$.

Figure 12: 2D Riemann problem from section 5.3 with initial condition configuration 11. Percentage of cells updated by CAT6, by CAT2, and by Rusanov (left); local orders used by CATMOOD6 (right). The numerical solution has been computed on the interval $[-1, 1] \times [-1, 1]$ adopting a mesh of $400 \times 400-$cells, $t_{\text{final}} = 0.3$ and CFL$= 0.4$. The cells are colored by the value of P from the CAT2P scheme used to update it, where $P \in \{0, 1, 3\}$ and $P = 0$ represents the first order HLLC scheme.

Configuration 11. The numerical solutions for configuration 11 are gathered in Figures 11-13a and the percentages of troubled cells in 12a for CATMOOD6 with HLLC as first order scheme. Similarly results have been observed using the Rusanov fluxas parachute. It is important to notice that the wave are visibly sharper with the high-order CATMOOD6 as expected. The complex internal pattern is also better captured, avoiding the excessive diffusion that can be seen when any 1st order scheme is employed. Even if the internal structure is not exactly known, the results in panel (c) better match accurate results from the literature [30, 24]. We can observe spurious features in CATMOOD6 results (lighter spots), see Figure 11c but they could be dissipated by



(a) CATMOOD6 with Rusanov flux as first order scheme at final time $t_{\text{final}} = 0.3$.



(b) Local order of the density with CATMOOD6 at final time $t_{\text{final}} = 0.3$.

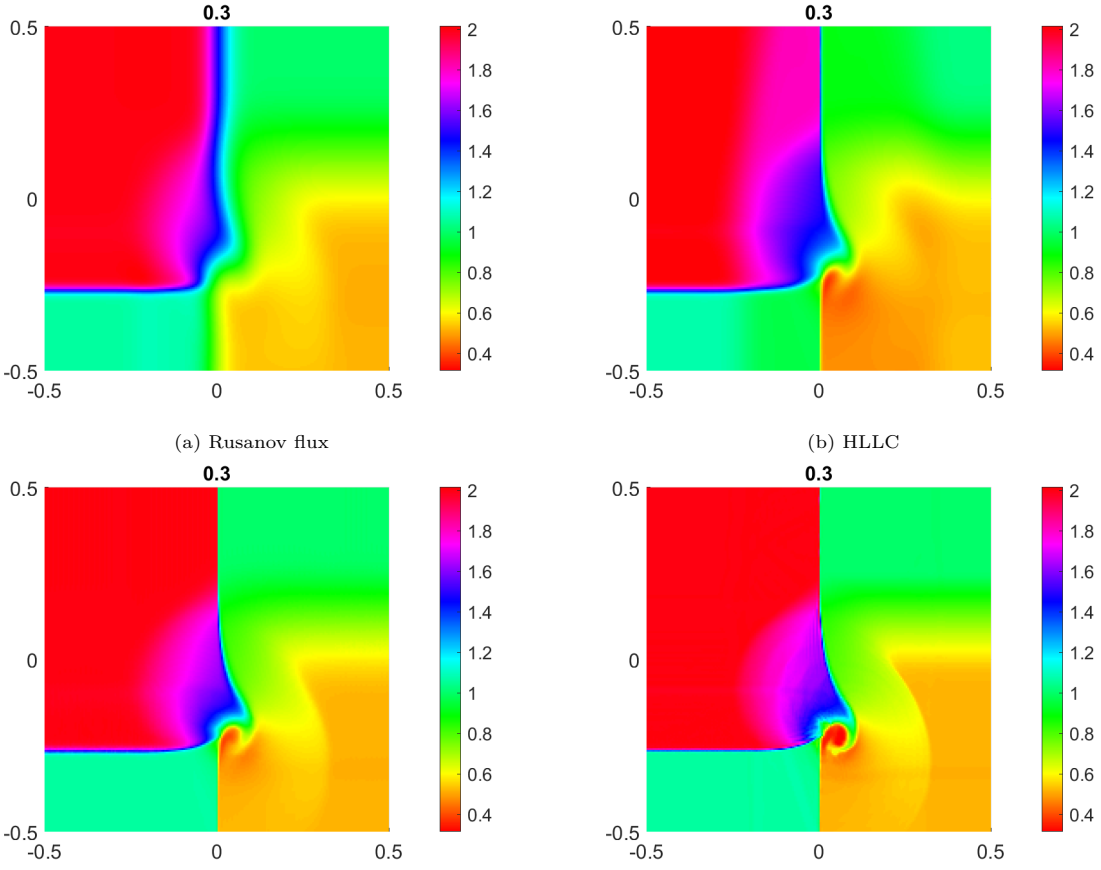
Figure 13: 2D Riemann problem from section 5.3 with initial condition configuration 11. Zoom of the numerical solution for density on the interval $[-1, 1] \times [-1, 1]$ adopting a mesh of $400 \times 400-$cells $t_{\text{final}} = 0.3$ and CFL$= 0.4$ obrained with CATMOOD6 where Rusanov flux has been used as parachute (left); local orders used by CATMOOD6 (right). The numerical solution has been computed on the interval $[-1, 1]\times[-1, 1]$ adopting a mesh of $400\times400-$cells, $t_{\text{final}} = 0.3$ and CFL$= 0.4$. The cells are colored by the value of P from the CAT2P scheme used to update it, where $P \in \{0, 1, 3\}$ and $P = 0$ represents the first order Rusanov flux.

using a more robust first order scheme, see Figure 13a. The cells in Figure 12b-13b are colored by the value of P from the CAT2P scheme used, where $P = 0$ (dark blue) represents the first order HLLC scheme or Rusanov flux respectively, $P = 1$ (light blue) CAT2 scheme and $P = 3$ (dark red) for CAT6. Since CATMOOD6 does not include CAT4, cells updated with $P = 2$ (yellow color) do not exist. Similarly to the previous configurations, the majority of cells updated with a first order schemes are located along the main shock waves.

Configuration 17. The computed density is reported in Figure 14 and Figure 15. This test presents a small vortex type of structure along with first and secondary waves emanating from the quadruple point. We can easily observe that CATMOOD6 can capture the vortex and the waves. Contrarily a diffusive scheme would require many more cells to get to this accuracy. Here the HLLC flux is adopted in the first order scheme of the CATMOOD6 scheme. The percentage of troubled cells are of the same order than previously, namely $97 - 98\%$ are updated with 6th order of accuracy and 2% and 1% with 2nd or 1st orders for CATMOOD6, (see Figure 15a). The extra cost compared to an unlimited CAT6 scheme is therefore acceptable. Table 4 displays, in its first row, the computational cost expressed in seconds for the Configuration 17 concerning Rusanov, HLLC, ACAT6, and CATMOOD6 schemes. Meanwhile, in the second row are presented the ratio between the computational costs with respect to the Rusanov scheme cost. Figure 14 clearly demonstrates the remarkable superiority of the adaptive *a posteriori* approach CAT+MOOD over the *a priori* technique ACAT. Despite exhibiting similar computational costs the CATMOOD6 numerical solution significantly outperforms the ACAT6 one in terms of accuracy. The cells in

	Rusanov	**HLLC**	**ACAT6**	**CATMOOD6**
CPU (s)	25.28	34.11	14563.40	12871.25
Ratio	1	1.35	576.08	509.15

Table 4: 2D Riemann problem configuration 17. First row: CPU computational costs expressed in seconds. Second row: ratio between the computational costs with respect to the Rusanov scheme.



(a) Rusanov flux

(b) HLLC

(c) ACAT6

(d) CATMOOD6

Figure 14: 2D Riemann problem from section 5.3 with initial condition configuration 17. Zoom of the numerical solution for density on the interval $[-1,1]\times[-1,1]$ adopting a mesh of $400\times400-$cells $t_{\text{final}}=0.3$ and CFL$=0.4$. Rusanov-flux (a); HLLC (b); ACAT6 (c); and CATMOOD6 (d).

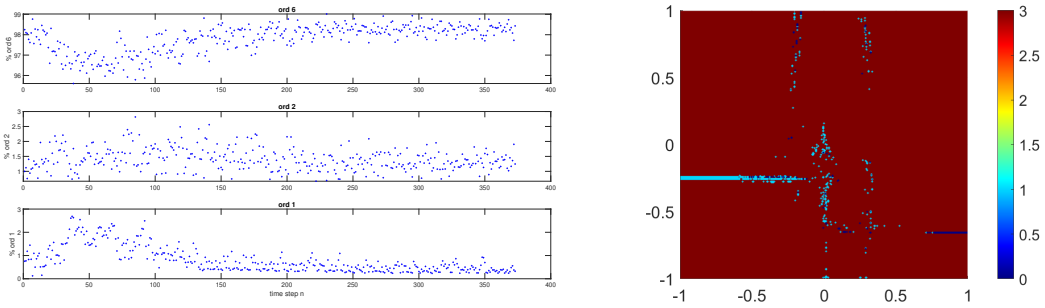
Figure 15b are colored by the value of P from the CAT2P scheme used, where $P=0$ (dark blue) represents the first order Rusanov flux, $P=1$ (light blue) CAT2 scheme and $P=3$ (dark red) for CAT6. We observe that only few bad cells in the interaction zone are detected, meaning that essentially CAT6 is almost always used even there. Obviously along the main discontinuities the low orders schemes are triggered as expected.

5.4. Mach 2000 astrophysical jet

Let us consider the high Mach number astrophysical jet problem [15, 14, 41]. The computational units for the jets are: 10^{11} km for length, 10^{10}s for time, 10km/s for speed, 100 H/cm^{-3} for density and 104.4 eV/cm^{-3} for energy and pressure. For this test in the interval $[0,1]\times[-0.25,0.25]$ we consider the following initial conditions:

$$(\rho^0,u^0,v^0,p^0)=\begin{cases}(5,800,0,0.4127) & \text{if } x=0 \text{ and } y\in[-0.05,0.05],\\ (0.5,0,0,0.4127) & \text{otherwise,}\end{cases} \tag{45}$$

where $\Gamma=5/3$.



(a) Percentage of cells updated by CAT6 (top), by CAT2 (center), and by HLLC (bottom).

(b) Local order of the density with CATMOOD6 at final time $t_{\text{final}} = 0.3$.

Figure 15: 2D Riemann problem from section 5.3 with initial condition configuration 17. Percentage of cells updated by CAT6, by CAT2, and by HLLC (left); local orders used by CATMOOD6 (right). The numerical solution has been computed on the interval $[-1, 1] \times [-1, 1]$ adopting a mesh of 400×400 cells, $t_{\text{final}} = 0.3$ and CFL$= 0.4$. The cells are colored by the value of P from the CAT2P scheme used to update it, where $P \in \{0, 1, 3\}$ and $P = 0$ represents the first order HLLC scheme.

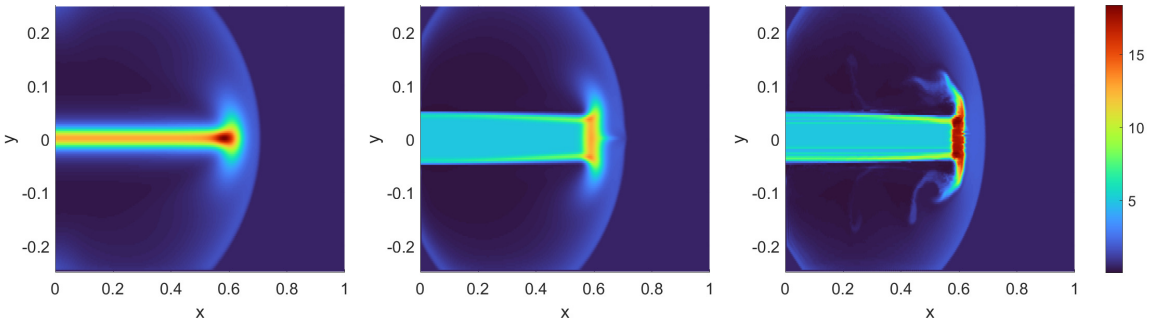


Figure 16: Mach 2000 from section 5.4. Numerical solution for density on the interval $[0, 1] \times [-0.25, 0.25]$ adopting a mesh of $300 \times 150-$ quadrangular cells $t_{\text{final}} = 0.001$ and a CFL$= 0.4$. The Rusanov flux (left); the HLLC (middle); and the CATMOOD6 with HLLC for the first order method (right).

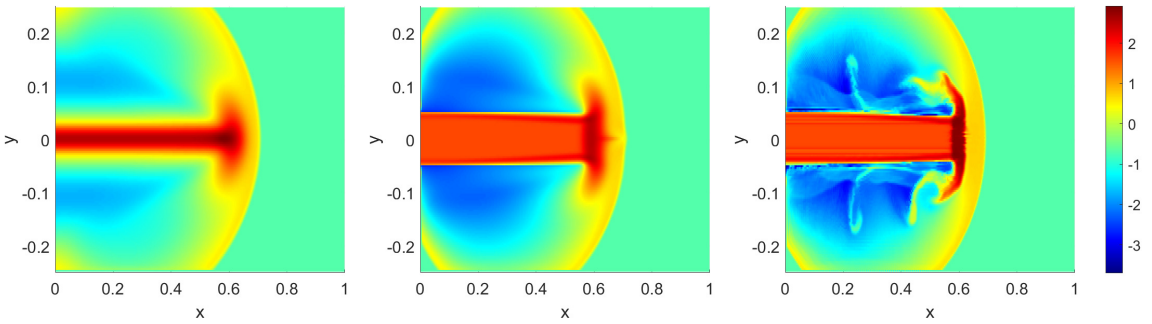
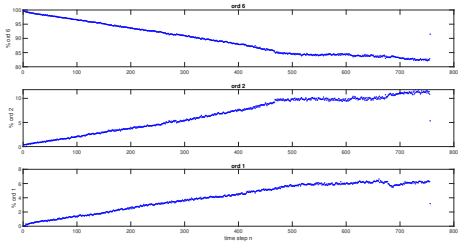
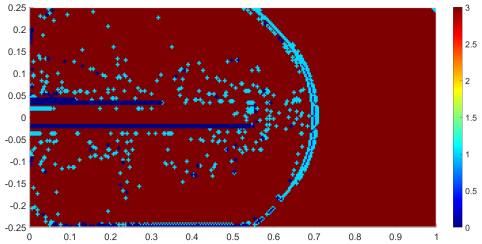


Figure 17: Mach 2000 from section 5.4. Numerical solution in the logarithm scale for density on the interval $[0, 1] \times [-0.25, 0.25]$ adopting a mesh of $300 \times 150-$cells $t_{\text{final}} = 0.001$ and CFL$= 0.4$. The Rusanov flux (Left); the HLLC (middle); and the CATMOOD6 with HLLC for the first order method (right).



(a) Percentage of cells updated by CAT6 (top), by CAT2 (center), and by HLLC (bottom).



(b) Local order of the density with CATMOOD6 at final time $t_{\text{final}} = 0.001$.

Figure 18: Mach 2000 from section 5.4 with initial condition (45). Percentage of cells updated by CAT6, by CAT2, and by HLLC (left); local orders used by CATMOOD6 (right). The numerical solution has been computed on the interval $[-1, 1] \times [-1, 1]$ adopting a mesh of 400×400 cells, $t_{\text{final}} = 0.001$ and CFL= 0.4. The cells are colored by the value of P from the CAT2P scheme used to update it, where $P \in \{0, 1, 3\}$ and $P = 0$ represents the first order HLLC scheme.

Initially we have a constant ambient gas at rest except for a segment in the left boundary of the domain for which the gas is 10 times denser and with a high x-component of velocity, corresponding to a Mach equals to 2000. Then, we impose inflow boundary conditions in this left segment, and outflow conditions otherwise. As such this simulates the penetration of a dense jet at hyper-velocity from a portion of the left boundary. This jet generates a bow shock ahead of the jet and a complex shape of the tip of the jet. Some reference solutions are reported in [14, 41, 36] for instance. Since the Mach number of the jet is extremely high, negative numerical pressure or density could easily appear during the computation, leading to the crash of the program. We compare the results of the simulations obtained by CATMOOD6 with those produced by some first order schemes. The mesh consists in 300×150 uniform quadrangular cells, and a CFL equals to 0.4 is adopted for a final time $t_{\text{final}} = 0.001$. Figures 16-17 present the numerical densities, with a logarithmic scale in the latter. The simulations are performed with CATMOOD6, first order Rusanov flux and first order HLLC schemes. The first order scheme for the MOOD cascade uses the HLLC numerical flux. Notice that the ACAT6 scheme fails for this test due to the generation of nonphysical states (negative pressure). Obviously all unlimited CAP2P schemes for $P \geq 1$ also fail.
The 1st order scheme can capture the bow shock position but the tip and body of the jet are truly diffused, especially if HLLC flux is not employed. A better shape is gained with HLLC type of flux. On the contrary CATMOOD6 is able to capture the complexity of the jet motion, its tip and the unstable lateral shear layers producing secondary waves and patterns into the post-shock region. These phenomena would be totally absent from low-accurate simulations for this mesh resolution, hence the need for truly accurate numerical methods. (They are indeed absent from the first order scheme results.) The small asymmetry in the profile is due to the high sensitivity of the solution of Euler equations to initial conditions, and to the intrinsic small lack of symmetry of CAT schemes. Indeed, since Euler equations do not have a minimal diffusion scale, which is present, for example, in Navier-Stokes equations, their solution may contain features at arbitrarily small scales. For such a reason several authors believe that the correct way to interpret the solution of Euler equations is in the sense of measure (see for example the recent book on the topic [13]). This phenomenon is even more clear with the logarithmic scale in Figures 17. We would like to emphasize that CATMOOD6 does not have any issue related to positivity, because the parachute scheme is indeed one of the the 1st order schemes which is robust enough. As such CAT+MOOD coupling is an almost 'fail-safe' strategy. This is not an obvious property of high-order methods using classical *a priori* limiters.

Figure 18a presents the percentage of troubled cells in CATMOOD6 as a function of time-steps. We plot the cells updated by the schemes in the MOOD cascade, that is with the unlimited CAT6 (top), CAT2 (center) and 1st order HLLC-base (bottom) schemes. In Figure 18b we plot the bad cells colored by the value of P from the CAT2P scheme used at final time. On this extreme test case the detection is able to select the cells along the main bow shock and the jet. Some bad cells are corrected in the complex interaction zone where Richtmyer-Meshkov instabilities develop.

This is a more advanced unsteady test case compared to the 2D Riemann problems seen in section 5.3 for which the solutions were self-similar. Here, the troubled cell evolution presents two phases: first, for about 500 time-steps the number of untroubled cells decreases linearly up to 85%, and, secondly, this number stagnates for about 250 time-steps. Interestingly, the number of cells updated with CAT2 scheme is of the order 10%, while truly demanding cells updated with the 1st order scheme represent about 6%. These two schemes seem to be useful in our CATMOOD6.
This test case is a single example which validates the CATMOOD6 scheme for extremely demanding simulations. Here, both accuracy and robustness are required.

6. Conclusion and Perspectives

In this paper we have presented an *a posteriori* way of limiting finite difference CAT2P schemes using MOOD paradigm. We have focused on CAT2P schemes of even orders devoted to solve Euler equations on 2D Cartesian mesh with a maximal order of accuracy 6. CAT2P schemes are nominally of order $2P$ on smooth solutions. Some extra dissipative mechanism must be supplemented to deal with steep gradients or discontinuous solutions, likewise for any high order scheme. Originally, CAT2P schemes were coupled with an automatic *a priori* limiter which blends high- and low-order fluxes as in [4]. However the difficulty with *a priori* limiters is the fact that they must (i) anticipate the occurrence of possible spurious oscillations from data at time t^n, (ii) tailor the appropriate amount of dissipation to stabilize the scheme, and, (iii) ensure that a physically admissible numerical solution is always produced. For 2nd order schemes such *a priori* limiters are available, but for higher orders they are not always performing well either on smooth solutions (lack of accuracy) or to ensure the physical admissibility (lack of robustness). In this work we rely on an *a posteriori* MOOD paradigm which computes an unlimited high-order candidate solution at time t^{n+1}, and, further detects troubled cells which are recomputed with a lower-order accurate scheme [9]. The detection procedure marks troubled cells according to Physical, Numerical and Computer admissible criteria which are at the core of our definition of an acceptable numerical solution. For a proof of concept, in this work, we have tested the so-called CATMOOD6 scheme based on the cascade of schemes: CAT6→CAT2→ 1st, where the last scheme is a first order robust scheme.
We have tested this scheme on a test suite of smooth solutions (isentropic vortex), simple shock waves (cylindrical Sedov blastwave), complex self-similar solutions involving contact, shock and rarefaction interacting waves (four state 2D Riemann problems) and, at last, on an extreme Mach 2000 astrophysical-like jet. CATMOOD6 has passed these tests. It has preserved the optimal accuracy on smooth parts of the solutions, an essentially-non-oscillatory behavior close to steep gradients, and, a physically valid solution. CATMOOD6 has been compared to some 1st order schemes to challenge its robustness and to unlimited CAT2P schemes to challenge its accuracy and cost. We have observed that the percentage of cells updated with the 6th order accurate scheme is in the range $85-100\%$, leaving only few percents to be re-computed by the low order schemes. As such CATMOOD6 has a total cost 20% superior to the unlimited CAT6 scheme on smooth solution, and, 10% less expensive than limited ACAT6 scheme on discontinuous solution (for which ACAT6 does not fail). From our test campaign we have observed that CATMOOD6 presents the robustness of its 1st order scheme, and, the accuracy of its 6th order one where appropriate. The detection procedure being able to sort out troubled cells from valid ones.

Concerning the perspectives, the extension to 3D is solely a question of implementation and testing in a parallel environment. At least, some extensions would be to consider different systems of PDEs, with source terms and well-balanced property, possibly stiff, or, more complex models such as Navier-Stokes equations.

Acknowledgments

We would like to thank Simone Chiochetti for providing some reference for Sedov problem and Prof. Enrique D. Fernandez-Nieto for providing some reference for 2D Riemann problem. This research has received funding from the European Union's NextGenerationUE – Project: Centro Nazionale HPC, Big Data e Quantum Computing, "Spoke 1" (No. CUP E63C22001000006). E. Macca was partially supported by GNCS No. CUP E53C22001930001 Research Project "Metodi numerici per problemi differenziali multiscala: schemi di alto ordine, ottimizzazione, controllo". E. Macca and G. Russo are members of the INdAM Research group GNCS. The research of C. Parés is part of the research project PDC2022-133663-C21 funded by MCIN/AEI/10.13039/501100011033 end the European Union 'Next GenerationEU/PRTR'and it was partially supported by the Regional Government of Andalusia through the research group FQM-216.

References

[1] S. Avgerinos, M.J. Castro, E. Macca, and G. Russo. A semi-implicit finite volume method for the Exner model of sediment transport. *Journal of Coputational Physics, Major revision*, 2023.

[2] P. Bacigaluppi, R. Abgrall, and S. Tokareva. "a posteriori" limited high order and robust schemes for transient simulations of fluid flows in gas dynamics. *Journal of Computational Physics*, 476, 2023.

[3] H. Carrillo, E. Macca, C. Parés, and G. Russo. Well-Balanced Adaptive Compact Approximate Taylor methods for systems of balance laws. *Journal of Computational Physics*, 478, 2023.

[4] H. Carrillo, E. Macca, C. Parés, G. Russo, and D. Zorío. An order-adaptive compact approximate taylor method for systems of conservation law. *Journal of Computational Physics*, 438:31, 2021.

[5] H. Carrillo and C. Parés. Compact approximate taylor methods for systems of conservation laws. *J. Sci. Comput.*, 80:1832–1866, 2019.

[6] H. Carrillo, C. Parés, and D. Zorío. Approximate Taylor methods with fast and optimized weighted essentially non-oscillatory reconstructions. *arXiv:2002.08426v1 [math.NA] 19 Feb 2020*, 2020.

[7] P.G. Ciarlet. Discrete maximum principle for finite-difference operators. *Aeq. Math.*, 4:338–352, 1970.

[8] S. Clain, S. Diot, and R. Loubère. A high-order finite volume method for systems of conservation laws – multi-dimensional optimal order detection (MOOD). *J. Comput. Phys.*, 230(10):4028 – 4050, 2011.

[9] S. Diot, S. Clain, and R. Loubère. Improved detection criteria for the multi-dimensional optimal order detection (MOOD) on unstructured meshes with very high-order polynomials. *Computers and Fluids*, 64:43 – 63, 2012.

[10] S. Diot, R. Loubère, and S. Clain. The MOOD method in the three-dimensional case: Very-high-order finite volume method for hyperbolic systems. *International Journal of Numerical Methods in Fluids*, 73:362–392, 2013.

[11] M. Dumbser, D. Balsara, E.F. Toro, and C.D. Munz. A unified framework for the construction of one-step finite-volume and discontinuous Galerkin schemes. *Journal of Computational Physics*, 227:8209–8253, 2008.

[12] C. Enaux, M. Dumbser, and E.F. Toro. Finite volume schemes of very high order of accuracy for stiff hyperbolic balance laws. *Journal of Computational Physics*, 227(2):3971–4001, 2008.

[13] Eduard Feireisl, Mária Lukáčová-Medviďová, Hana Mizerová, and Bangwei She. *Numerical analysis of compressible fluid flows*, volume 20. Springer, 2021.

[14] Y. Ha, C. Gardner, A. Gelb, and C.W. Shu. Numerical simulation of high Mach number astrophysical jets with radiative cooling. *J. Sci. Comput.*, 24:597–612, 2005.

[15] Youngsoo Ha and Carl L. Gardner. Positive scheme numerical simulation of high mach number astrophysical jets. *Journal of Scientific Computing*, 34(3):247–259, October 2007.

[16] C. Hirsch. *Numerical computation of internal and external flows (volume 1): fundamentals of numerical discretization.* John Wiley & Sons, Inc., New York, NY, USA, 2007.

[17] C. Hirsch. *Numerical computation of internal and external flows (volume 2): the fundamentals of computational fluid dynamics.* John Wiley & Sons, Inc., New York, NY, USA, 2007.

[18] F. Kemm. A comparative study of tvd-limiters - well-known limiters and an introduction of new ones. *Internationale Journal of Numerical Methods in Fluids*, 67:404–440., 2010.

[19] P. Lax and B. Wendroff. Systems of conservation laws. *Communications Pure and Applied Mathematics*, 13(2):217–237, 1960.
[20] P. Lax and B. Wendroff. Difference schemes for hyperbolic equations with high order accuracy. *Communications Pure and Applied Mathematics*, XVII(2):381–393, 1964.
[21] P. D. Lax. Hyperbolic systems of conservation laws, II. *CPAM*, 10:537–566, 1957.
[22] R.J. LeVeque. *Finite volume methods for hyperbolic problems*. Cambridge Texts in Applied Mathematics. Cambridge University Press., 1 edition, 2002.
[23] R.J. LeVeque. *Finite difference methods for ordinary and partial differential equations: steady-state and time-dependent problems (Classics in Applied Mathematics)*. Society for Industrial and Applied Mathematics, Philadelpia, PA. USA., 1 edition, 2007.
[24] Knut-Andreas Lie. Two-dimensional riemann problems for ideal gas dynamics, 2002.
[25] R. Loubère, M. Dumbser, and S. Diot. A new family of high order unstructured MOOD and ADER finite volume schemes for multidimensional systems of hyperbolic conservation laws. *Communication in Computational Physics*, 16:718–763, 2014.
[26] R. Loubère, E. Macca, C. Parés, and G. Russo. CAT-MOOD methods for conservation laws in one space dimension. In C. Parés, M.J.Castro, M.L. Muñoz, and T. Morales de Luna, editors, *Theory, Numerics and Applications of Hyperbolic Problems*, SEMA-SIMAI Springer Series, 2023. Proceedings of HYP2022.
[27] R. Loubère and M.J. Shashkov. A subcell remapping method on staggered polygonal grids for arbitrary-lagrangian–eulerian methods. *Journal of Computational Physics*, 209(1):105–138, 2005.
[28] E. Macca and G. Russo. Boundary effects on wave trains in the Exner model of sedimental transport. *Bollettino Unione Matematica Italiana*, 2023.
[29] Robert W MacCormack. The effect of viscosity in hypervelocity impact cratering. In *Frontiers of Computational Fluid Dynamics 2002*, pages 27–43. World Scientific, 2002.
[30] C.W. Schulz-Rinne. Classification of the riemann problem for two-dimensional gas dynamics. *SIAM Journal on Mathematical Analysis*, 24(1):76–88, 1993.
[31] T. Schwartzkopff, C. D. Munz, and E.F. Toro. A high-order approach for linear hyperbolic systems in 2d. *Journal of Scientific Computing*, 17:231–240, 2002.
[32] L.I. Sedov. *Similarity and Dimensional Methods in Mechanics*. Academic Press, New York, 1. edition, 1959.
[33] Chi-Wang Shu and Stanley Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *Journal of computational physics*, 77(2):439–471, 1988.
[34] C.W. Shu. Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic Conservation Laws. *NASA/CR-97-206253 ICASE Report No.97-65*, November 1997.
[35] P. K. Sweby. High resolution schemes using flux limiters for hyperbolic conservation laws. *SIAM J. Numer. Anal.*, 21 (5):995–1011., 1984.
[36] Siengdy Tann, Xi Deng, Yuya Shimizu, Raphaël Loubère, and Feng Xiao. Solution property preserving reconstruction for finite volume scheme: a boundary variation diminishing+multidimensional optimal order detection framework. *International Journal for Numerical Methods in Fluids*, 92(6):603–634, 2020.
[37] V.A. Titarev and E.F. Toro. ADER: Arbitrary high order Godunov approach. *Journal of Scientific Computing*, 17:609–618, 2002.
[38] E.F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics*. Springer, third edition, 2009.
[39] E.F. Toro, R.C. Millington, and L.A.M Nejad. Towards very high order Godunov schemes. *Godunov Methods. Theory and Applications E.F. Toro ed., Kluwer/Plenum Academic Publishers*, pages 907–940, 2001.
[40] B. Wendroff. Theoretical numerical analysis. *Academic Press, New York*, 1966.
[41] X. Zhang and C.W. Shu. On positivity-preserving high order discontinuous Galerkin schemes for compressible Euler equations on rectangular meshes. *J. Comput. Phys.*, 229:8918–8934, 2010.
[42] D. Zorío, A. Baeza, and P. Mulet. An approximate Lax-Wendroff-type procedure for high order accurate scheme for hyperbolic conservation laws. *J. Sci. Comput.*, 71(1):246–273, 2017.

7. Appendices

7.1. Fourth order version – CAT4

For a complete presentation of the general procedure to compute the $k-$th time derivative of flux, we detail hereafter the CAT4 scheme, that is with $P = 2$. Unlike CAT2 description, CAT4 scheme's description presents the high-order CAT automatism by introducing in detail the development of the iterative procedure used to compute the time derivatives of the flux. With this in mind, see Figure 19 to have a graphic idea of the necessary stencil.

The iterative CAT4 algorithm proceeds with the following steps to compute the flux $F^P_{i+\frac{1}{2}}$:

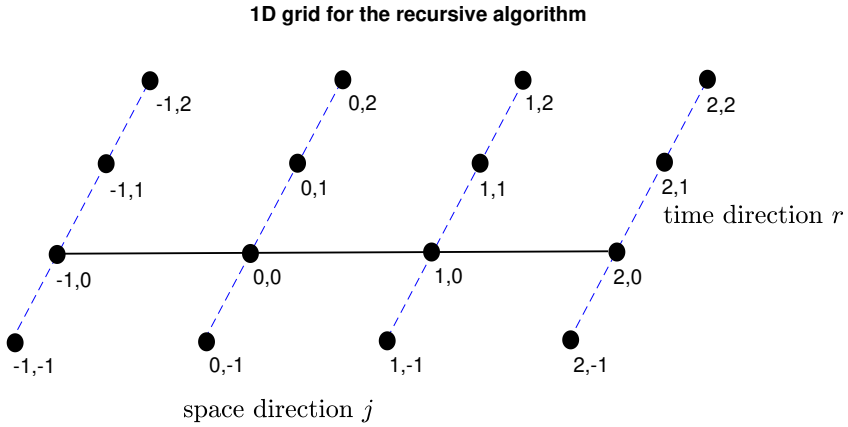


Figure 19: Local space-time grid where approximations of U are computed to calculate $F^P_{i+1/2}$ with $P = 2$. For simplicity a pair j, r represents the point (x_{i+j}, t_{n+r}). Taylor expansions in time are used to obtain these approximations following the blue lines. These Taylor expansions are centered in the points lying on the black line.

Step 1: Compute $f^{(0)}_{i+\frac{1}{2}}$ adopting the interpolation formula on stencil $\mathcal{S}^2_{i+\frac{1}{2}} = \left\{u^n_{i-1}, u^n_i, u^n_{i+1}, u^n_{i+2}\right\}$, so that $-P+1 = -1 \leq p \leq 2 = P$, as:

$$f^{(0)}_{i+\frac{1}{2}} = \sum_{p=-1}^{P=2} \gamma^{0,\frac{1}{2}}_{2,p}\, f^n_{i+p},$$

where $\gamma^{0,\frac{1}{2}}_{2,p}$ is the interpolation coefficient for $P = 2$ and label p.

Step 2: Compute the first time derivative of u at time t^n at position x_{i+p} with $p = -1, \ldots, 2$ through the numerical compact Cauchy-Kovaleskaya identity

$$\partial^k_t u = -\partial_x f^{(k-1)}(u), \quad k = 1, 2, \ldots \tag{46}$$

For $k = 1$ we have:

$$u^{(1)}_{i,j} = -\frac{1}{\Delta x} \sum_{j=-1}^{2} \gamma^{1,0}_{2,j} f^n_{i+j}.$$

Step 3: Compute the Taylor expansion of u in time truncated at first term at time t^{n+r} with $r = -1, \ldots, 2$ for each position x_{i+j} with $j = -1, \ldots, 2$ as:

$$u^{1,n+r}_{i,j} = u^n_{i+j} + r\Delta t\, u^{(1)}_{i,j}.$$

Step 4: Compute the first time derivative of f at time t^n at position x_{i+j} with $j = -1, \ldots, 2$ using the four fluxes $f^{1,n+r}_{i,j} = f\left(u^{1,n+r}_{i,j}\right)$ as:

$$f^{(1)}_{i,j} = \frac{1}{\Delta t} \sum_{r=-1}^{2} \gamma^{1,0}_{2,r}\, f^{1,n+r}_{i,j}.$$

Step 5: Compute $f^{(1)}_{i+\frac{1}{2}}$ adopting the interpolation formula on stencil $\mathcal{S}^2_{i+\frac{1}{2}}$ as:

$$f^{(1)}_{i+\frac{1}{2}} = \sum_{j=-1}^{2} \gamma^{0,\frac{1}{2}}_{2,j}\, f^{(1)}_{i,j};$$

Step 6: Compute the second time derivative of u at time t^n and position x_{i+j} with $j = -1, \dots, 2$ from the first time derivatives of f using the (46) as:

$$u_{i,j}^{(2)} = -\frac{1}{\Delta x} \sum_{s=-1}^{2} \gamma_{2,s}^{1,j} f_{i,s}^{(1)};$$

Step 7: Compute the Taylor expansion in time truncated at second term at time t^{n+r} with $r = -1, \dots, 2$ for each position x_{i+j} with $j = -1, \dots, 2$ as:

$$u_{i,j}^{2,n+r} = u_{i+j}^{n} + r\Delta t u_{i,j}^{(1)} + \frac{(r\Delta t)^2}{2} u_{i,j}^{(2)};$$

Step 8: Compute the second time derivative of f at time t^n at position x_{i+j} with $j = -1, \dots, 2$ using the four fluxes $f_{i,j}^{2,n+r}$ as:

$$f_{i,j}^{(2)} = \frac{1}{\Delta t^2} \sum_{r=-1}^{2} \gamma_{2,r}^{2,0} f_{i,j}^{2,n+r}.$$

Step 9: Compute $f_{i+\frac{1}{2}}^{(2)}$ adopting the interpolation formula on stencil $\mathcal{S}_{i+\frac{1}{2}}^{2}$ as:

$$f_{i+\frac{1}{2}}^{(2)} = \sum_{j=-1}^{2} \gamma_{2,j}^{0,\frac{1}{2}} f_{i,j}^{(2)}.$$

Step 10: Compute the third time derivative of U at time t_n and position x_{i+j} with $j = -1, \dots, 2$ from the second time derivatives of f using the (46) as:

$$u_{i,j}^{(3)} = -\frac{1}{\Delta x} \sum_{s=-1}^{2} \gamma_{2,s}^{1,j} f_{i,s}^{(2)};$$

Step 11: Compute the Taylor expansion in time truncated at third term at time t^{n+r} with $r = -1, \dots, 2$ for each position x_{i+j} with $j = -1, \dots, 2$ as:

$$u_{i,j}^{3,n+r} = u_{i+j}^{n} + r\Delta t u_{i,j}^{(1)} + \frac{(r\Delta t)^2}{2} u_{i,j}^{(2)} + \frac{(r\Delta t)^3}{6} u_{i,j}^{(3)}.$$

Step 12: Compute the third and last time derivative of f at time t^n at position x_{i+j} with $j = -1, \dots, 2$ using the four fluxes $f_{i,j}^{3,n+r}$ as:

$$f_{i,j}^{(3)} = \frac{1}{\Delta t^3} \sum_{r=-1}^{2} \gamma_{2,r}^{3,0} f_{i,j}^{3,n+r}.$$

Step 13: Compute $f_{i+\frac{1}{2}}^{(3)}$ adopting the interpolation formula on stencil $\mathcal{S}_{i+\frac{1}{2}}^{2}$ as:

$$f_{i+\frac{1}{2}}^{(3)} = \sum_{j=-1}^{2} \gamma_{2,j}^{0,\frac{1}{2}} f_{i,j}^{(3)};$$

Step 14: Deduce $F_{i+\frac{1}{2}}^{2}$ from (20) as:

$$F_{i+\frac{1}{2}}^{2} = f_{i+\frac{1}{2}}^{(0)} + \Delta t\, f_{i+\frac{1}{2}}^{(1)} + \frac{\Delta t^2}{2} f_{i+\frac{1}{2}}^{(2)} + \frac{\Delta t^3}{6} f_{i+\frac{1}{2}}^{(3)}$$

7.2. Computational complexity

In this section we focus on the local computational complexity for the CAT2P scheme applied to scalar case. With this in mind, observing that the size of the stencil $S^P_{i+\frac{1}{2}}$ is $2P$, we can divide the CAT algorithm into four parts and notice that:

1. $f^n_{i,j} = f(u^n_{i+j})$ for all $j = -P+1, \dots, P$.
2. For $k = 1, \dots, 2P-1$, the procedure to compute $u^{(k)}_{i,j}$, $u^{k,n+r}_{i,j}$, $f(u^{k,n+r}_{i,j})$ and $f^{(k)}_{i,j}$ does not depend explicitly on k (see expressions (2a-2d) below).
3. For $k = 0, \dots, 2P-1$, the computational formula for the approximation of the $(k-1)$-th time derivative of the flux at position $x = x_{i+1/2}$, $f^{(k-1)}_{i+1/2} = A^{0,1/2}_P\left(f^{(k-1)}_{i,*}, \Delta x\right) = \sum_{j=-P+1}^{P} \gamma^{0,\frac{1}{2}}_{P,j} f^{(k-1)}_{i,j}$, does not depend explicitly on k, since the interpolatory formula is invariant for each k.
4. Compute $F^P_{i+\frac{1}{2}}$, u^{n+1}_i and precomputed constants such as Δt^k or $\Delta t^k/k!$, etc.

For this reason, the computational complexity is so structured:

1. $f^n_{i,j} = f(u^n_{i+j})$ for all $j = -P+1, \dots, P$. 1 evaluation multiplied by the size of u[5].
2. For $k = 1, \dots, 2P-1$
   (a) $u^{(k)}_{i,j} = -\frac{1}{\Delta x}\sum_{s=-P+1}^{P} \gamma^{1,j}_{P,s} f^{(k-1)}_{i,s}$. $2P(2P+1)$ flop for each k, $2P$ related to j and $2P+1$ related to the summation.
   (b) $u^{k,n+r}_{i,j} = u^n_{i+j} + \sum_{m=1}^{k} \frac{(r\Delta t)^m}{m!} u^{(m)}_{i,j}$ under the assumption $u^{k,n+r}_{i,j} = u^{k-1,n+r}_{i,j} + \frac{c^k_r}{k!} u^{(k)}_{i,j}$ where $c_r = r\Delta t$ and $u^{0,n+r}_{i,j} = u^n_{i+j}$. $(2P)(2P-1)$ flop for each k under the assumption that $c^k_r/k!$ are precomputed and, when $r = 0$, $u^{k,n}_{i,j} \equiv u^n_{i+j}$.
   (c) $f^{k,n+r}_{i,j} = f(u^{k,n+r}_{i,j})$. $(2P)(2P-1)$ evaluations for each k[6], $(2P)$ related to j and $(2P-1)$ related to r. When $r = 0$, $f(u^{k,n+r}_{i,j}) = f(u^n_{i+j})$ and the computational complexity relapses to expression (1).
   (d) $f^{(k)}_{i,j} = A^{k,j}_P\left(f^{k,*}_{i,j}, \Delta t\right) = \frac{1}{\Delta t^k}\sum_{r=-P+1}^{P} \gamma^{k,0}_{P,r} f^{k,n+r}_{i,j}$. $2P(2P+1)$ flop for each k under the assumption that Δt^k are precomputed.
3. For $k = 1, \dots, 2P$, $f^{(k-1)}_{i+\frac{1}{2}}$. $2P$ flop.
4. The computational complexity of $F^P_{i+\frac{1}{2}}$, u^{n+1}_i and precomputed terms
   (a) $F^P_{i+1/2} = \sum_{k=1}^{2P} \frac{\Delta t^{k-1}}{k!} f^{(k-1)}_{i+1/2}$ under the assumption that $\Delta t^{k-1}/k!$ are precomputed. $(2P-1)$ flop since for $k = 1$ there are no products.
   (b) $u^{n+1}_i = u^n_i - \frac{\Delta t}{\Delta x}\left(F^P_{i+1/2} - F^P_{i-1/2}\right)$. 2 flop.
   (c) for $k = 1, \dots, 2P-1$ Δt^k. $k-1$ flop.
   (d) for $k = 1, \dots, 2P-1$ and for $r = -P+1, \dots, P$, with $r \neq 0$, $c^k_r/k!$ where $c_r = r\Delta t$. $(k+1)(2P-1)$ flop.
   (e) for $k = 1, \dots, 2P-1$ $\Delta t^{k-1}/k!$. 1 flop.

Finally, the computational cost differentiating between cost due to the function evaluation n_{func} and flops n_{flop} is given by

1. $n_{\text{func}} \leftarrow 1$ evaluation.
2. (a) $n_{\text{flop}} \leftarrow (2P)^3 - (2P)$ flop;

[5] $M \times N$ for a discrete system with $M-$variables and N cells.

[6] The evaluations of $f(u^{k,n+r}_{i,j})$ are much expensive than $f(u^n_{i+j})$. Indeed, for each k the size of $f(u^{k,n+r}_{i,j})$ is $M \times N$, the size of $f(u^{k,n+r}_{i,j})$ is $M \times N \times 2P \times 2P-1$, respectively for j and r. For the scalar case $M = 1$.

(b) $n_{\text{flop}} \leftarrow n_{\text{flop}} + (2P)^3 - 2(2P)^2 + (2P)$ flop;
(c) $n_{\text{func}} \leftarrow n_{\text{func}} + (2P)^3 - 2(2P)^2 + (2P)$ evaluations;
(d) $n_{\text{flop}} \leftarrow n_{\text{flop}} + (2P)^3 - (2P)$ flop.

3. $n_{\text{flop}} \leftarrow n_{\text{flop}} + (2P)^2$ flop.
4. (a) $n_{\text{flop}} \leftarrow n_{\text{flop}} + (2P) - 1$ flop;
   (b) $n_{\text{flop}} \leftarrow n_{\text{flop}} + 2$ flop;
   (c) $n_{\text{flop}} \leftarrow n_{\text{flop}} + 0.5(2P)^2 - 1.5(2P)$ flop;
   (d) $n_{\text{flop}} \leftarrow n_{\text{flop}} + 0.5(2P^3) - 0.5(2P)$ flop;
   (e) $n_{\text{flop}} \leftarrow n_{\text{flop}} + (2P) - 1$ flop.

Finally, the operation count per cell per time step for CAT2P, $P > 1$, applied to the scalar case is $3.5(2P)^3 - 1.5(2P)^2 + (2P)$ flop plus $(2P)^3 - 2(2P)^2 + (2P) + 1$ function evaluations. For CAT2 this gives 24 flop and 3 function evaluations, while for CAT4 we get 204 flop and 37 function evaluations.

Scheme	P	Cost per flux $C(P) = 3.5(2P)^3 - 1.5(2P)^2 + (2P)$	Ratio $C(P+1)/C(P)$	Ratio CAT2P/CAT2
CAT2	1	29	1	1
CAT4	2	299	10.31	10.31
CAT6	3	1097	3.67	37.83
CAT8	4	2711	2.47	93.48
CAT10	5	5429	2.00	187.2
⋮	⋮	⋮	⋮	⋮
CAT2P	P	$C(P)$	1.00	∞

Table 5: Cost of the CAT schemes in term of number of operations to update one interface flux as a function of P.