Numerical resolution of a two-component compressible fluid model with interfaces

Bruno Després and Frédéric Lagoutière

February 11, 2005

Abstract

We study a totally conservative algorithm for moving interfaces and in a two-component compressible fluid. We propose to use the limited downwind scheme developed in [9] in order to avoid artificial numerical spreading of interfaces. The numerical treatment of the mixture is shown to be free of spurious oscillations near the contact discontinuity. Various numerical simulations show the interest of this approach, for interfaces in dimension 1, 2 and 3. Simplicity of the coding is an important feature of the algorithm. An application to dynamic mixing is also shown.

1 Introduction

We address a numerical scheme for handling interfaces in a two-component compressible fluid. In this work, we essentially focus on the numerical algorithm. We assume that the fluids are immiscible so that they are separated by interfaces for any time $T \geq 0$. In some cases, such as interface instabilities (Richtmyer-Meshkov or Kelvin-Helmholtz, e.g.), the length of the interface may increase dramatically in time, but, if no mixing process assumption is done, the fluids are expected to stay separated, which is not obvious to have on the numerical stage due to numerical diffusion. Interface tracking or interface reconstruction algorithms are differents methods whose aim is to keep or recover sharp interfaces. Among many algorithms for interface tracking, interface reconstruction or pseudo-interface reconstruction, let us refer to [12] for interface tracking, [31] for VoF type methods, [11] for the Ghost Fluids Method, and [24] for the SLIC algorithm, which is the first algorithm published for this kind of problems. In the case of a multi-component fluid problem with interfaces, the validity of these algorithms has been demonstrated since a long time: a recent review about VoF methods is [6] (see also references therein). Even if VoF type methods have many advantages, one can feel the need for a more partial-differential-equations based formulation of interface multi-component fluid problems: an example is the Level Set method [25]. An alternative to interface tracking or interface reconstruction is to solve the mass conservation of each component, or, equivalently,

to follow some concentration equations for the fluid components, considering equation

$$\partial_t(\rho c) + \mathrm{div}(\rho c \vec{u}) = 0, \tag{1}$$

where c is a mass fraction (of one of the two fluids), volume fraction or "color" function, ρ is the mass density of the global fluid, $\vec{u}$ a velocity. The two fluids are separated in the case where c takes only the values 0 and 1. To discretize the concentration equation (1) (cf. [3] for example) is then enough to compute the transport of each fluid. However this approach has the well-known drawback of standard Finite Volume numerical schemes: they are dissipative, especially for discontinuous initial profiles (and so they are known to introduce numerical dissipation, numerical mixing where an interface is expected). The goal of this paper is to propose a new algorithm to compute approximate solutions of compressible immiscible multi-fluid models. The method we develop in the following takes place in the framework of finite volume algorithm, that are known to be well adapted for compressible fluids. The difficulty is then to find a stable finite volume algorithm with the ability to detect and compute sharp discontinuities such as interfaces.

The main idea of this work is to discretize (1) with the limited downwind scheme (also called the Ultra-Bee scheme when applied to linear advection with constant velocity in [29]): see [10], [22], [9]. Other references are [4] and [30]. In [9] it is proved that the limited downwind scheme in dimension 1 is, in some sense, exact for the transport of discontinuous data, which is a non-dissipation property (cf. theorem 1 below). An x-y-z splitting strategy on Cartesian grids with the limited downwind scheme appears to be a solution for computations in dimension 3.

The general design of the algorithm developed relies on a "Lagrange-projection" or "Lagrange-re-mapping" operator splitting, which consists in solving first all the partial differential equations of the considered system in Lagrangian coordinates and then, in a second step, to project the solution on the fixed Eulerian grid (at each time step). The second part of this procedure, the projection part, can be viewed as an advection step, and allows to use the limited downwind scheme. The mass fraction of each component only evolve in the projection step: that is why the non-dissipativity of the limited downwind scheme is here recovered for the mass fraction (and thus for interfaces).

The Lagrange step of the algorithm is in the spirit of VoF methods when applied to multi-material flows with interfaces ([6], [19]), but the projection (or transport) step is completely different and probably simpler than many other algorithms. It is clearly the non-dissipation property of the transport algorithm used in this work that allows to compute sharp interfaces (located in only one cell in dimension one, with similar resolution in dimensions two and three). The remainder of the algorithm (i.e. the Lagrange part) is a more classical one: it can be viewed as a Roe scheme and it will not be extensively studied here. Note that this particular design, the Lagrange-projection splitting, has the advantages of allowing to get entropy inequalities rather simply, and to split non-linear fields (in the Lagrange part) from linear fields (in the transport part),

so that the anti-dissipative algorithm (limited downwind) for linear advection equation can be used for the advection of mass fractions.

A list of the mathematical and numerical properties of the algorithm proposed in this work is as follows.

Anti-dissipativity. The limited downwind scheme (17) allows to compute exact solutions of advection equation for initial conditions that are step functions, provided that the cells are small enough: at least 3 cells per step of the initial condition (see theorem 1). In dimension 2, the algorithm is exact for the transport of squares, and for patches of squares. In practice the algorithm is exact at machine accuracy for a lot of interface problems.

Absence of Oscillations near contact discontinuities. Pressure and velocity spurious oscillations do not appear at contact discontinuities (see theorem 3).

Conservativity in any dimension. The algorithm is conservative for each mass, for the total mass, for the total impulse and the total energy.

TVD properties. In dimension 1, the algorithm is TVD and L^∞-stable for mass fraction. In dimensions 3 and 2 the algorithm is only L^∞-stable for mass fraction and the TVD property is lost.

The coding is simple We use a directional splitting to get a two-D and three-D code. Applications to detonation problems is possible with good results (in preparation), so as to problems with more than two components.

It also possible to prove that the scheme the scheme verifies entropy properties under a CFL-like condition. It means the numerical value of the physical entropy of each component increases locally at each time step. In particular one has ${S_r}_j^{n+1} \geq \min_j({S_r}_j^n) \quad r = 1, 2$ and $\forall j, n$, where ${S_r}_j^n$ denotes the entropy of fluid $r = 1, 2$ in the cell j at time step n. See propositions 1 and 2 in the appendix, a detailed but quite lengthy proof is in [22].

This paper is organized as follows. In section 2 we present a family of models for two-component compressible fluids. Section 3 is devoted to the presentation of the main features of the limited downwind scheme. In section 4 we generalize the limited downwind scheme to the two-component fluid models and give some of its mathematical properties. In section 5 we present various numerical simulations in order to give an overview of the capabilities of the algorithm; we give in section 6 a simple example with dynamic mixing.

2 A family of models

We here briefly present the systems of partial differential equations for multi-fluids that will be considered in the whole paper. The basic model (for the global fluid, composed of different fluids with possibly different pressure and

temperature laws) is the Euler system: global mass, global momentum and total energy conservations. We now have to present the choices done for the mixture model (that is, inside the global fluid, how each fluid behaves). As already written, we also take into account the conservation of each component mass.

For the sake of simplicity, the presentation is restricted to a two-fluid model (see [22] for the general case) in dimension one. The density of the global fluid is denoted as ρ, and ρc_1 (resp. ρc_2) is the partial density of fluid number one (resp. number two), where c_1 (resp. c_2) is the mass fraction of fluid 1 (resp. 2). It means that $c_1 + c_2 = 1$. The velocity of the global fluid is denoted as u. We make the assumption that the velocities of both fluids, u_1 and u_2, are equal, that is

$$u_1 = u_2 = u. \tag{2}$$

Let us denote, as usual, e the total energy. The first equations we take are thus

$$\begin{cases} \partial_t \rho + \partial_x(\rho u) = 0 & \text{(total mass conservation)}, \\ \partial_t(\rho c_1) + \partial_x(\rho u c_1) = 0 & \text{(mass conservation of fluid 1)}, \\ \partial_t(\rho c_2) + \partial_x(\rho u c_2) = 0 & \text{(mass conservation of fluid 2)}, \\ \partial_t(\rho u) + \partial_x(\rho u^2 + p) = 0 & \text{(momentum conservation)}, \\ \partial_t(\rho e) + \partial_x(\rho u e + p u) = 0 & \text{(total energy conservation)}, \end{cases}$$

where p is the pressure in the global fluid and has to be determined by setting a mixture model, what we briefly do in the following lines. One important assumption that is now done is that there exists a scale at which the two fluids are separated. In other words it means that the pressure is not a function of the apparent densities $\rho_1 = \rho c_1$ and $\rho_2 = \rho c_2$, but is a function of the (true) densities: this consideration is based on the notion of additivity of volumes. We assume that any elementary volume Δv is the sum of two sub-volumes Δv_1 and Δv_2, $\Delta v = \Delta v_1 + \Delta v_2$, and that only the first (resp. second) fluid is present in Δv_1 (resp. Δv_2). Here we do not make any hypothesis on the shape of these Δv_1 and Δv_2, as in figure 1.

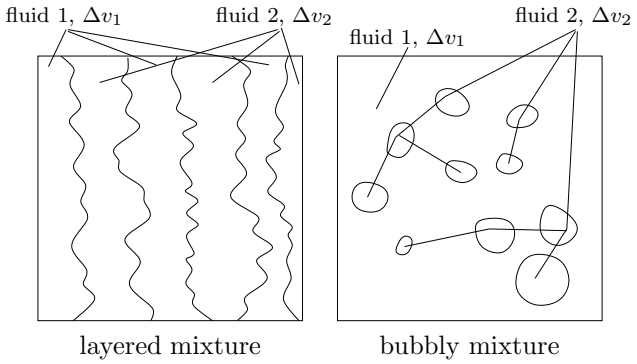


Figure 1: Some possible microscopic structures of the mixture.

The total mass in Δv is $\Delta m = \rho \Delta v$. Mass fractions c_1, c_2 and partial masses Δm_1 and Δm_2 are related through $c_1 = \frac{\Delta m_1}{\Delta m}$ and $c_2 = \frac{\Delta m_2}{\Delta m}$. Since $\Delta m_1 + \Delta m_2 = \Delta m$, one has $c_1 + c_2 = 1$. The true density of fluid number 1 (resp. 2) is denoted as $\overline{\rho}_1$ (resp. $\overline{\rho}_2$). It means that $\Delta m_1 = \overline{\rho}_1 \Delta v_1 = c_1 \rho \Delta v$ and $\Delta m_2 = \overline{\rho}_2 \Delta v_2 = c_2 \rho \Delta v$. We deduce that $\frac{1}{\rho}(\Delta v_1 + \Delta v_2) = (\frac{c_1}{\overline{\rho}_1} + \frac{c_2}{\overline{\rho}_2})\Delta v$. Thus the additivity of volumes is equivalent to $\frac{1}{\rho} = \frac{c_1}{\overline{\rho}_1} + \frac{c_2}{\overline{\rho}_2}$. Defining the specific volume of each fluid $\tau_{1,2} = \frac{1}{\overline{\rho}_{1,2}}$ this equation reads

$$\tau = c_1 \tau_1 + c_2 \tau_2, \tag{3}$$

which is, let us repeat it, the mathematical expression of the additivity of volumes and a consequence of the hypothesis that there exists a scale at which the fluids are separated.

In the same way, we assume the additivity of internal energies, that is

$$\varepsilon = c_1 \varepsilon_1 + c_2 \varepsilon_2,$$

where ε_1 and ε_2 are the internal energies of each fluid and ε the one for the global fluid, naturally leading to

$$e = c_1 \varepsilon_1 + c_2 \varepsilon_2 + \frac{1}{2} u^2 \tag{4}$$

The quantities τ_1, τ_2, ε_1, ε_2 allow to compute one pressure for each of the two component, using some Equations of States (EOS):

$$p_1 = p_1(\tau_1, \varepsilon_1) \text{ and } p_2 = p_2(\tau_2, \varepsilon_2). \tag{5}$$

For simplicity we will consider perfect gas approximations: $p_1 = (\gamma_1 - 1)\frac{\varepsilon_1}{\tau_1}$, $p_2 = (\gamma_2 - 1)\frac{\varepsilon_2}{\tau_2}$.

It remains to describe a mixing model in order to close the system of PDE, i.e. to furnish a way to compute the global pressure denoted as p. A large number of models are possible (leading to good mathematical properties: see [22] for this study), but we here only present two of them:
1) The "**isobar-isothermal**" closure

$$p_1 = p_2, \;\; T_1 = T_2 \tag{6}$$

where $C_{v1} T_1 = \varepsilon_1$ and $C_{v2} T_2 = \varepsilon_2$ for some constants $C_{v1} > 0$ and $C_{v2} > 0$.
2) The "**isobar-iso**δQ" model

$$p_1 = p_2, \;\; \delta Q_1 = \delta Q_2, \tag{7}$$

where, from the fundamental law of thermodynamics, we have $\delta Q_1 = T_1 dS_1 = d\varepsilon_1 + p_1 d\tau_1$ and $\delta Q_2 = T_2 dS_2 = d\varepsilon_2 + p_2 d\tau_2$. So it is possible to replace $\delta Q_1 = \delta Q_2$ by the incremental partial differential equation

$$\mathrm{D}_t \varepsilon_1 + p_1 \mathrm{D}_t \tau_1 = \mathrm{D}_t \varepsilon_2 + p_2 \mathrm{D}_t \tau_2, \tag{8}$$

where D_t denotes material derivative: $\mathrm{D}_t = \partial_t + u\partial_x$ in dimension one (for regular solutions). This system seems to be more complex than the preceding one. The reason why we use it is that it allows to obtain entropy inequalities after discretization (see paragraph B in the annex), and to have some very good numerical results, especially near interfaces and contact discontinuities, where no spurious oscillations are detected (theorem 3).

The corresponding whole systems are

$$\left\{\begin{array}{l} \partial_t\rho + \partial_x(\rho u) = 0, \\ \partial_t(\rho c_1) + \partial_x(\rho u c_1) = 0, \\ \partial_t(\rho c_2) + \partial_x(\rho u c_2) = 0, \\ \partial_t(\rho u) + \partial_x(\rho u^2 + p) = 0, \\ \partial_t(\rho e) + \partial_x(\rho u e + p u) = 0, \\ c_1\tau_1 + c_2\tau_2 = \tau = 1/\rho, \\ e = c_1\varepsilon_1 + c_2\varepsilon_2 + \frac{1}{2}u^2 \\ p_1 = p_2 = p, \\ \textbf{isothermal } T_1 = T_2 \text{ or } \textbf{iso-}\delta Q \ \mathrm{D}_t\varepsilon_1 + p_1\mathrm{D}_t\tau_1 = \mathrm{D}_t\varepsilon_2 + p_2\mathrm{D}_t\tau_2. \end{array}\right. \tag{9}$$

Among good properties of this models (which are not the subject of this presentation), let us mention

Lemma 1 These two models are hyperbolic (see [22] for the proof).

These two models are very similar. The isothermal model has the advantage to be fully conservative, while the non conservative iso-δQ model has the advantage to be independent on the determination of C_{v_1} and C_{v_2} because it does not involve the temperature. The second model in (9) is equivalent to the non-conservative partial differential equation

$$\begin{array}{l} \partial_t\varepsilon_1 + u\partial_x\varepsilon_1 + p(\partial_t\tau_1 + u\partial_x\tau_1) \\ = \partial_t\varepsilon_2 + u\partial_x\varepsilon_2 + p(\partial_t\tau_2 + u\partial_x\tau_2). \end{array} \tag{10}$$

When fluids 1 and 2 are separated by an interface, (9) is in some sense degenerated. It simply means that c_1 (resp. c_2) takes values only in $\{0, 1\}$: thus $c_1c_2 = 0$ everywhere. In this case the mixture equations are used for the treatment of mixed cells at the numerical stage. The numerical treatment of the interface configuration is responsible for the numerical mixing configuration. Figure 2 gives an example. Nevertheless (9) can also be used in the case of a true mixing zone, see section 6 for examples in dimension 1.

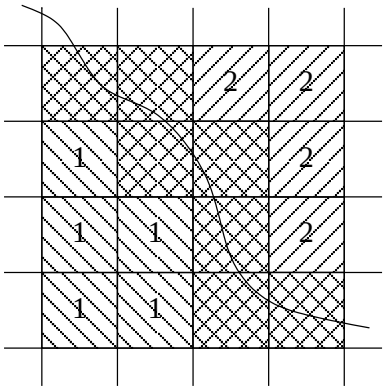


Figure 2: Interface and mixed cells

Let us now turn to the description of the algorithm we use to solve (9). We propose an anti-dissipative algorithm which is easier to understand on a simplified configuration: the case of pure advection. We thus present this simplified case in section 3 and the complete general scheme is written and studied in section 4.

3 The basic scheme for moving mass fractions

Let us assume in (9) that the velocity and the pressure are constant in space (and thus in time). It then reduces to pure transport. The numerical algorithm has to degenerate to the numerical solution of $\partial_t c + u\partial_x c = 0$. We begin by a presentation of some properties of the transport algorithm in dimension 1, more details are in [9].

Let us define a space cell size Δx and a time increment Δt. The solution c of the advection equation with initial condition c^0 is approximated by the constant-by-cell function with value c_j^n at time step n in the cell j. The finite volume algorithm reads

$$\begin{cases} \dfrac{c_j^{n+1} - c_j^n}{\Delta t} + u \dfrac{c_{j+\frac{1}{2}}^n - c_{j-\frac{1}{2}}^n}{\Delta x} = 0 \quad \forall j \in \mathbb{Z}, \forall n \in \mathbb{N}, \\ c_j^0 = \dfrac{1}{\Delta x} \displaystyle\int_{(j-1/2)\Delta x}^{(j+1/2)\Delta x} c^0(x)\, dx \quad \forall j. \end{cases} \tag{11}$$

The new value of the unknown is c_j^{n+1}. It is a function of c_j^n and of the fluxes $(c_{j+\frac{1}{2}}^n)$. As usual, a CFL condition is assumed: $|u|\Delta t \leq \Delta x$. It remains to choose a formula for the fluxes. Let us describe the main ingredient of our basic transport scheme: we here recall some features about the limited downwind scheme.

It may be useful to describe the motivation for the downwind (instead of the usual upwind) choice for the fluxes (of course keeping in mind that it is unstable and then will need to be limited to turn to a stable scheme).

First let us look at the very simple situation where the initial solution c^0 is a Heavyside function: this is the prototype of an interface. We also assume, for the sake of simplicity, that $u > 0$.

$$c_l^0 = 1, \quad \forall l \leq j, \text{ and } c_l^0 = 0, \quad \forall l > j. \tag{12}$$

Let us assume that the time step is not the maximal time step. For example $u\frac{\Delta t}{\Delta x} = \frac{1}{3} < 1$. For the initial condition (12), the exact solution at the first time step is $c^0(x - \frac{\Delta x}{3})$. After projection on the grid, it is $c_l^1 = \frac{1}{\Delta x}\int_{(l-\frac{1}{2})\Delta x}^{(l+\frac{1}{2})\Delta x} c^0(x - \frac{\Delta x}{3})dx$, that is,

$$c_l^1 = 1 \quad l \leq j, \quad c_{j+1}^1 = \frac{1}{3}, \quad c_l^1 = 0 \quad l > j+1. \tag{13}$$

At the second time step, the exact solution is $c^0(x - \frac{2\Delta x}{3})$ while the projected exact solution is $c_l^2 = \frac{1}{\Delta x}\int_{(l-\frac{1}{2})\Delta x}^{(l+\frac{1}{2})\Delta x} c^0(x - \frac{2\Delta x}{3})dx$ that is

$$c_l^2 = 1 \quad \forall l \leq j, \quad c_{j+1}^2 = \frac{2}{3}, \quad c_l^2 = 0 \quad \forall l > j+1. \tag{14}$$

After a third time step, the exact solution is $c^0(x - \Delta x)$. Its projection on the grid is

$$c_l^3 = 1, \quad \forall l \leq j+1, \text{ and } c_l^3 = 0, \quad \forall l > j+1, \tag{15}$$

and is again equal to the exact solution 4. Now we forget that (12), (13), (14) and (15) are some cell-averages of the exact solution, and consider that these numerical profiles are given by a finite volume scheme (11). If we try to define some numerical fluxes such that the scheme (11) applied to the initial condition (12) (resp. (13) or (14)) gives (13) (resp. (14) or (15)), a solution for the cell j is $c_{j-\frac{1}{2}}^{1,2} = 1$ and $c_{j+\frac{1}{2}}^{1,2} = 0$. Indeed it implies $c_{j+\frac{1}{2}}^1 = 0$, $c_j^1 = \frac{1}{3}$, $c_{j+1}^1 = 0$. So we arrive at the conclusion that in this situation the "exact numerical flux", between cell j and cell $j+1$, is equal to the down-winded value of the exact solution, that is $c_{j+\frac{1}{2}}^{1,2} = c_{j+1}^{1,2}$. Now we raise this simple fact is a general principle for the choice of the numerical flux. The numerical flux will be chosen as closed as possible to the down-winded value of the numerical solution. However it is well-known that the downwind linear scheme (i.e. $c_{j+\frac{1}{2}} = c_{j+1}$) is unstable. So we need to incorporate some stability and TVD (which stands for Total Variation Diminishing) notions in order to get a stable and convergent scheme. What we emphasize on is that it is possible to add some TVD constraints in the choice $c_{j+\frac{1}{2}} = c_{j+1}$ such that: if possible we take $c_{j+\frac{1}{2}} = c_{j+1}$; otherwise we take $c_{j+\frac{1}{2}}$ to be the closest as possible value to c_j. Following the approach presented in detail in [9] in dimension 1 and defining $m_j^n = \min(c_{j-1}^n, c_j^n)$, $M_j^n =$

$\max(c_{j-1}^n, c_j^n)$, we arrive at

$$\left\{\begin{array}{l} m_{j+1}^n \le c_{j+\frac{1}{2}} \le M_{j+1}^n, \quad \forall j \in \mathbb{Z}, \\ M_j^n + \frac{\Delta x}{u\Delta t}(c_j^n - M_j^n) \le c_{j+\frac{1}{2}}^n \le m_j^n + \frac{\Delta x}{u\Delta t}(c_j^n - m_j^n), \quad \forall j \in \mathbb{Z}, \\ |c_{j+1}^n - c_{j+\frac{1}{2}}^n| \text{ is minimum}, \quad \forall j \in \mathbb{Z}. \end{array}\right. \tag{16}$$

This minimization problem reduces to the explicit formula
first: compute some bounds

$$\left\{\begin{array}{l} b_j^n = \max(m_{j+1}^n, M_j^n + \frac{\Delta x}{u\Delta t}(c_j^n - M_j^n)), \\ B_j^n = \min(M_{j+1}^n, m_j^n + \frac{\Delta x}{u\Delta t}(c_j^n - m_j^n)). \end{array}\right.$$

then: compute the flux according to

$$c_{j+\frac{1}{2}}^n = \left\{\begin{array}{ll} b_j^n & \text{if } c_{j+1}^n < b_j^n, \\ c_{j+1}^n & \text{if } b_j^n \le c_{j+1}^n \le B_j^n, \\ B_j^n & \text{if } B_j^n < c_{j+1}^n. \end{array}\right. \tag{17}$$

Of course, an equivalent algorithm can be obtained in the case $u < 0$.

3.1 Properties of the scheme

We here just give an overview of the properties of this scheme. More details may be found in [22] and [9]. First we note that the limited downwind scheme given in (16) or (17) is equivalent to the Ultra-Bee limiter ([29], [17], [28]), in the sense that

$$c_{j+\frac{1}{2}}^n = c_j^n + \max(0, \min((\frac{1}{\mu} - 1)r_j^n), 1)(c_{j+1}^n - c_j^n), \quad r_j^n = \frac{c_j^n - c_{j-1}^n}{c_{j+1}^n - c_j^n}, \tag{18}$$

where $\mu = u\Delta t/\Delta x$ is the CFL number. The following result (proved in [9] and [22]) states that the limited downwind scheme is an exact scheme for a "dense" in L^1 set of functions, which are is staircase functions.

Theorem 1 Let us assume that the discrete function $(c_j^n)_{j\in\mathbb{Z}}$ is a staircase function. That is there exists $\alpha \in [0, 1[$ such that $\forall j \in \mathbb{Z}$, $c_{3j+1}^n = c_{3j}^n$ and $c_{3j+2}^n = \alpha c_{3j+1}^n + (1-\alpha)c_{3j+3}^n$. Then

either $0 \le \alpha + \mu < 1$**.** Let us set $0 \le \bar{\alpha} = \alpha + \mu - 1 \le 1$. Then for all j, $c_{3j+1}^{n+1} = c_{3j}^{n+1} = c_{3j}^n$ and $c_{3j+2}^{n+1} = (\bar{\alpha})c_{3j+1}^{n+1} + (1-\bar{\alpha})c_{3j+3}^{n+1}$;

or $1 \le \alpha + \mu < 2$ Let us set $0 \le \bar{\alpha} = \alpha + \mu - 1 \le 1$. Then for all j $c_{3j+2}^{n+1} = c_{3j+1}^{n+1} = c_{3j+1}^n$ and $c_{3j+3}^{n+1} = (\bar{\alpha})c_{3j+2}^{n+1} + (1-\bar{\alpha})c_{3j+4}^{n+1}$.

This theorem says that if the initial condition is a staircase function, the numerical solution is still a staircase function at any time step and is the projection on the grid of the exact solution, at any time. In theorem 1, the convex

combination coefficient α is the same between each three points step. It is possible to replace three points step ($c_{3j+1} = c_{3j}$, $c_{3j+2} = \alpha c_{3j+1} + (1-\alpha)c_{3j+2}$), by four (or more) points steps with non-constant α ($c_{3j+2} = c_{3j+1} = c_{3j}$, $c_{3j+3} = \alpha_{3j+2}c_{3j+2} + (1-\alpha_{3j+2})c_{3j+4}$).

We now turn to the same problem in dimension 2: $\partial_t c + u\partial_x c + v\partial_y c = 0$. We use a very simple extension of what we have just presented: a dimensional splitting (the so-called Alternate Direction method) of the two-dimensional equation in order to have only to solve one-dimensional numerical problems. The principle is to solve alternatively the one-dimensional equations $\partial_t c + u\partial_x c = 0$ and $\partial_t c + v\partial_y c = 0$. This method of course involves a Cartesian mesh. For each phase of the transport (x-phase and y-phase), we use the limited downwind scheme. Recall that $\mu = u\Delta t/\Delta x$ and let us define $\nu = v\Delta t/\Delta y$. The two-dimensional extension of theorem 1 is given in

Theorem 2 Let us assume that the 2-D discrete function $(c^n_{j,k})_{j\in\mathbb{Z}}$ is a staircase function in the sense that there exists $\alpha \in [0,1[$ and $\beta \in [0,1[$ such that $\forall j \in \mathbb{Z}$ and $\forall k \in \mathbb{Z}$,

- $c^n_{3j+1,k} = c^n_{3j,k}$ and $c^n_{3j+2,k} = \alpha c^n_{3j+1,k} + (1-\alpha)c^n_{3j+3,k}$;
- $c^n_{j,3k+1} = c^n_{j,3k}$ and $c^n_{j,3k+2} = \beta c^n_{j,3k+1} + (1-\beta)c^n_{j,3k+3}$.

Then

either $0 \le \alpha+\mu < 1$ **and** $0 \le \beta+\nu < 1$**.** Let us set $0 \le \bar\alpha = \alpha+\mu \le 1$ and $0 \le \bar\beta = \beta+\nu \le 1$. Then for all j and all k,

- $c^{n+1}_{3j+1,k} = c^{n+1}_{3j,k} = c^n_{3j,k}$ and $c^{n+1}_{3j+2,k} = (\bar\alpha)c^{n+1}_{3j+1,k} + (1-\bar\alpha)c^{n+1}_{3j+3,k}$;
- $c^{n+1}_{j,3k+1} = c^{n+1}_{j,3k} = c^n_{j,3k}$ and $c^{n+1}_{j,3k+2} = (\bar\beta)c^{n+1}_{j,3k+1} + (1-\bar\beta)c^{n+1}_{j,3k+3}$;

or $0 \le \alpha+\mu < 1$ **and** $1 \le \beta+\nu < 2$**.** Let us set $0 \le \bar\alpha = \alpha+\mu \le 1$ and $0 \le \bar\beta = \beta+\nu-1 \le 1$. Then for all j and all k,

- $c^{n+1}_{3j+1,k} = c^{n+1}_{3j,k} = c^n_{3j,k}$ and $c^{n+1}_{3j+2,k} = (\bar\alpha)c^{n+1}_{3j+1,k} + (1-\bar\alpha)c^{n+1}_{3j+3,k}$;
- $c^{n+1}_{j,3k+2} = c^{n+1}_{j,3k+1} = c^n_{j,3k+1}$ and $c^{n+1}_{j,3k+3} = (\bar\beta)c^{n+1}_{j,3k+2} + (1-\bar\beta)c^{n+1}_{j,3k+4}$;

or $1 \le \alpha+\mu < 2$ **and** $0 \le \beta+\nu < 1$**.** Let us set $0 \le \bar\alpha = \alpha+\mu-1 \le 1$ and $0 \le \bar\beta = \beta+\nu \le 1$. Then for all j and all k,

- $c^{n+1}_{3j+2,k} = c^{n+1}_{3j+1,k} = c^n_{3j+1,k}$ and $c^{n+1}_{3j+3,k} = (\bar\alpha)c^{n+1}_{3j+2,k} + (1-\bar\alpha)c^{n+1}_{3j+4,k}$;
- $c^{n+1}_{j,3k+1} = c^{n+1}_{j,3k} = c^n_{j,3k}$ and $c^{n+1}_{j,3k+2} = (\bar\beta)c^{n+1}_{j,3k+1} + (1-\bar\beta)c^{n+1}_{j,3k+3}$;

or $1 \le \alpha+\mu < 2$ **and** $1 \le \beta+\nu < 2$**.** Let us set $0 \le \bar\alpha = \alpha+\mu-1 \le 1$ and $0 \le \bar\beta = \beta+\nu-1 \le 1$. Then for all j and all k,

- $c^{n+1}_{3j+2,k} = c^{n+1}_{3j+1,k} = c^n_{3j+1,k}$ and $c^{n+1}_{3j+3,k} = (\bar\alpha)c^{n+1}_{3j+2,k} + (1-\bar\alpha)c^{n+1}_{3j+4,k}$;
- $c^{n+1}_{j,3k+2} = c^{n+1}_{j,3k+1} = c^n_{j,3k+1}$ and $c^{n+1}_{j,3k+3} = (\bar\beta)c^{n+1}_{j,3k+2} + (1-\bar\beta)c^{n+1}_{j,3k+4}$;

A proof is given in [22]; note that this is a straightforward consequence of theorem 1. This result expresses that an initial staircase condition is exactly numerically advected by the limited downwind scheme with directional splitting: the solution remains a staircase function and the values on the stairs are advected at the right velocity.

The extension to dimension 3 is let to the reader.

3.2 Numerical results in dimension 1 for the advection case

Let us now present some numerical results in order to give an overview of the capability of the scheme. These test cases have been computed with periodic boundary conditions and unit velocity $u = 1$.

3.2.1 Characteristic function, figure 3

The first result is for a characteristic function in the interval $[0, 1]$. The initial condition is $c^0(x) = 1$ if $0.4 \leq x \leq 0.6$ and $c^0(x) = 0$ otherwise. It illustrates the theorem (1) of exact advection. We see that this approximate solution is not at all dissipated (figure 3, CFL number $u\Delta t/\Delta x$ is 0.1)). Note that the result with the limited downwind scheme is exactly the projection of the continuous solution on the mesh. It is in this sense that we say that the limited downwind scheme is optimal for this kind of profiles. This behavior is particularly interesting for interface computations in 2-components fluids. The figure allows comparison with the classical upwind scheme and the Super-Bee limiter.

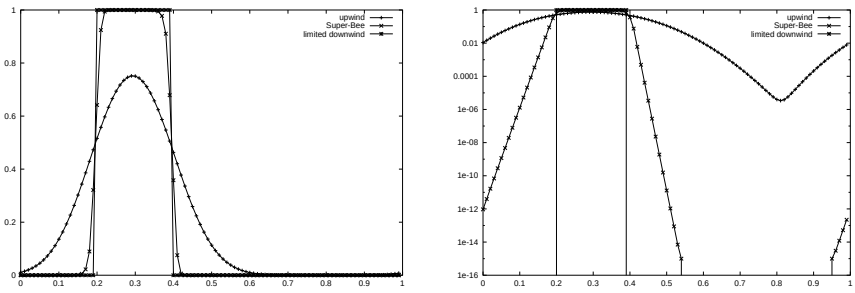


Figure 3: Initial condition and results (limited downwind, Super-Bee limiter, upwind) for $t = 1$ (after one period). Log-scaled on the right. The initial condition (also exact solution) and the limited downwind result are confounded.

3.2.2 Affine mixing zone, figure 4

The initial condition is periodic and such that $c^0(x) = 1 - 2x$ if $0 \leq x < 1/2$ and $c^0(x) = 2(x - 1/2)$ if $1/2 \leq x < 1$. The solution is computed for time $t = 100$ (after 100 periods) and the Courant number is 0.3. One sees that the upwind

is highly dissipative, so that the numerical profile is a straight line for large times. On the contrary the Super-Bee scheme is subjected to the so-called over-compressivity pathology: the numerical profile is squared for large times. The important point is that the limited downwind scheme (also named Ultra-Bee scheme) is so over-compressive that local instantaneous overcompressivity takes place everywhere just after $t = 0$. The scheme is highly "oscillating" around smooth profiles, since this scheme is close to the linear downwind scheme. After this linearly unstable process, the scheme reaches its non-linear stability (TVD stability) and becomes exact.

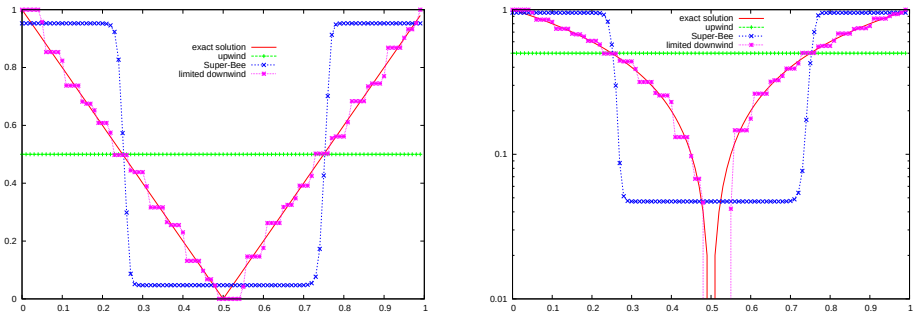


Figure 4: Upwind, Super-Bee ($t = 1000$), limited downwind: 100 cells. Log-scaled on the right.

3.3 Numerical results in dimension 2 for the advection case

Let us now present some numerical results in order to give an overview of the capability of the scheme. These test cases have been computed with periodic boundary conditions.

3.3.1 Transport of a square in dimension 2

We present a few numerical results for the advection equation in dimension 2 in figure 5. The spatial domain we here consider is the square $[0, 1[\times[0, 1[\in \mathbb{R}^2$, again with periodic boundary conditions on $x = 0$, $x = 1$, $y = 0$ and $y = 1$. We choose as velocity the diagonal vector: $u = 1$, $v = 1$. The result reported in figure 5 for the characteristic function $\mathbb{1}_{[0.1,0.5]\times[0.3,0.7]}$ is equivalent to that obtained for one-dimensional advection: we have absolutely no dissipation for this test case. The numerical solution is equal to the exact solution after one revolution. Once more it is in accordance with theoretical properties of the scheme, see theorem 2. This property of exact advection of square characteristic functions has been proved for all directions $\vec{u} = (u, v) \in \mathbb{R}^2$.

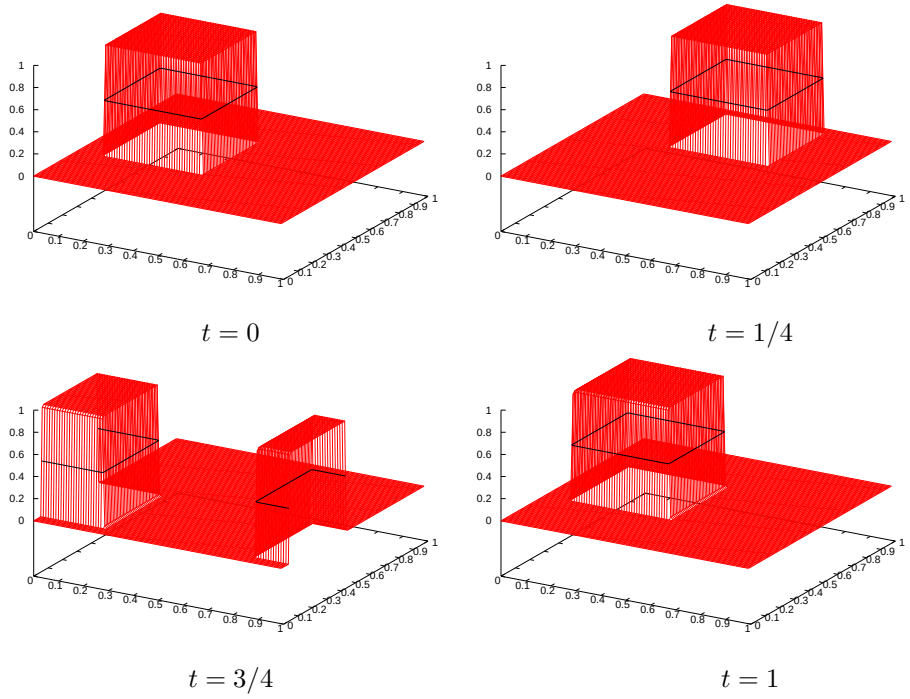


Figure 5: Exact transport of a square. The velocity is $u = v = 1$.

We also consider the same poblem in the rotating case: the velocity field is non constant in space and given by

$$u = -2\pi(y - 0.5), \quad v = 2\pi(x - 0.5).$$

Since the solver uses the alternate direction strategy, each x (resp. y) sweep is simple to code with a unique velocity on each horizontal (resp. vertical) line. The results are quite good and show almost no disspation. The final shape is correct.

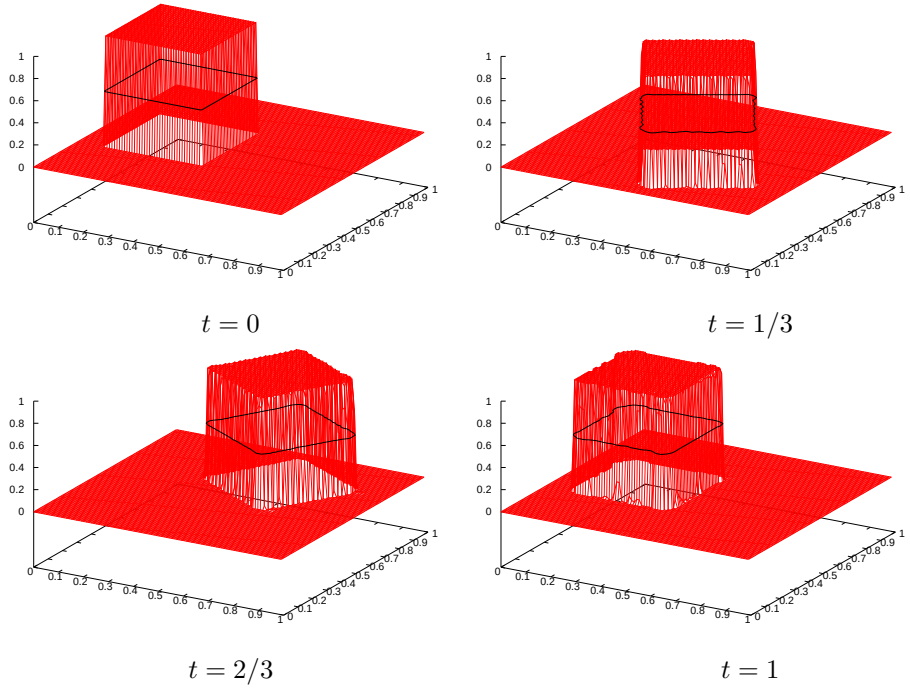
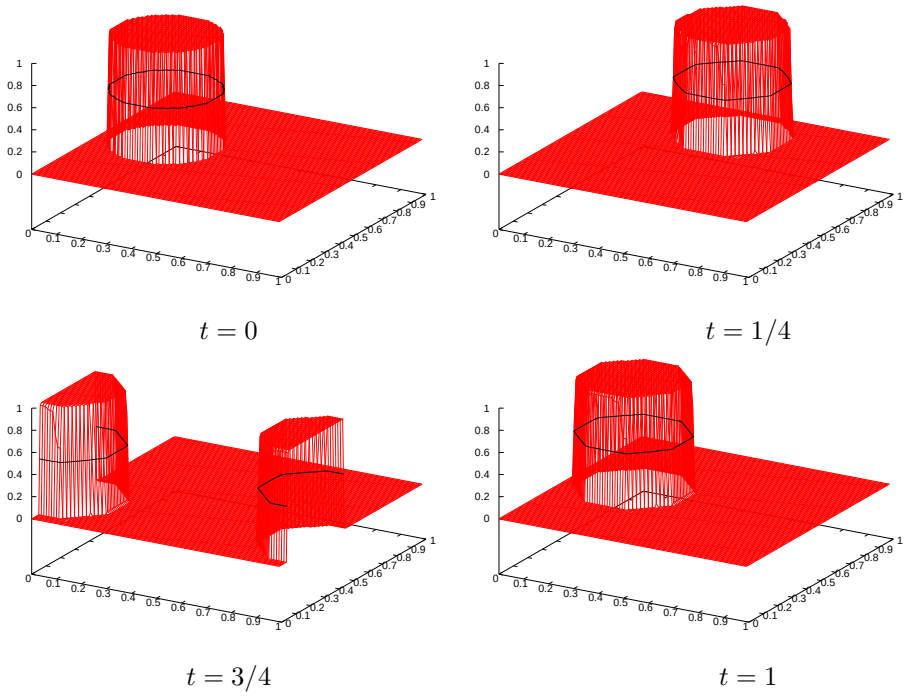


Figure 6: Transport of a square. The velocity is $u = -2\pi(y - 0.5)$ and $v = 2\pi(x - 0.5)$

3.3.2 Transport of a circle in dimension 2

The datas are the same as in the previous subsection, except that the initial data is the characteristic function $\mathbb{1}_D$ where $D = \{(x, y) \in \mathbb{R}^2 \text{ s.t. } \sqrt{(x - 0.25)^2 + (y - 0.25)^2} \leq 0.2\}$. The quality of the results is quite good. In particular the numerical transition from $c = 1$ to $c = 0$ is less than 2 cells everywhere at the boundary of the circle. For some computations, the final size of the circle is possibly closer to an octogone. This is quite similar with the results of [30], obtained with another anti-dissipative scheme. These computations give an indication that this very simple transport algorithm can be efficient also for non constant general velocity fields.



$t = 0$ $t = 1/4$

$t = 3/4$ $t = 1$

Figure 7: Transport of a circle. The velocity is $u = v = 1$.

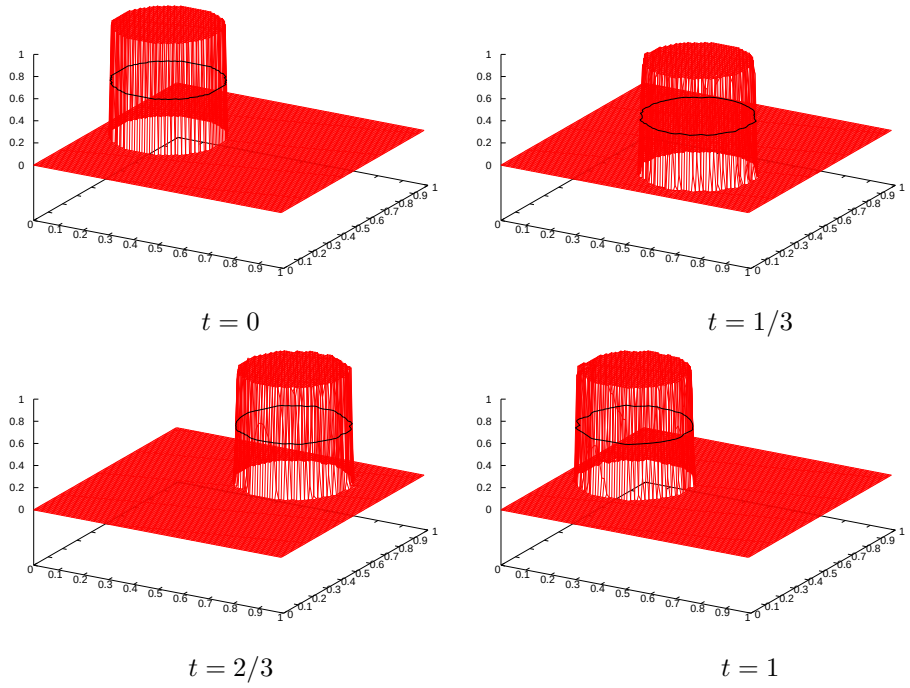


Figure 8: Transport of a circle. The velocity is $u = -2\pi(y - 0.5)$ and $v = 2\pi(x - 0.5)$.

3.3.3 Zalesak's test-case

We at last present result for the Zalesak test-case defined in [31]. Figure 9 shows the good behavior of the scheme for complex structures.

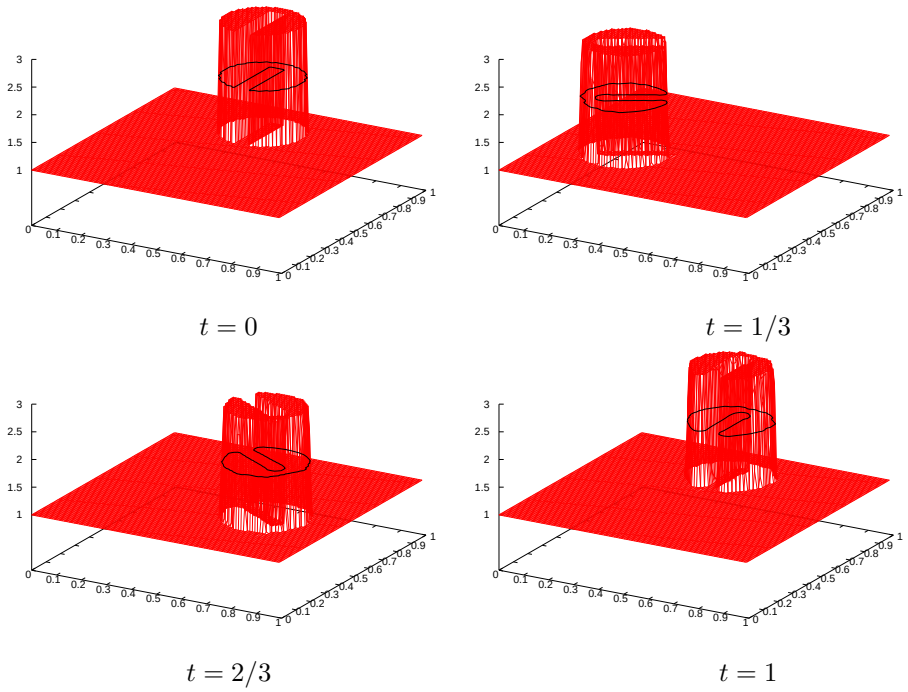


Figure 9: Zalesak's test-case. The velocity is $u = -2\pi(y - 0.5)$ and $v = 2\pi(x - 0.5)$.

4 The complete scheme

The complete scheme for the discretization of the two-components compressible model (9) in dimension one consists in a Lagrange-projection splitting. Dimensional splitting reduces the n-dimensional problem to a series of one-dimensional problems. Each one-dimensional problem is solved using a Lagrange-projection of the partial differential equations: see for example [15] or [8]. In dimensions two and three, the algorithm thus consists in a double splitting: a Lagrange-projection splitting and a dimensional splitting. The originality of the approach presented in this work is to use a limited downwind scheme in the re-mapping (projection) stage of the algorithm instead of the classical upwind projection. In what follows we write only the one-dimensional algorithm.

We first recall some basic facts about what is a Lagrange-projection splitting. As a simple example, let us consider the one-dimensional Euler system of compressible gas dynamics (with only one component). Is is straightforward to prove that the system on the left of (19) is, for smooth solutions with $\rho \neq 0$,

equivalent to the system on the right:

$$\left\{\begin{array}{l}\partial_t\rho+\partial_x(\rho u)=0,\\ \partial_t\rho u+\partial_x(\rho u^2+p)=0,\\ \partial_t\rho ue+\partial_x(\rho ue+pu)=0,\end{array}\right. \Longleftrightarrow \left\{\begin{array}{l}\rho D_t\tau-\partial_x u=0,\\ \rho D_t u+\partial_x p=0,\\ \rho D_t e+\partial_x pu=0,\end{array}\right. \tag{19}$$

where $D_t=\partial_t+u\partial_x$ is the convective derivative, and $\tau=1/\rho$ is the specific volume. The numerical discretization of the system on the right of (19) is the Lagrange part of the algorithm. Note that the Lagrange stage is a finite volume discretization of fluid dynamics in the comobile frame, that is in a reference frame which moves with the fluid. Then, of course, the re-mapping stage of the scheme is used to project the moving Lagrangian frame on the fixed Eulerian frame. The key point of the algorithm proposed here is more in the re-mapping stage, where the projection of the mass fractions uses a limited downwind procedure. It is important to notice that, even if neither the Lagrange step nor the projection one is conservative, the global algorithm is conservative. This is a very important property to get accurate numerical results for discontinuous solutions such as shocks and contact discontinuities.

4.1 The Lagrange computation

It corresponds to a finite volume numerical integration of the Lagrangian reformulation of (9) in the moving Lagrangian frame. We get, for the two models presented in section 2,

$$\left\{\begin{array}{l}\rho\mathrm{D}_t\tau-\partial_x u=0,\\ \rho\mathrm{D}_t c_1=0,\\ \rho\mathrm{D}_t c_2=0,\\ \rho\mathrm{D}_t u+\partial_x p=0,\\ \rho\mathrm{D}_t e+\partial_x pu=0,\\ p_1=p_2=p,\\ T_1=T_2,\end{array}\right. \quad\text{or}\quad \left\{\begin{array}{l}\rho\mathrm{D}_t\tau-\partial_x u=0,\\ \rho\mathrm{D}_t c_1=0,\\ \rho\mathrm{D}_t c_2=0,\\ \rho\mathrm{D}_t u+\partial_x p=0,\\ \rho\mathrm{D}_t e+\partial_x pu=0,\\ p_1=p_2=p,\\ \mathrm{D}_t\varepsilon_1+p_1\mathrm{D}_t\tau_1=\mathrm{D}_t\varepsilon_2+p_2\mathrm{D}_t\tau_2.\end{array}\right. \tag{20}$$

For every quantity z, let us denote $\widetilde{z}$ the same quantity after the discrete Lagrangian evolution (this will allow to avoid time indices n). The discretization then reads (forgetting for the moment the mixture part)

$$\left\{\begin{array}{l}\rho_j\dfrac{\widetilde{\tau}_j-\tau_j}{\Delta t}-\dfrac{u_{j+1/2}-u_{j-1/2}}{\Delta x}=0,\\ \widetilde{c_1}_j-{c_1}_j=0,,\\ \widetilde{c_2}_j-{c_2}_j=0,,\\ \rho_j\dfrac{\widetilde{u}_j-u_j}{\Delta t}+\dfrac{p_{j+1/2}-p_{j-1/2}}{\Delta x}=0,\\ \rho_j\dfrac{\widetilde{e}_j-e_j}{\Delta t}+\dfrac{p_{j+1/2}u_{j+1/2}-p_{j-1/2}u_{j-1/2}}{\Delta x}=0,\end{array}\right. \tag{21}$$

with $\tau_j=1/\rho_j$ for all cells (that is for all j). The fluxes in this stage are defined by the following formulae which are very close to some standard Lagrangian

Roe type fluxes

$$\left\{\begin{array}{l} \rho c^*_{j+1/2} = \sqrt{\max(\rho_j {c_j}^2, \rho_{j+1} {c_{j+1}}^2)\min(\rho_j, \rho_{j+1})}, \\ p_{j+1/2} = \dfrac{p_j + p_{j+1}}{2} + \dfrac{\rho c^*_{j+1/2}}{2}(u_j - u_{j+1}), \\ u_{j+1/2} = \dfrac{1}{2\rho c^*_{j+1/2}}(p_j - p_{j+1}) + \dfrac{1}{2}(u_j + u_{j+1}), \end{array}\right. \tag{22}$$

with $p_j = p(\rho_j, u_j, e_j)$ and $c_j = c(\rho_j, u_j, e_j)$ (c the sound speed of the global fluid). This can be understood as a kind of Roe scheme. The details of this Lagrange part is not the subject of the paper and we refer to [8] for a study of the entropy and positivity properties of the scheme. We recall the kind of entropy inequalities obtained in the appendix.

It remains to solve the mixture model. Since $(\widetilde{\tau}_j, \widetilde{u}_j, \widetilde{e}_j)$ have already been computed in (21), we first compute $\widetilde{\varepsilon}_j = e_j - \frac{1}{2}(\widetilde{u}_j)^2$, and then the following isobar-isothermal system composed of four equations:

$$\left\{\begin{array}{l} c_1\widetilde{\tau_1}_j + c_2\widetilde{\tau_2}_j = \widetilde{\tau}_j, \\ c_1\widetilde{\varepsilon_1}_j + c_2\widetilde{\varepsilon_2}_j = \widetilde{\varepsilon}_j, \\ p_1(\widetilde{\tau_1}_j, \widetilde{\varepsilon_1}_j) = p_2(\widetilde{\tau_2}_j, \widetilde{\varepsilon_2}_j), \\ T_1(\widetilde{\tau_1}_j, \widetilde{\varepsilon_1}_j) = T_2(\widetilde{\tau_2}_j, \widetilde{\varepsilon_2}_j). \end{array}\right. \tag{23}$$

This is a system of four equations with four unknowns: $(\widetilde{\tau_1}_j, \widetilde{\tau_2}_j, \widetilde{\varepsilon_1}_j, \widetilde{\varepsilon_2}_j)$. It is possible to prove that this system has a unique solution provided both EOS are thermodynamically consistent (cf. [22]). For perfect gas pressure laws, it is simple to get the exact solution (with explicit formulae). Indeed the isothermal equation simplifies in $\frac{\varepsilon_1}{c_{v1}} = \frac{\varepsilon_2}{c_{v2}}$, thus one gets after elimination in the second equation of the system (23) $\widetilde{\varepsilon_1} = \frac{\widetilde{\varepsilon}}{c_1 + c_2\frac{c_{v2}}{c_{v1}}}$ and $\widetilde{\varepsilon_2} = \frac{\widetilde{\varepsilon}}{c_1\frac{c_{v1}}{c_{v2}} + c_2}$. We have skip the index j which does not play any role. The isobar prescription gives $\widetilde{\tau_1} = \frac{(\gamma_1 - 1)\widetilde{\varepsilon_1}}{\widetilde{p}}$ and $\widetilde{\tau_2} = \frac{(\gamma_2 - 1)\widetilde{\varepsilon_2}}{\widetilde{p}}$. After elimination in the first equation of the system (23) one gets the value of the pressure $\widetilde{p} = \frac{c_1(\gamma_1 - 1)\widetilde{\varepsilon_1} + c_2(\gamma_2 - 1)\widetilde{\varepsilon_2}}{\widetilde{\tau}}$. It is then an easy matter to get $\widetilde{\tau_1}$ and $\widetilde{\tau_2}$.

We now present the discrete isobar-isoδQ model. Let us propose the following implicit one.

$$\left\{\begin{array}{l} c_1\widetilde{\tau_1}_j + c_2\widetilde{\tau_2}_j = \widetilde{\tau}_j, \\ c_1\widetilde{\varepsilon_1}_j + c_2\widetilde{\varepsilon_2}_j = \widetilde{\varepsilon}_j, \\ p_1(\widetilde{\tau_1}_j, \widetilde{\varepsilon_1}_j) = p_2(\widetilde{\tau_2}_j, \widetilde{\varepsilon_2}_j), \\ \left[\varepsilon_{1_j}\right] + p_1(\widetilde{\tau_1}_j, \widetilde{\varepsilon_1}_j)\left[\tau_{1_j}\right] = \left[\varepsilon_{2_j}\right] + p_2(\widetilde{\tau_2}_j, \widetilde{\varepsilon_2}_j)\left[\tau_{2_j}\right] \end{array}\right. \tag{24}$$

where $[z]$ stands for $\widetilde{z} - z$. An interesting theoretical stability result (via entropy inequalities) is stated in the appendix for the implicit system (24). Explicit discretization, is possible with the advantage of being cheaper from the computational point of view. However if one assumes perfect gas pressure laws, it is a simple exercise to get the exact analytical solution of (24). Indeed the procedure

is very close than for the isobar-isothermal model (23). The last equation of (24) may be recast as $\gamma_1\widetilde{\varepsilon_1}-\gamma_2\widetilde{\varepsilon_2}=\tau_1-\tau_2+\widetilde{p}(\tau_1-\tau_2)$. Using the last equation of the system (24) one can find an expression of $\widetilde{\varepsilon_1}$ and $\widetilde{\varepsilon_2}$ as an affine expression with respect to the pressure $\widetilde{p}$. Using once more the isobar prescription $\widetilde{\tau_1}=\frac{(\gamma_1-1)\widetilde{\varepsilon_1}}{\widetilde{p}}$ and $\widetilde{\tau_2}=\frac{(\gamma_2-1)\widetilde{\varepsilon_2}}{\widetilde{p}}$, one finds out that $\widetilde{\tau_1}$ and $\widetilde{\tau_2}$ are some rational expressions with respect to the pressure, the denominator and the numerator being first order polynomials. Using these expressions in the first equation of the system (24), one can compute the value of the pressure $\widetilde{p}$. Once the pressure has been computed, everything is straightforward.

4.2 The advection part

This part of the scheme, also called projection or re-mapping, is a numerical resolution of the convective derivative equation $\mathrm{D}_tU=0$ where U is the vector of all unknowns. It consists in the numerical approximation of $\partial_t\rho+u\partial_x\rho=0$, $\partial_tc_1+u\partial_xc_1=0$, $\partial_tc_2+u\partial_xc_2=0$, $\partial_tu+u\partial_xu=0$, $\partial_tv+u\partial_xv=0$ and $\partial_te+u\partial_xe=0$, with either the isobar-isothermal closure ($p_1=p_2=p$, $T_1=T_2$) or the isobar-isoδQ closure $p_1=p_2=p$ plus $\partial_t\varepsilon_1+p\partial_t\tau_1+u\partial_x\varepsilon_1+pu\partial_x\tau_1=\partial_t\varepsilon_2+p\partial_t\tau_2+u\partial_x\varepsilon_2+pu\partial_x\tau_2$. The discretization of these two systems will imply the non-classical limited downwind fluxes, but it is much easier to understand it via the classical upwind discretization: let us first make the use of these dissipative fluxes, assuming $u_{j+1/2}>0\quad\forall j\in\mathbb{Z}$, and denoting by $\widehat{z}$ the quantity z after the evolution due to this projection step.

Let us begin with the presentation of the standard **upwind discrete advection equations**. It reads

$$\left\{\begin{array}{l}
\dfrac{\widehat{\rho_j}-\widetilde{\rho_j}}{\Delta t}+u_{j-1/2}\dfrac{\widetilde{\rho_{j+1/2}}-\widetilde{\rho_{j-1/2}}}{\Delta x}=0,\\
\dfrac{\widehat{\rho_j}\widehat{c_{1,j}}-\widetilde{\rho_j}\widetilde{c_{1,j}}}{\Delta t}+u_{j-1/2}\dfrac{\widetilde{\rho_{j+1/2}}\widetilde{c_{1,j+1/2}}-\widetilde{\rho_{j-1/2}}\widetilde{c_{1,j-1/2}}}{\Delta x}=0,\\
\dfrac{\widehat{\rho_j}\widehat{c_{2,j}}-\widetilde{\rho_j}\widetilde{c_{2,j}}}{\Delta t}+u_{j-1/2}\dfrac{\widetilde{\rho_{j+1/2}}\widetilde{c_{2,j+1/2}}-\widetilde{\rho_{j-1/2}}\widetilde{c_{2,j-1/2}}}{\Delta x}=0,\\
\dfrac{\widehat{\rho_j}\widehat{u_j}-\widetilde{\rho_j}\widetilde{u_j}}{\Delta t}+u_{j-1/2}\dfrac{\widetilde{\rho_{j+1/2}}\widetilde{u_{j+1/2}}-\widetilde{\rho_{j-1/2}}\widetilde{u_{j-1/2}}}{\Delta x}=0,\\
\dfrac{\widehat{\rho_j}\widehat{e_j}-\widetilde{\rho_j}\widetilde{e_j}}{\Delta t}+u_{j-1/2}\dfrac{\widetilde{\rho_{j+1/2}}\widetilde{e_{j+1/2}}-\widetilde{\rho_{j-1/2}}\widetilde{e_{j-1/2}}}{\Delta x}=0
\end{array}\right.$$

where of course $\widetilde{\rho_j}=1/\widetilde{\tau_j}$ and with the upwind fluxes $\widetilde{\rho_{j+1/2}}=\widetilde{\rho_j}$, $\widetilde{c_{1,j+1/2}}=\widetilde{c_{1,j}}$, $\widetilde{c_{2,j+1/2}}=\widetilde{c_{2,j}}$, $\widetilde{u_{j+1/2}}=\widetilde{u_j}$ and $\widetilde{e_{j+1/2}}=\widetilde{e_j}$. The discretization of the mixture model will be presented after. First notice that this upwind discretization

combined with the Lagrange part (21) gives

$$\left\{\begin{array}{l}
\dfrac{\widehat{\rho_j}-\rho_j}{\Delta t}+\dfrac{\widetilde{\rho_{j+1/2}}u_{j+1/2}-\widetilde{\rho_{j-1/2}}u_{j-1/2}}{\Delta x}=0,\\
\dfrac{\widehat{\rho_j}\widehat{c_{1,j}}-\rho_j c_{1,j}}{\Delta t}+\dfrac{\widetilde{\rho_{j+1/2}c_{1,j+1/2}}u_{j+1/2}-\widetilde{\rho_{j-1/2}}\widetilde{c_{1,j-1/2}}u_{j-1/2}}{\Delta x}=0,\\
\dfrac{\widehat{\rho_j}\widehat{c_{2,j}}-\rho_j c_{2,j}}{\Delta t}+\dfrac{\widetilde{\rho_{j+1/2}}\widetilde{c_{2,j+1/2}}u_{j+1/2}-\widetilde{\rho_{j-1/2}}\widetilde{c_{2,j-1/2}}u_{j-1/2}}{\Delta x}=0,\\
\dfrac{\widehat{\rho_j}\widehat{u_j}-\rho_j u_j}{\Delta t}+\dfrac{\widetilde{\rho_{j+1/2}}\widetilde{u_{j+1/2}}u_{j+1/2}+p_{j+1/2}\cdots}{\Delta x}\\
\qquad\qquad\dfrac{\ldots-\widetilde{\rho_{j-1/2}}\widetilde{u_{j-1/2}}u_{j-1/2}-p_{j-1/2}}{\Delta x}=0,\\
\dfrac{\widehat{\rho_j}\widehat{e_j}-\rho_j e_j}{\Delta t}+\dfrac{\widetilde{\rho_{j+1/2}}\widetilde{e_{j+1/2}}u_{j+1/2}+p_{j+1/2}u_{j+1/2}\cdots}{\Delta x}\\
\qquad\qquad\dfrac{\ldots-\widetilde{\rho_{j-1/2}}\widetilde{e_{j-1/2}}-p_{j-1/2}u_{j-1/2}}{\Delta x}=0
\end{array}\right. \tag{25}$$

which shows that the global algorithm is consistent with Euler equations and conservative for all the natural conservative variables, even for the mass of each component. Thus

Lemma 2 The upwind scheme (25) is conservative for the mass of each component, for the total impulse and for the total energy.

Let us then present the **limited downwind extension of the upwind scheme (25)**. The observation is that (25) is obviously very dissipative, as any upwind scheme. In order to reduce the numerical dissipation we compute some limited downwind fluxes for ρ, c_1, c_2, u and e. The rule to construct all the required fluxes is the following : all thermodynamic variables such as ρ, e are upwinded, but the mass fractions c_1, c_2 are downwinded using an extension of the limited downwind scheme given in section 3. Since conservativity of the global scheme is a feature we want to preserve, we formulate these limited downwind fluxes in the conservative form of the global scheme (25).

Let us present the fluxes for the thermodynamic variables. In a second step we will give the anti-dissipative fluxes for mass fractions. As just explained, the expression of the fluxes for the global thermodynamic variables (ρ, u, v, e) requires the knowledge of the fluxes $\widetilde{c_{1\,j+1/2}}$ and $\widetilde{c_{2\,j+1/2}}$. It also requires the fluxes of thermodynamic quantities of each component: $\widetilde{\tau_{1,j+1/2}}$, $\widetilde{\tau_{2,j+1/2}}$, $\widetilde{\varepsilon_{1,j+1/2}}$, $\widetilde{\varepsilon_{2,j+1/2}}$. We simply take for these thermodynamic variables an upwind discretization:

- if $u_{j+1/2} \geq 0$:
  $\widetilde{\tau_{1,j+1/2}} = \widetilde{\tau_{1,j}}$, $\widetilde{\tau_{2,j+1/2}} = \widetilde{\tau_{2,j}}$, $\widetilde{\varepsilon_{1,j+1/2}} = \widetilde{\varepsilon_{1,j}}$ and $\widetilde{\varepsilon_{2,j+1/2}} = \widetilde{\varepsilon_{2,j}}$.
- if $u_{j+1/2} < 0$:
  $\widetilde{\tau_{1,j+1/2}} = \widetilde{\tau_{1,j+1}}$, $\widetilde{\tau_{2,j+1/2}} = \widetilde{\tau_{2,j+1}}$, $\widetilde{\varepsilon_{1,j+1/2}} = \widetilde{\varepsilon_{1,j+1}}$ and $\widetilde{\varepsilon_{2,j+1/2}} = \widetilde{\varepsilon_{2,j+1}}$.

We now can define fluxes for the global thermodynamic variables as follows: $\widetilde{f_{j+1/2}} = \widetilde{c_{1,j+1/2}}\widetilde{f_{1,j+1/2}} + \widetilde{c_{2,j+1/2}}\widetilde{f_{2,j+1/2}}$, for $f = \tau$ or ε. The flux for the velocity variable is also upwinded. This gives explicitly (26-27)

$$\textbf{if } u_{j+1/2} \geq 0 \begin{cases} \widetilde{\rho_{j+1/2}} = \dfrac{1}{\widetilde{\tau_{j+1/2}}} = \dfrac{1}{\widetilde{c_{1,j+1/2}}\widetilde{\tau_{1,j}} + \widetilde{c_{2,j+1/2}}\widetilde{\tau_{2,j}}}, \\ \widetilde{u_{j+1/2}} = \widetilde{u}_j, \\ \widetilde{e_{j+1/2}} = \widetilde{c_{1,j+1/2}}\widetilde{\varepsilon_{1,j}} + \widetilde{c_{2,j+1/2}}\widetilde{\varepsilon_{2,j}} + \dfrac{1}{2}\widetilde{u_j}^2; \end{cases} \tag{26}$$

$$\textbf{if } u_{j+1/2} < 0 \begin{cases} \widetilde{\rho_{j+1/2}} = \dfrac{1}{\widetilde{\tau_{j+1/2}}} = \dfrac{1}{\widetilde{c_{1,j+1/2}}\widetilde{\tau_{1,j+1}} + \widetilde{c_{2,j+1/2}}\widetilde{\tau_{2,j+1}}}, \\ \widetilde{u_{j+1/2}} = \widetilde{u_{j+1}}, \\ \widetilde{e_{j+1/2}} = \widetilde{c_{1,j+1/2}}\widetilde{\varepsilon_{1,j+1}} + \widetilde{c_{2,j+1/2}}\widetilde{\varepsilon_{2,j+1}} + \dfrac{1}{2}\widetilde{u_{j+1}}^2. \end{cases} \tag{27}$$

Equations (26) and (27) can be easily understood: since the exact solution verifies very similar relations, this is why we force them at the numerical level. We now give the numerical closures for the isobar-isothermal and for the isobar-isoδQ models.

For the isobar-isothermal model we get $\widehat{c_{1,j}}\widehat{\tau_{1,j}} + \widehat{c_{2,j}}\widehat{\tau_{2,j}} = \widehat{\tau_j}$, $\widehat{c_{1,j}}\widehat{\varepsilon_{1,j}} + \widehat{c_{2,j}}\widehat{\varepsilon_{2,j}} = \widehat{\varepsilon_j}$, $p_1(\widehat{\tau_{1,j}}, \widehat{\varepsilon_{1,j}}) = p_2(\widehat{\tau_{2,j}}, \widehat{\varepsilon_{2,j}})$ and $T_1(\widehat{\tau_{1,j}}, \widehat{\varepsilon_{1,j}}) = T_2(\widehat{\tau_{2,j}}, \widehat{\varepsilon_{2,j}})$. As in the Lagrange part, it can be shown that this system of four equations with four unknowns $(\widehat{\tau_{1,j}}, \widehat{\tau_{2,j}}, \widehat{\varepsilon_{1,j}}, \widehat{\varepsilon_{2,j}})$ has a solution under natural thermodynamic hypothesis (strict concavity of the physical entropies $S_i(\varepsilon_i, \tau_i)$, see [22]). In the case of 2 ideal gas, it is easy to get the explicit analytical expression of the solution.

For the isobar-isoδQ model we get the discrete system (28), where $[f] = \widehat{f} - \widetilde{f}$. The right-hand side term in last equation above in an approximation of $\Delta t(\rho u \partial_x(\varepsilon_2 - \varepsilon_1) + \rho u p \partial_x(\tau_2 - \tau_1))$. This last equation is therefore consistent with $\rho\partial_t(\varepsilon_1 - \varepsilon_2) + \rho p \partial_t(\tau_1 - \tau_2) + \rho u \partial_x(\varepsilon_1 - \varepsilon_2) + \rho u p \partial_x(\tau_1 - \tau_2) = 0$. Once more, it can be shown that in the case of two perfect gas, this system can be solved explicitly. We have already written it, the isobar-isoδQ model has the advantage of being numerically entropy-consistent through the consistent

discretization (28): see proposition 2.

$$\left\{\begin{array}{l}
\widehat{c_{1,j}}\widehat{\tau_{1,j}} + \widehat{c_{2,j}}\widehat{\tau_{2,j}} = \widehat{\tau_j},\\
\widehat{c_{1,j}}\widehat{\varepsilon_{1,j}} + \widehat{c_{2,j}}\widehat{\varepsilon_{2,j}} = \widehat{\varepsilon_j},\\
p_1(\widehat{\varepsilon_{1,j}}, \widehat{\tau_{1,j}}) = p_2(\widehat{\varepsilon_{2,j}}, \widehat{\tau_{2,j}}) = \widehat{p_j},\\
\widehat{\rho_j}([\varepsilon_{1,j}] + \widehat{p_j}[\tau_{1,j}] - [\varepsilon_{2,j}] - \widehat{p_j}[\tau_{2,j}])\\
\quad = \lambda(u_{j+1/2}\widetilde{\rho_{j+1/2}}(\dfrac{\widetilde{c_{1,j+1/2}}}{\widehat{c_{1,j}}}((\widetilde{\varepsilon_{1,j}} - \widetilde{\varepsilon_{1,j+1/2}}) + \widehat{p_j}(\widetilde{\tau_{1,j}} - \widetilde{\tau_{1,j+1/2}}))\\
\qquad\qquad - \dfrac{\widetilde{c_{2,j+1/2}}}{\widehat{c_{2,j}}}((\widetilde{\varepsilon_{2,j}} - \widetilde{\varepsilon_{2,j+1/2}}) + \widehat{p_j}(\widetilde{\tau_{2,j}} - \widetilde{\tau_{2,j+1/2}})))\\
\qquad - u_{j-1/2}\widetilde{\rho_{j-1/2}}(\dfrac{\widetilde{c_{1,j-1/2}}}{\widehat{c_{1,j}}}((\widetilde{\varepsilon_{1,j}} - \widetilde{\varepsilon_{1,j-1/2}}) + \widehat{p_j}(\widetilde{\tau_{1,j}} - \widetilde{\tau_{1,j-1/2}}))\\
\qquad\qquad - \dfrac{\widetilde{c_{2,j-1/2}}}{\widehat{c_{2,j}}}((\widetilde{\varepsilon_{2,j}} - \widetilde{\varepsilon_{2,j-1/2}}) + \widehat{p_j}(\widetilde{\tau_{2,j}} - \widetilde{\tau_{2,j-1/2}}))))
\end{array}\right. \tag{28}$$

where λ stands for $\Delta t/\Delta x$. All the algorithm for the projection part relies on the definition of the fluxes for mass fractions, and it is time now to describe these fluxes. We need some fluxes $\widetilde{c_{1,j+1/2}}$ and $\widetilde{c_{2,j+1/2}}$ for the numerical mass conservation equations

$$\left\{\begin{array}{l}
\widehat{\rho_j}\widehat{c_{1,j}} = \rho_j c_{1,j} - \lambda(u_{j+1/2}\widetilde{\rho_{j+1/2}}\widetilde{c_{1,j+1/2}} - u_{j-1/2}\widetilde{\rho_{j-1/2}}\widetilde{c_{1,j-1/2}}),\\
\widehat{\rho_j}\widehat{c_{2,j}} = \rho_j c_{2,j} - \lambda(u_{j+1/2}\widetilde{\rho_{j+1/2}}\widetilde{c_{2,j+1/2}} - u_{j-1/2}\widetilde{\rho_{j-1/2}}\widetilde{c_{2,j-1/2}}),
\end{array}\right.$$

with

$$\left\{\begin{array}{l}
\widehat{\rho_j} = \rho_j - \lambda(u_{j+1/2}\widetilde{\rho_{j+1/2}} - u_{j-1/2}\widetilde{\rho_{j-1/2}}),\\
\widetilde{\rho_{j+1/2}} = \dfrac{1}{\widetilde{c_{1,j+1/2}}\widetilde{\tau_{1,j}} + \widetilde{c_{2,j+1/2}}\widetilde{\tau_{2,j}}} \text{ if } u_{j+1/2} \geq 0,\\
\widetilde{\rho_{j+1/2}} = \dfrac{1}{\widetilde{c_{1,j+1/2}}\widetilde{\tau_{1,j+1}} + \widetilde{c_{2,j+1/2}}\widetilde{\tau_{2,j+1}}} \text{ if } u_{j+1/2} < 0.
\end{array}\right. \tag{29}$$

The choice we do is to use limited downwind fluxes as presented in section 3 for the advection equation. This leads once more to an L^∞-stable, TVD and non-dissipative scheme. Of course the definition of the fluxes is here more complex than in the pure advection case, in particular because the equation is here given for ρc_1 and ρc_2 and not c_1 and c_2. It turns out that it is not possible to do the complete description in these pages. Let us just insist on the fact that the principle is to take the most downwind possible value for $\widetilde{c_{1,j+1/2}}$ and $\widetilde{c_{2,j+1/2}}$ under the constraint that c_1 and c_2 both are L^∞ locally decreasing and TVD. A straightforward property is

Lemma 3 The limited downwind scheme (defined by (25) where each flux $\tilde{f}_{j+1}$ is replaced by $\tilde{f}_{j+\frac{1}{2}}$) is conservative for the mass of each component, for the total impulse and for the total energy.

A more complete presentation of this scheme together with some analysis of its properties can be found in [22], and a concise description of the algorithm is reported here in A. The following section is devoted to an important stability result which proves that the algorithm is free of spurious oscillations near contact discontinuities.

4.3 Control of spurious oscillations near contact discontinuities

It has been stressed in many places ([5] for example) that spurious oscillations may dramatically occur near contact discontinuities for multicomponent discrete algorithms. Many different methods have been derived to avoid these spurious oscillations: see [2], [26], [23], [11], [20], [21]. Up to our knowledge, such non-oscillatory schemes are not exactly conservative for each mass component, total momentum and total energy (they are at most quasi-conservative).

In this section we prove the scheme that is proposed in this work for the discretization of the isobar-isoδQ model is free of these spurious oscillations when dealing with two ideal gases. More precisely we prove that if a mixture of two different fluids (with different γ) is at rest from the mechanical point of view (that is the velocity and the pressure are constant), the scheme preserves this property. A complete presentation of this scheme together with some analysis of its properties can be found in [22], and a concise description of the algorithm is reported here in A. The following section is devoted to oscillations near contact discontinuities.

Theorem 3 Consider the Lagrange-projection scheme for the model with the the isobar-isoδQ closure (21, 22, 24, 28) with two ideal gases which pressure laws are $p_1 = (\gamma_1 - 1)\varepsilon_1/\tau_1$, $p_2 = (\gamma_2 - 1)\varepsilon_2/\tau_2$. Assume the mixture is at mechanical rest at the beginning of the time step, $u_j = u$ and $p_j = p\ \forall j$. Then the mixture is still at mechanical rest at the end of the time step, $\widehat{u_j} = u$ and $\widehat{p_j} = p\ \forall j$.

Proof Note that, from (21, 22), $\widetilde{u_j} = u$, and that from (25), $\widetilde{u_j} = u$, $\forall j \in \mathbb{Z}$. Thus, the difficulty is to prove that the pressure remains constant in space and in time.

We have to prove that pressure remains constant after the Lagrange step and also after the projection step. In each case the method of the proof is to show separately that there exists a discrete solution of the nonlinear discrete equations that preserves the pressure and then to prove that the discrete solution of the nonlinear discrete equations is unique.

Lagrange stage

After the Lagrange step, $(\widetilde{\tau}_{1,j}, \widetilde{\tau}_{2,j}, \widetilde{\varepsilon}_{1,j}, \widetilde{\varepsilon}_{2,j}) = (\tau_{1,j}, \tau_{2,j}, \varepsilon_{1,j}, \varepsilon_{2,j})$ is clearly a solution and since the solution is unique (see the discussion about how to solve (24) for perfect gas laws), then we deduce that nothing changes in the Lagrange step: we trivially have $\widetilde{p_{1,j}} = \widetilde{p_{2,j}} = \widetilde{p_j} = p$.

Projection stage

This stage is more complex and reveals the interest of the isobar-isoδQ model with discretization (28). Let us assume that $u > 0$. One can prove that

$$\begin{cases} \widehat{\rho_j}[\varepsilon_{1,_j}] = -\lambda u \widetilde{\rho_{j-1/2}} \dfrac{\widetilde{c_{1,j-1/2}}}{\widehat{c_{1,j}}} (\widetilde{\varepsilon_{1,_j}} - \widetilde{\varepsilon_{1,j-1}}), \\ \widehat{\rho_j}[\varepsilon_{2,_j}] = -\lambda u \widetilde{\rho_{j-1/2}} \dfrac{\widetilde{c_{2,j-1/2}}}{\widehat{c_{2,j}}} (\widetilde{\varepsilon_{2,_j}} - \widetilde{\varepsilon_{2,j-1}}), \\ \widehat{\rho_j}[\tau_{1,_j}] = -\lambda u \widetilde{\rho_{j-1/2}} \dfrac{\widetilde{c_{1,j-1/2}}}{\widehat{c_{1,j}}} (\widetilde{\tau_{1,_j}} - \widetilde{\tau_{1,j-1}}), \\ \widehat{\rho_j}[\tau_{2,_j}] = -\lambda u \widetilde{\rho_{j-1/2}} \dfrac{\widetilde{c_{2,j-1/2}}}{\widehat{c_{2,j}}} (\widetilde{\tau_{2,_j}} - \widetilde{\tau_{2,j-1}}) \end{cases} \tag{30}$$

is the solution of the discrete isobar-isoδQ system. Indeed, let us prove that (30) implies (28). Using the assumption $u > 0$, (30) implies that

$$\begin{cases} \widehat{c_{1,_j}}\widehat{\tau_{1,_j}} + \widehat{c_{2,_j}}\widehat{\tau_{2,_j}} = \widehat{\tau}_j, \\ \widehat{c_{1,_j}}\widehat{\varepsilon_{1,_j}} + \widehat{c_{2,_j}}\widehat{\varepsilon_{2,_j}} = \widehat{\varepsilon}_j, \\ p_1(\widehat{\varepsilon_{1,_j}}, \widehat{\tau_{1,_j}}) = p_2(\widehat{\varepsilon_{2,_j}}, \widehat{\tau_{2,_j}}) = \widehat{p}_j, \\ \widehat{\rho_j}([\varepsilon_{1,_j}] + \widehat{p_j}[\tau_{1,_j}] - [\varepsilon_{2,_j}] - \widehat{p_j}[\tau_{2,_j}]) \\ \quad = \lambda(-u\widetilde{\rho_{j-1/2}}(\dfrac{\widetilde{c_{1,j-1/2}}}{\widehat{c_{1,j}}}((\widetilde{\varepsilon_{1,_j}} - \widetilde{\varepsilon_{1,j-1}}) + \widehat{p_j}(\widetilde{\tau_{1,_j}} - \widetilde{\tau_{1,j-1}})) \\ \qquad\qquad - \dfrac{\widetilde{c_{2,j-1/2}}}{\widehat{c_{2,j}}}((\widetilde{\varepsilon_{2,_j}} - \widetilde{\varepsilon_{2,j-1}}) + \widehat{p_j}(\widetilde{\tau_{2,_j}} - \widetilde{\tau_{2,j-1}})))). \end{cases} \tag{31}$$

The last equation of (31) is a direct consequence of (30). Then, using the first two equations of (30), we get

$$\begin{aligned} \widehat{\rho_j}(\widehat{c_{1,_j}}\widehat{\varepsilon_{1,_j}} + \widehat{c_{2,_j}}\widehat{\varepsilon_{2,_j}}) &= \widehat{\rho_j}(\widehat{c_{1,_j}}\widetilde{\varepsilon_{1,_j}} + \widehat{c_{2,_j}}\widetilde{\varepsilon_{2,_j}}) \\ &\quad - \lambda u \widetilde{\rho_{j-1/2}}(\widetilde{c_{1,j-1/2}}(\widetilde{\varepsilon_{1,_j}} - \widetilde{\varepsilon_{1,j-1}}) + \widetilde{c_{2,j-1/2}}(\widetilde{\varepsilon_{2,_j}} - \widetilde{\varepsilon_{2,j-1}})). \end{aligned}$$

Then replacing $\widehat{\rho_j}\widehat{c_{1,_j}}$ and $\widehat{\rho_j}\widehat{c_{2,_j}}$ in the right-hand side by the expressions given by equations (25), we obtain after little simplifications,

$$\widehat{\rho_j}(\widehat{c_{1,_j}}\widehat{\varepsilon_{1,_j}} + \widehat{c_{2,_j}}\widehat{\varepsilon_{2,_j}}) = \widetilde{\rho_j}\widetilde{\varepsilon_j} - \lambda u(\widetilde{\rho_{j+1/2}}\widetilde{\varepsilon_{j+1/2}} - \widetilde{\rho_{j-1/2}}\widetilde{\varepsilon_{j-1/2}}).$$

Thus,

$$\begin{aligned} \widehat{\rho_j}(\widehat{c_{1,_j}}\widehat{\varepsilon_{1,_j}} + \widehat{c_{2,_j}}\widehat{\varepsilon_{2,_j}}) &= \widetilde{\rho_j}(\widetilde{\varepsilon_j} + \frac{u^2}{2}) - \widetilde{\rho_j}\frac{u^2}{2} \\ &\quad - \lambda u(\widetilde{\rho_{j+1/2}}(\widetilde{\varepsilon_{j+1/2}} + \frac{u^2}{2}) + pu - \widetilde{\rho_{j-1/2}}(\widetilde{\varepsilon_{j-1/2}} + \frac{u^2}{2}) - pu) \\ &\quad + \lambda u(\widetilde{\rho_{j+1/2}} - \widetilde{\rho_{j-1/2}})\frac{u^2}{2}, \end{aligned}$$

where the right-hand side is equal to

$$\widetilde{\rho_j}\widetilde{e_j} - \lambda u(\widetilde{\rho_{j+1/2}}\widetilde{e_{j+1/2}} + pu - \widetilde{\rho_{j-1/2}}\widetilde{e_{j-1/2}} - pu) - \widehat{\rho_j}\frac{u^2}{2}.$$

Recall that we have already proved that $\widehat{u_j} = u$. We then have

$$\widehat{\rho_j}(\widehat{c_{1,j}}\widehat{\varepsilon_{1,j}} + \widehat{c_{2,j}}\widehat{\varepsilon_{2,j}}) = \widehat{\rho_j}\widehat{e_j} - \widehat{\rho_j}\frac{u^2}{2} = \widehat{\rho_j}\widehat{\varepsilon_j}.$$

Similarly we show that $\widehat{c_{1,j}}\widehat{\tau_{1,j}} + \widehat{c_{2,j}}\widehat{\tau_{2,j}} = \widehat{\tau_j}$. Thus (30) is the solution of (28).

Let us finally show that (30) implies that mechanical rest is preserved. It remains only to check that $p_1(\widehat{\varepsilon_{1,j}}, \widehat{\tau_{1,j}}) = p_2(\widehat{\varepsilon_{2,j}}, \widehat{\tau_{2,j}}) = \widehat{p_j} = p$. For this, let us define

$$\beta_j = \lambda u \frac{\widetilde{\rho_{j-1/2}}\widetilde{c_{1,j-1/2}}}{\widehat{\rho_j}\widehat{c_{1,j}}}.$$

We see from (30) that $\widehat{\varepsilon_{1,j}} = (1-\beta_j)\widetilde{\varepsilon_{1,j}} + \beta_j\widetilde{\varepsilon_{1,j-1}}$ and $\widehat{\tau_{1,j}} = (1-\beta_j)\widetilde{\tau_{1,j}} + \beta_j\widetilde{\tau_{1,j-1}}$. Consequently $p_1(\widehat{\varepsilon_{1,j}}, \widehat{\tau_{1,j}}) = (\gamma_1 - 1)\frac{\widehat{\varepsilon_{1,j}}}{\widehat{\tau_{1,j}}} = (\gamma_1-1)\frac{(1-\beta_j)\widetilde{\varepsilon_{1,j}} + \beta_j\widetilde{\varepsilon_{1,j-1}}}{(1-\beta_j)\widetilde{\tau_{1,j}} + \beta_j\widetilde{\tau_{1,j-1}}}$. Now, recalling that $(\gamma_1 - 1)\widetilde{\varepsilon_{1,j}}/\widetilde{\tau_{1,j}} = p \ \forall j \in \mathbb{Z}$, we get

$$p_1(\widehat{\varepsilon_{1,j}}, \widehat{\tau_{1,j}}) = (\gamma_1 - 1)\frac{(1-\beta_j)\frac{p}{\gamma_1-1}\widetilde{\tau_{1,j}} + \beta_j\frac{p}{\gamma_1-1}\widetilde{\tau_{1,j-1}}}{(1-\beta_j)\widetilde{\tau_{1,j}} + \beta_j\widetilde{\tau_{1,j-1}}} = p.$$

The same computation gives also $p_2(\widehat{\varepsilon_{2,j}}, \widehat{\tau_{2,j}}) = p$. Thus the pressure is the same is all cells and is equal to the pressure at the beginning of the time step. One has the same property for the velocity.

Remark 1 Here is a remark about the singularity in formula (28). Indeed, note that the case $\widehat{c_{1j}} = 0$ (resp. $\widehat{c_{2j}} = 0$) is singular: the last formula in (28) is then meaningless. Actually, it is not a problem since the cell then is a pure cell with only one component and there is no need to compute τ_1, ε_1 (resp. τ_2, ε_2). One can also check that the right hand side of (28) takes reasonable values even for $\widehat{c_{1j}}$ small (resp. $\widehat{c_{2j}}$ small). Let us give a simple example using (31) which corresponds to $u > 0$. Assume for instance that $c_{1,j} = 0$ and $\widehat{c}_{1,j} \neq 0$ is small : in this case one has $\widehat{\rho}_j\widehat{c}_{1,j} = \lambda u\tilde{\rho}_{j-\frac{1}{2}}$. Then $\frac{\lambda u\tilde{\rho}_{j-\frac{1}{2}}}{\widehat{c}_{1,j}} = \widehat{\rho}_j$ and is non singular near $\widehat{c}_{1,j} \approx 0$, provided the density is non zero. All numerical experiments have shown that (28) is non singular and very robust.

A direct consequence of theorems 1, 2 and 3 is

Theorem 4 Consider, in dimension 1, 2 or 3 the Lagrange-projection scheme for the isobar-isoδQ model (21, 22, 24, 28) in each direction, for two ideal gases which pressure laws are given by $p_1 = (\gamma_1 - 1)\varepsilon_1/\tau_1$, $p_2 = (\gamma_2 - 1)\varepsilon_2/\tau_2$. Assume the mixture is at mechanical rest at the beginning of the time step, $u_j = u$ and $p_j = p \ \forall j$ and that the mass fraction c_1 is a staircase function in the sense of

theorem 1 or 2 (or its natural extension in dimension 3). Then the mixture is still at mechanical rest at the end of the time step with $\widehat{u_j} = u$ and $\widehat{p_j} = p$ $\forall$, and the mass fraction is still a staircase function.

5 Numerical results

This section is devoted to the presentation of some numerical results in dimensions one, two and three. The scheme used for these solutions is the one that is described before for the isobar-isoδQ model (21, 22, 24, 28) for two ideal gases. The 2- and 3-D codes are obtained with a directional splitting strategy that consists in solving alternately one-dimensional problems along directions x and y (and z in 3-D the test).

The 3-D system corresponding to (9) is

$$\begin{cases} \partial_t \rho + \mathrm{div}(\rho \vec{u}) = 0, \\ \partial_t (\rho c_1) + \mathrm{div}(\rho \vec{u} c_1) = 0, \\ \partial_t (\rho c_2) + \mathrm{div}(\rho \vec{u} c_2) = 0, \\ \partial_t (\rho \vec{u}) + \mathrm{div}(\rho \vec{u} \otimes \vec{u} + pI) = 0, \\ \partial_t (\rho e) + \mathrm{div}(\rho \vec{u} e + p\vec{u}) = 0, \\ c_1\tau_1 + c_2\tau_2 = \tau = 1/\rho, \\ e = c_1\varepsilon_1 + c_2\varepsilon_2 + \frac{1}{2}|u|^2 \\ p_1 = p_2 = p, \\ \textbf{isothermal } T_1 = T_2 \text{ or } \textbf{iso-}\delta Q \ \mathrm{D}_t\varepsilon_1 + p_1 \mathrm{D}_t\tau_1 = \mathrm{D}_t\varepsilon_2 + p_2 \mathrm{D}_t\tau_2 \end{cases}$$

where $\mathrm{D}_t = \partial_t + \vec{u} \cdot \vec{\nabla}$ and I is the identity matrix in dimension 3.

To assess the stability and consistency of the scheme, we first present the result of the computations of two different Riemann problems in dimension 1. Then we study hydrodynamic instabilities, the first one is the Kelvin-Helmholtz instability in dimension 2, the second one the Richtmyer-Meshkov instability in dimension 3. We will present in section 6 a result with dynamic mixing in dimension one.

5.1 Riemann problems in dimension 1

We first compute the solution of the standard Sod shock tube problem with two fluids having the same ideal pressure law, which is equivalent to a mono-fluid problem. The results are given in figure 12.

$0 \le x < 0.5$	$c_{1L} = 1$	$\rho_L = 1$	$p_L = 1$	$u_L = 0$	$\gamma_L = \gamma_1 = 1.4$
$0.5 \le x < 1$	$c_{1R} = 0$	$\rho_R = 0.125$	$p_R = 0.1$	$u_R = 0$	$\gamma_R = \gamma_2 = 1.4$

Table 1: Sod shock tube parameters

The parameters of the multimaterial Sod shock tube test case are nearly the same, except that the γ parameters of the gas laws are different. The results are given in figure 13.

$0 \leq x < 0.5$	$c_{1L} = 1.$	$\rho_L = 1.$	$p_L = 1.$	$u_L = 0$	$\gamma_L = \gamma_1 = 1.4$
$0.5 \leq x < 1$	$c_{1R} = 0.$	$\rho_R = 0.125$	$p_R = 0.1$	$u_R = 0$	$\gamma_R = \gamma_2 = 2$

Table 2: Multimaterial Sod shock tube parameters

First we give a few results computed with the **iso**p-**iso**T model for both cases (monomaterial and multimaterial). Figure 10 reports results for the velocity. We notice well-known fact that the velocity is not constant at the contact discontinuity. Moreover the **iso**p-**iso**T model needs a law to compute the temperature, that is to say, for ideal gases, coefficients c_{v1} and c_{v2} such that $T_1 = \varepsilon_1/c_{v1}$ and $T_1 = \varepsilon_2/c_{v2}$. These quantities are sometime quite difficult to know for "real-life" applications. This is a good reason to prefer the **iso**p-**iso**δQ model. Another reason is given by figure 11, where the internal energy ε is reported for the monomaterial and the multimaterial Sod test cases with different coefficients c_{v1} and c_{v2} and the limited downwind projection algorithm.

A pike at the contact discontinuity is present in both cases, but it is much smaller with the **iso**p-**iso**δQ. The reason is the artificial continuity of temperature that is forced by the **iso**p-**iso**T model.

These are the reasons why the following two- and three-dimensional computations are done with the **iso**p-**iso**δQ model.

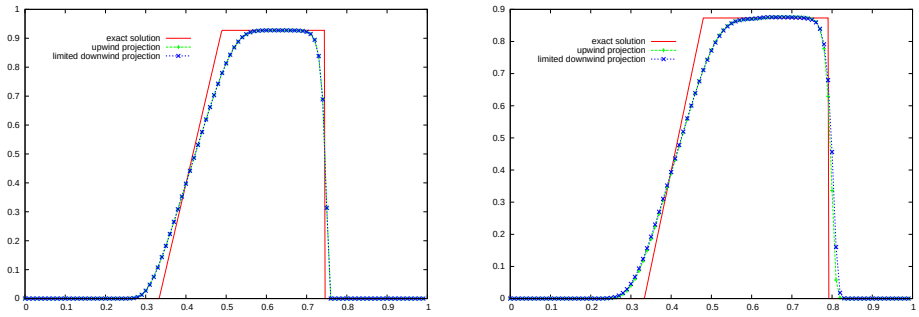


Figure 10: On the left: Sod shock tube computed with the **iso**p-**iso**T model. On the right: multimaterial Sod shock tube computed with the **iso**p-**iso**T model.

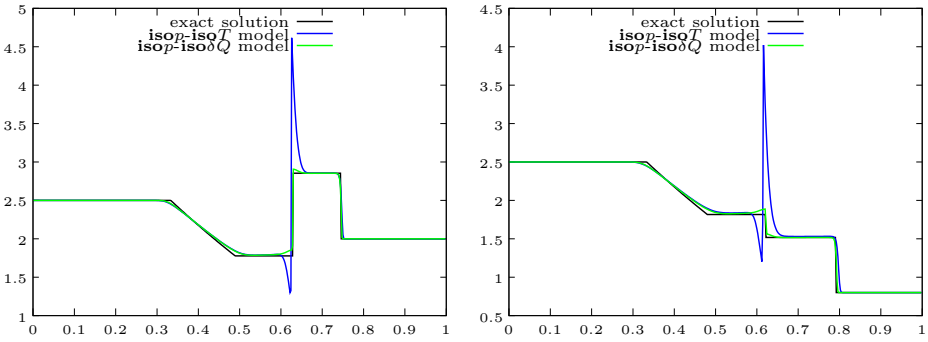


Figure 11: Sod shock tube with $\gamma_1 = \gamma_2 = 1.4$ on the left and $\gamma_1 = 1.4$, $\gamma_2 = 2$ on the right, and $c_{v1} = 1$ and $c_{v2} = 4$ on both. A huge pike is visible (with the **iso***p*-**iso***T* model). The height of the pike is a function of the ratio $\frac{c_{v1}}{c_{v2}}$. This pike is orders of magnitude greater than the similar discrepancy on the solutions computed with the **iso***p*-**iso**δQ model.

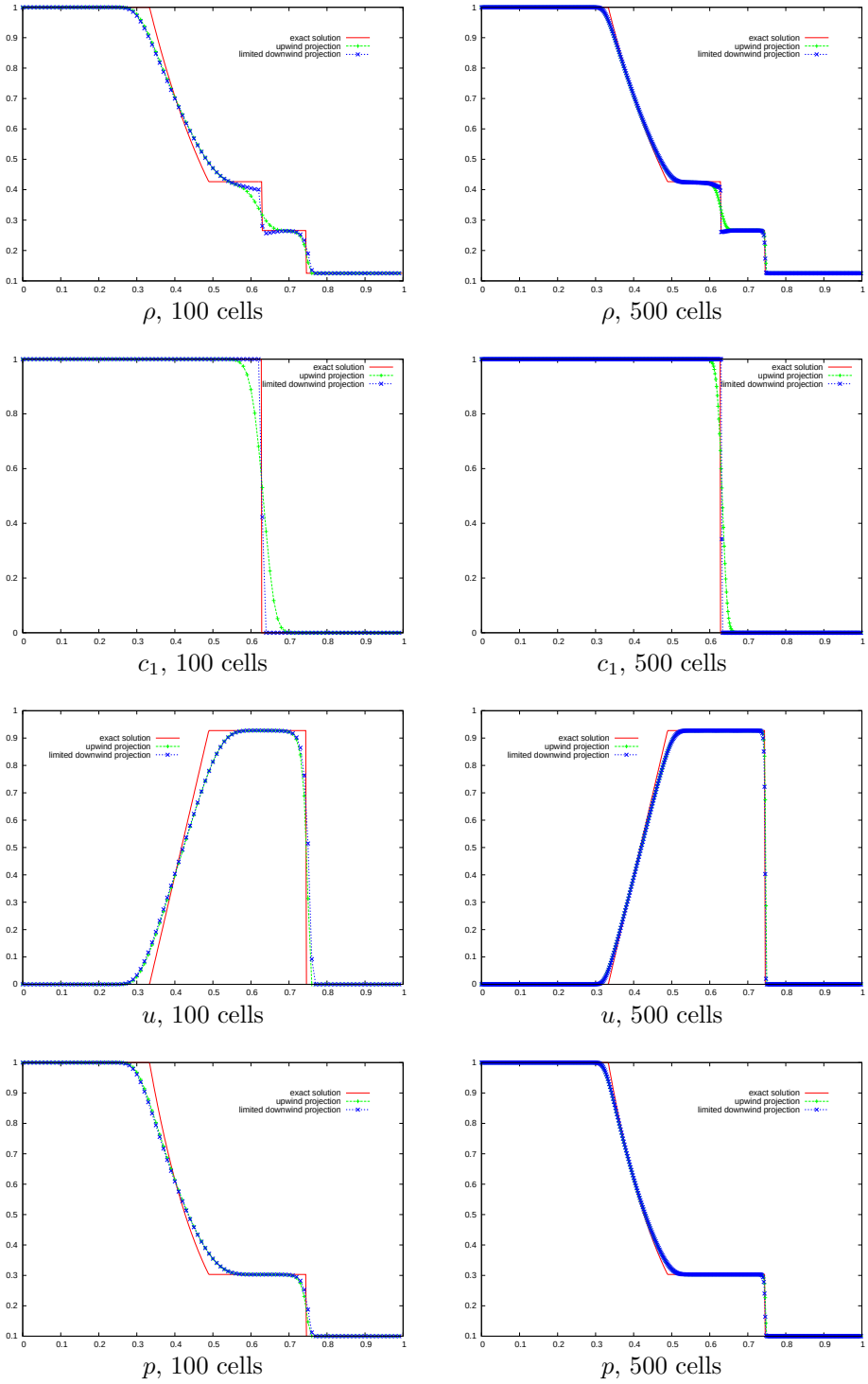


Figure 12: Sod shock tube. On the left (right) the numerical solution with 100 (500) cells for density, mass fraction, velocity and pressure. We have used the **iso**p-**iso**δQ model.

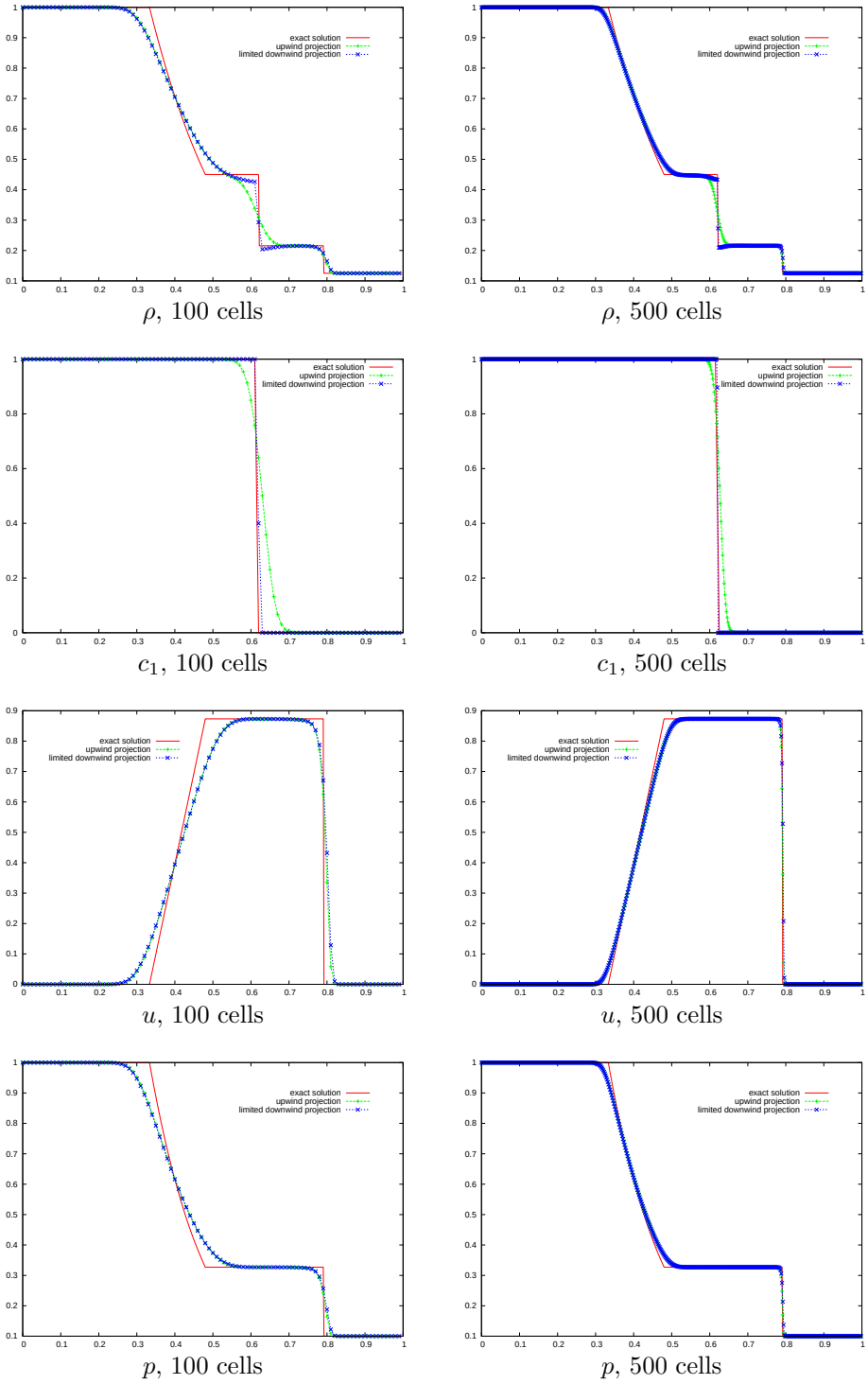
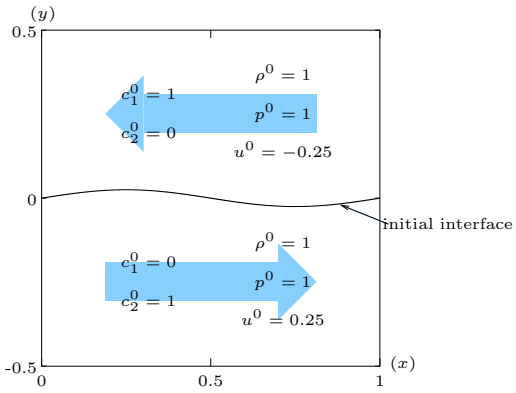


Figure 13: Multimaterial Sod shock tube: $\gamma_L = 1.4$, $\gamma_R = 2$. On the left (right) the solution with 100 (500) cells for density, mass fraction, velocity and pressure. We have used the **iso**p-**iso**δQ model. One can notice that the pressure and the velocity are constant in the area of the contact discontinuity.

5.2 A Kelvin-Helmholtz instability in dimension 2

The Kelvin-Helmholtz instability is a shear layer hydrodynamic instability.



Here the equation of the initial interface between the 2 fluids is $y = f(x) = 0,025\sin(2\pi x)$. The gas is the same on both parts of the interface with the same pressure law $\gamma_1 = \gamma_2 = \frac{5}{3}$. This interface is a contact discontinuity where the tangential part of the velocity is discontinuous. On the boundaries $x = 0$ and $x = 1$, we impose periodic conditions (simulating an infinite sinusoid) and some wall-condition on $y = 0$ and $y = 1$. On the left is the result obtained with a classical upwind discretization of the projection stage. On the right is the result obtained limited downwind re-mapping stage. For $t > 0$ the interface winds and becomes a kind of spiral.

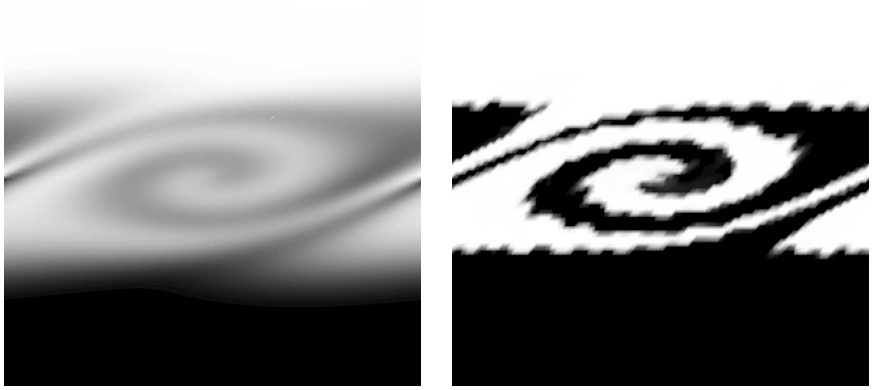


Figure 14: Kelvin-Helmholtz instability, 10,000 cells.

The results have been computed with 100×100 cells (figure 14), 500×500 cells (figure 15), and, finally, with 1000×1000 cells (figure 16), for time $t = 8$.

We can compare solutions computed with the classical upwind scheme and the limited downwind scheme (both used for the projection of mass fractions). We observe that the scheme with limited downwind fluxes for the mass fractions does not introduce any spreading of the interface. This allows to follow with an increased accuracy the interfacial instability. The presence of steps along the interface is a small drawback in this case: indeed, the results show that the length of each step seems to converge towards 0 when refining. These results are qualitatively comparable to those obtained with front tracking methods.

Figure 15: Kelvin-Helmholtz instability, 250,000 cells.

Figure 16: Kelvin-Helmholtz instability, 1,000,000 cells.

5.3 A Richtmyer-Meshkov instability in dimension 3

This instability appears when a shock impinges an interface between 2 components, coming from the light fluid and going into the heavy fluid. We here choose initial conditions of the test-case proposed by [13] (this is the so called "Stony Brook" test case). The interface equation is

$$f(x,y) = 0.005\cos(2\pi\sqrt{x^2+y^2}/0,036) + 0.12.$$

The gas is the same on both parts of the interface with the same pressure law $\gamma_1 = \gamma_2 = \frac{5}{3}$. The computational domain is $(x,y,z) \in [0,0.018] \times [0,0.018] \times [0,0.06]$ with wall-conditions on $x = 0$, $x = 0.018$, $y = 0$, $y = 0.018$ and "neutral" condition on $z = 0$ and $z = 0.06$. Initial values are (the velocity components being denoted as u, v, w)

$z > f(x,y)$	$c_1 = 1$	$\rho = 2.95$	$p = 50000$	$u = v = 0$	$w = -453,$
$0.09 < z < f(x,y)$	$c_1 = 0$	$\rho = 1.87$	$p = 50000$	$u = v = 0$	$w = -453,$
$z < 0.09$	$c_1 = 0$	$\rho = 6.01$	$p = 753000$	$u = v = 0$	$w = 55.5.$

The equations in dimension 3 are straightforward from the system written in dimension 2. Since the discontinuity at $z = 0.09$ is a pure shock, the solution is such that a pure shock arrives on the interface. After this, a typical "mushroom" instability is observed: this is the Richtmyer-Meshkov instability. Let us observe on figures 17 and 18 the mass fraction c_1 at $t = 0.00105$, computed with the scheme proposed in this work. The sharpness of the interface is almost perfect with the limited downwind projection, and the difference with the classical scheme is remarkable. The value of c_1 is showed on $x = 0$ and $y = 0$: in order to show the anti-dissipativity of the presented scheme, we represent on figure 17, the regions where $0 \leq c_1 \leq 0.1$ (black), $0.1 \leq c_1 \leq 0.9$ (grey) and $0.9 \leq c_1 \leq 1$ (white). The top of the figure shows the result of the upwind remapping stage, the mass fraction is spread over a large domain; this is not the case with the limited downwind scheme (bottom of the figure). Figure 18 shows the same result but with mass fraction truncated at $c_1 = 0.5$: $0 \leq c_1 \leq 0.5$ (black), $0.5 \leq c_1 \leq 1$ (white). The result with the limited downwind scheme is almost the same than in figure 17, showing that the mass fraction takes values 0 and 1 exepted in very few cells along the interface. Moreover, the isosurface $c_1 = 0.5$ shows a lot more details than with the upwind projection stage.

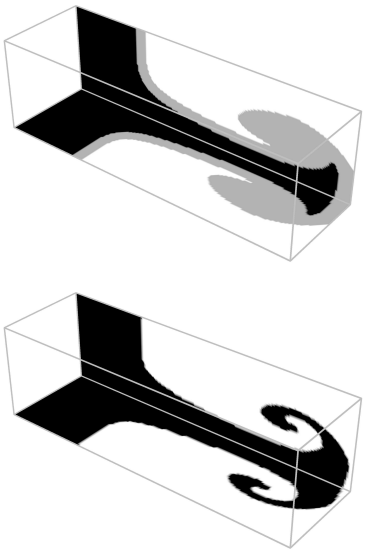


Figure 17: Richtmyer-Meshkov instability with the upwind scheme (top) and limited downwind scheme (bottom) in the remapping stage, $65 \times 65 \times 216$ cells.

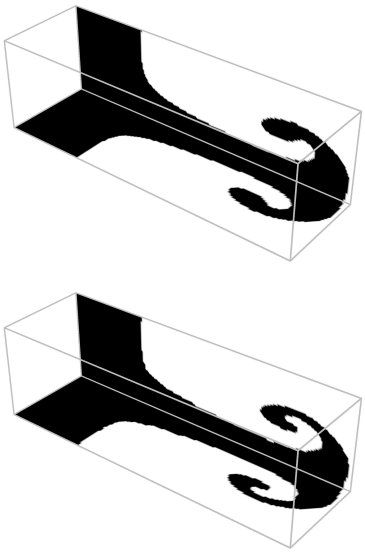


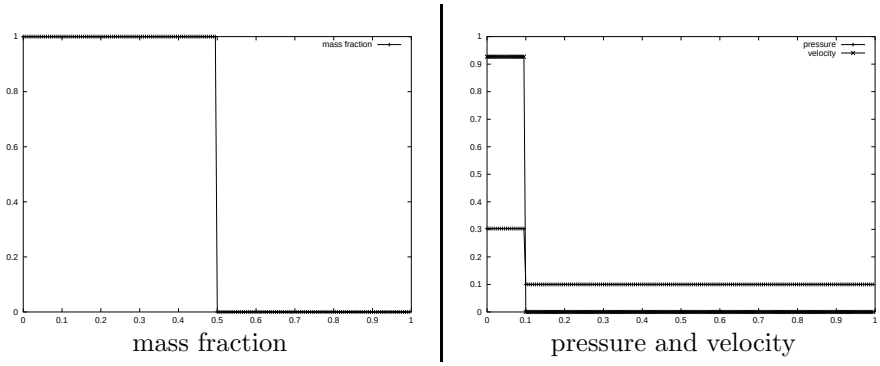
Figure 18: Richtmyer-Meshkov instability with the upwind scheme (top) and limited downwind scheme (bottom) in the remapping stage, $65 \times 65 \times 216$ cells.

6 Dynamic mixing in dimension 1

This section is devoted to present a very simple extension of the model and the algorithm to **dynamic mixing**. Dynamic mixing is here understood as any situation where the underlying physics tends to replace the interface with a diffuse interface. The mechanism responsible for this phenomena may be transition to turbulence. The model is (32)

$$\begin{cases} \partial_t \rho + \partial_x(\rho u) = 0, \\ \partial_t(\rho c_1) + \partial_x(\rho u c_1) = \partial_x(K \partial_x c_1), \\ \partial_t(\rho c_2) + \partial_x(\rho u c_2) = \partial_x(K \partial_x c_2), \\ \partial_t(\rho u) + \partial_x(\rho u^2 + p) = 0, \\ \partial_t(\rho e) + \partial_x(\rho u e + p u) = 0, \\ p_1 = p_2 = p, \\ T_1 = T_2. \end{cases} \tag{32}$$

For plasma computations, the $T_1 = T_2$ model is convenient. We add a Fick diffusion right-hand side in (9) in order to model the transition from an interface at $t = 0$ to a mixing zone with compact support. In contrary to the results of section 5, there is no mixing zone at $t = 0$, but there is one for some time $T > 0$, which is characteristic of dynamic mixing. The model (32) that we study is an extension of the isobar-isothermal one-dimensional model where the diffusion coefficient is a non linear functional with respect to the mass fraction $K = a(c_1 c_2)^m$ ($a > 0$ and $m > 0$). Numerical results for a simple experiment are given in figures 21 to 23. The initial condition is given in figure 19. The numerical results were obtained with a splitting strategy: first we solve the homogeneous hyperbolic left-hand side of system (32) with the Lagrange-projection anti-dissipative algorithm described in previous sections, second we solve an implicit discretization of the non linear diffusion equation with an explicit value of the diffusion coefficient.



mass fraction | pressure and velocity

Figure 19: $t = 0$: the interface ($x = 0.5$) is ahead of the shock ($x = 0.1$)

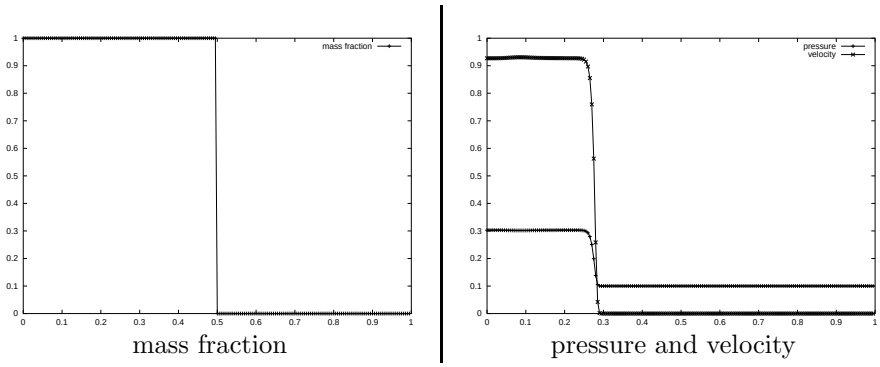


Figure 20: The interface ($x = 0.5$) is still ahead of the shock ($x \approx 0.3$)

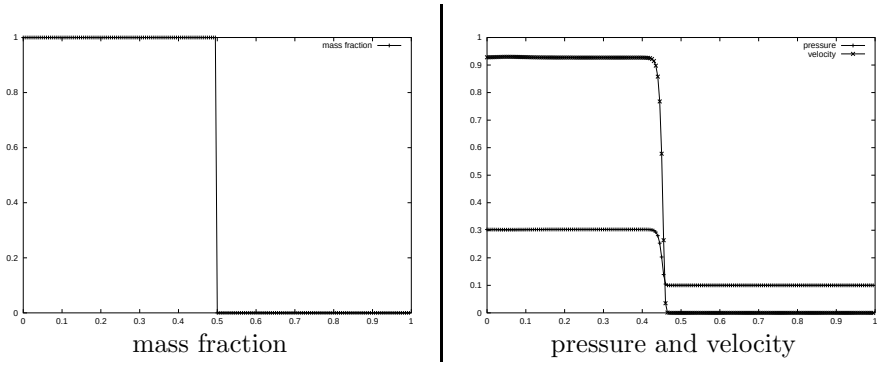


Figure 21: $t = 0.2$: the interface ($x = 0.5$) is just ahead of the shock ($x \approx 0.47$)

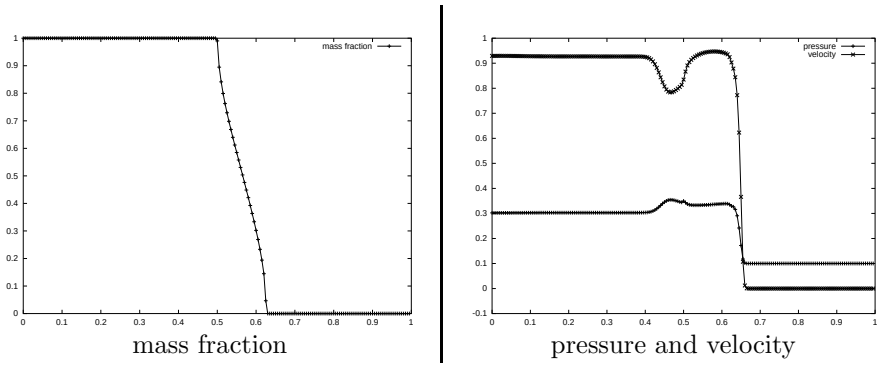


Figure 22: $t = 0.3$: the mixing zone $0.51 \leq x \leq 0.61$ has been dynamically created by the shock

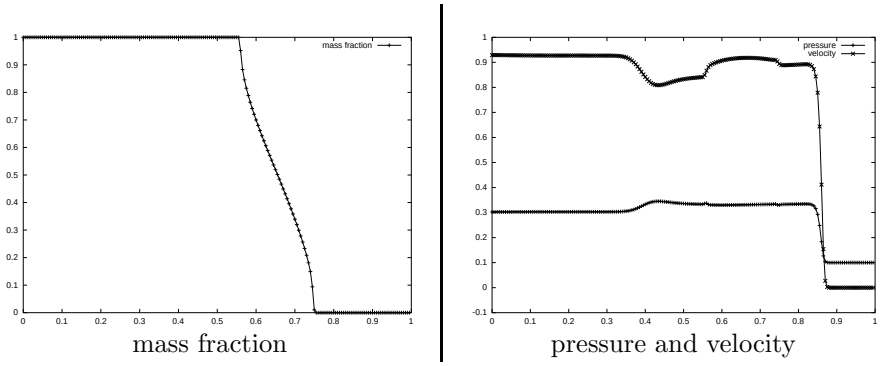


Figure 23: $t = 0.4$: the shock is now away from the mixing zone

Let us comment briefly the results: $a = 0.1$, $m = 2$, $CFL = 0.6$. At $t = 0$, figure 19, the interface is at $x = 0.5$. We use different pressure laws on both side of the interface $\gamma_1 = 1.4$ and $\gamma_2 = 2$. Then a shock comes on the interface. As long as the shock does not interact with the interface, the numerical value of the diffusion is $K = a(c_1 c_2)^m = 0$ in the neighborhood of the interface, thus the diffusion operator does not play any role in the simulation: indeed until figure 20, we see that the problem reduces to a pure hydrodynamic computation. But in figure 21 the shock is on the interface. Now due to the fact that the velocity is non-zero after the shock, both mass fractions c_1 and c_2 are transported on the mesh. So these mass fractions take intermediate values in at least one cell. In these cells one has $c_1 c_2 > 0$ so the coefficient in the diffusion operator is positive: it leads to non linear diffusion effect which is, in this simulation, the cause of the dynamic mixing. This is illustrated in figures 22 and 23. This scenario is typical of a dynamical mixing. One sees no staircase profile in figures 22 and

23 : the reason is that we solve at each time step a diffusion operator which is non degenerate inside the dynamic mixing zone. The Muscl limiter coupled with the limited downwind flux introduced in previous section is probably useless in this case. The limited downwind flux is enough.

This kind of numerical results are heavily dependent on some mathematical properties of the non-linear degenerate diffusion operator which is on the right-hand side of (32). It is indeed the fact that the same Cauchy problem for some given non linear degenerate diffusion heat equation has two different solutions which render possible this kind of modeling of dynamic mixing.

A Non-dissipative algorithm for the mass fractions

The method for deriving the present scheme is rigorously the same as the one reported in [9] for linear advection equation and for Euler equation. The principle is to write some stability (L^∞ and TVD) constraints on the mass fractions c_1 and c_2 and to deduce from it some sufficient conditions on each mass fraction flux. Avoiding explaining the algorithm in detail (which is done in [22]), let us just briefly describe it, first concentrating on the computation of fluxes for c_1.

Let us define the following quantities $M_{j+1/2} = \max(c_{1,j}, c_{1,j+1})$, $m_{j+1/2} = \min(c_{1,j}, c_{1,j+1})$. Then

$$\left\{\begin{array}{l} \widetilde{s_{j+1/2}} = \lambda u_{j+1/2} + \rho_j(\widetilde{M_{j-1/2}} - \widetilde{c_{1,j}})(\widetilde{\tau_{1,j}} - \widetilde{\tau_{2,j}}), \\ \widetilde{t_{j+1/2}} = \rho_j(\widetilde{c_{1,j}} - \widetilde{M_{j-1/2}})\widetilde{\tau_{2,j}} + \widetilde{M_{j-1/2}}\lambda u_{j+1/2}, \\ \widetilde{v_{j+1/2}} = \lambda u_{j+1/2} + \rho_j(\widetilde{m_{j-1/2}} - \widetilde{c_{1,j}})(\widetilde{\tau_{1,j}} - \widetilde{\tau_{2,j}}), \\ \widetilde{w_{j+1/2}} = \rho_j(\widetilde{c_{1,j}} - \widetilde{m_{j-1/2}})\widetilde{\tau_{2,j}} + \widetilde{m_{j-1/2}}\lambda u_{j+1/2}, \end{array}\right.$$

In order the scheme (25) to be stable for mass fractions, it is sufficient to have

$$\widetilde{c_{1,j+1/2}}\widetilde{s_{j+1/2}} \geq \widetilde{t_{j+1/2}},\ \widetilde{c_{1,j+1/2}}\widetilde{v_{j+1/2}} \leq \widetilde{w_{j+1/2}},\ \widetilde{m_{j+1/2}} \leq \widetilde{c_{1,j+1/2}} \leq \widetilde{M_{j+1/2}}.$$

Thus explicit form of the fluxes $c_{1j+1/2}$ and $c_{2j+1/2}$ depends on the signs of $\widetilde{s_{j+1/2}}$ and $\widetilde{v_{j+1/2}}$. First remark that because of their definition, these two numbers cannot be simultaneously negative.

- If $\widetilde{s_{j+1/2}} > 0$ and $\widetilde{v_{j+1/2}} > 0$, define $\widetilde{\gamma_{j+1/2}} = \max(\widetilde{t_{j+1/2}}/\widetilde{s_{j+1/2}}, \widetilde{m_{j+1/2}})$ and $\widetilde{\Gamma_{j+1/2}} = \min(\widetilde{w_{j+1/2}}/\widetilde{v_{j+1/2}}, \widetilde{M_{j+1/2}})$.
- if $\widetilde{s_{j+1/2}} > 0$ and $\widetilde{v_{j+1/2}} < 0$, we define $\widetilde{\gamma_{j+1/2}} = \max(\widetilde{t_{j+1/2}}/\widetilde{s_{j+1/2}}, \widetilde{w_{j+1/2}}/\widetilde{v_{j+1/2}}, \widetilde{m_{j+1/2}})$ and $\widetilde{\Gamma_{j+1/2}} = \widetilde{M_{j+1/2}}$.
- if $\widetilde{s_{j+1/2}} < 0$, $\widetilde{v_{j+1/2}} > 0$ and we define $\widetilde{\gamma_{j+1/2}} = \widetilde{m_{j+1/2}}$ and $\widetilde{\Gamma_{j+1/2}} = \min(\widetilde{t_{j+1/2}}/\widetilde{s_{j+1/2}}, \widetilde{w_{j+1/2}}/\widetilde{v_{j+1/2}}, \widetilde{M_{j+1/2}})$.

- if $\widetilde{s_{j+1/2}} = 0$ (resp. $\widetilde{v_{j+1/2}} = 0$), we define $\widetilde{\gamma_{j+1/2}} = \widetilde{m_{j+1/2}}$ and $\widetilde{\Gamma_{j+1/2}} = \widetilde{M_{j+1/2}}$.

Then the limited downwind flux is defined as

$$\begin{array}{ll} \widetilde{c_{1,j+1/2}} = \widetilde{\gamma_{j+1/2}} & \text{if } \widetilde{c_{1,j+1}} \leq \widetilde{\gamma_{j+1/2}}, \\ \widetilde{c_{1,j+1/2}} = \widetilde{c_{1,j+1}} & \text{if } \widetilde{\gamma_{j+1/2}} \leq \widetilde{c_{1,j+1}} \leq \widetilde{\Gamma_{j+1/2}}, \\ \widetilde{c_{1,j+1/2}} = \widetilde{\Gamma_{j+1/2}} & \text{if } \widetilde{\Gamma_{j+1/2}} \leq \widetilde{c_{1,j+1}}. \end{array}$$

Then, the fluxes for c_2 can be either computed with the same formulae or (it is shown in [22] that it is equivalent) by $c_{2,j+1/2} = 1 - c_{1,j+1/2}$.

B Entropy properties

For the sake of completness of this presentation we state some general stability properties of the scheme that can be proved but only with the isobar-isoδQ model. One has two numerical entropy inequalities for the scheme used for the isobar-isoδQ model: one for the Lagrange part and one for the projection part. We refer to [7] for a complete and quite lengthy proof of these results.

Proposition 1 Entropy property in the Lagrange part Let us consider the Lagrangian scheme (21-24). Let $S_1(\varepsilon_1, \tau_1)$ and $S_2(\varepsilon_2, \tau_2)$ be concave entropies for fluids 1 and 2. Then there exist $c \in \mathbb{R}$ such that under the CFL condition $c\Delta t/\Delta x \leq 1$,

$$S_1(\widetilde{\varepsilon_{1,j}}, \widetilde{\tau_{1,j}}) \geq S_1(\varepsilon_{1,j}, \tau_{1,j}) \text{ and } S_2(\widetilde{\varepsilon_{2,j}}, \widetilde{\tau_{2,j}}) \geq S_2(\varepsilon_{2,j}, \tau_{2,j}) \quad \forall j \in \mathbb{Z}. \quad (33)$$

The number c is an approximate value of the maximum of the (mixture) sound speed in all the cells. Inequality (33) simply explains that the scheme is entropy consistent in the Lagrange part. The following proposition 2 now states an entropy result for the scheme independent on the mass fraction fluxes $\widetilde{c_{1,j+1/2}}$, $\widetilde{c_{2,j+1/2}}$.

Proposition 2 Entropy property in the re-mapping part Assume that some very natural conditions are verified: positivity of temperatures, mass fractions, densities. The positivity of temperature is simply $T_1(\widehat{\varepsilon_{1,j}}, \widehat{\tau_{1,j}}) > 0$ and $T_2(\widehat{\varepsilon_{2,j}}, \widehat{\tau_{2,j}}) > 0$. We refer to [22] for the precise statement of all other positive inequalities. Then we obtain entropy inequalities for each entropy in the re-mapping part of the algorithm, in the sense that for every $j \in \mathbb{Z}$ ($r = 1, 2$)

$$S_r(\widehat{\varepsilon_{r,j}}, \widehat{\tau_{r,j}}) \geq \min\left(S_r(\widetilde{\varepsilon_{r,j}}, \widetilde{\tau_{r,j}}), S_r(\widetilde{\varepsilon_{r,j-1}}, \widetilde{\tau_{r,j-1}}), S_r(\widetilde{\varepsilon_{r,j+1}}, \widetilde{\tau_{r,j+1}})\right).$$

This proves that the entropy in each cell after the projection part is greater than the entropy before. A more precise statement of this property is in [22]. A very remarkable feature of these inequalities is that there are true whatever the fluxes

of the mass fractions are, the only requirement being that all other fluxes are compatible with the fluxes of the mass fractions (26-27). So it explains that the thermodynamic stability of the scheme is independent on the scheme used for the mass fractions and justify the use of a highly anti-dissipative down-winded scheme for the mass fractions in conjunction with an up-winded scheme for all thermodynamic variables.

Acknowledgements This research was done when both authors were at the Commissariat à l'Énergie Atomique, France. Both authors are greatly indebted to H. Jourdren for his presentation of multi-component compressible fluid dynamics simulation and thank A. Davroux for the numerical simulations of the coupled algorithm for mixing zones, and L. Pierrejean for the numerical study of the dynamical mixing zone.

References

[1] R. Abgrall, Généralisation du schéma de Roe pour le calcul d'écoulements de mélanges de gaz à concentrations variables, La recherche aérospatiale, 1989.

[2] R. Abgrall, How to prevent pressure oscillations in multicomponent flow calculations: a quasi-conservative approach, J. Comp. Phys., 125: 150–160, 1996.

[3] G. Allaire, S. Clerc and S. Kokh, A five-equation model for the numerical simulation of interfaces in two-phase flows, Comptes Rendus de l'Académie des Sciences, 2000.

[4] F. Bouchut, An antidiffusive entropy scheme for monotone scalar conservation laws, 21, JSC, 2004, pp 1–30.

[5] D. Chargy, R. Abgrall, L. Fezoui and B. Larrouturou, Comparisons of several upwind schemes for multi-component one-dimensional inviscid flows, INRIA, 1990.

[6] P. Colella, Volume-of-Fluid methods for partial differential equations, Godunov methods, published by E. Toro, Kluwer Academic/Plenum Publishers, 2001.

[7] B. Després, Construction, analyse et discrétisation d'un modèle de dynamique des fluides compressibles multi-constituants, Technical report CEA, 1997.

[8] B. Després, Inégalités entropiques pour un solveur de type Lagrange + projection des équations de l'hydrodynamique, Technical report CEA, 1997.

[9] B. Després and F. Lagoutière, Contact discontinuity capturing schemes for linear advection and compressible gas dynamics. J. Sci. Comput. 16, no. 4, 479–524 (2002)

[10] B. Després and F. Lagoutière, Un schéma non linéaire anti-dissipatif pour l'équation d'advection linéaire, Comptes Rendus de l'Académie des Sciences, 1999.

[11] R. P. Fedkiw, T. Aslam, B. Merriman and S. Osher, A non-oscillatory eulerian approach to interfaces in multi-material flows (the ghost fluid method), Journal of Computational Physics, 152, 457–492, 1999.

[12] I. L. Chern, J. Glimm, O. McBryan, B. Plohr and S. Yaniv, Front tracking method for hyperbolic conservation laws, Journal of Computational Physics, 62, 83–110, 1986.

[13] Y. Chen, J. Glimm, D. H. Sharp and Q. Zhang, A two-phase flow model of the Rayleigh-Taylor mixing zone, Report of the University of Stony Brook (1995).

[14] J. Glimm, J. W. Grove, X. L. Li, K.-M. Shyue, Y. Zeng and Q. Zhang, Three dimensional front tracking, Report of the University of Stony Brook (1996).

[15] E. Godlewski and P.-A. Raviart, Numerical approximation of hyperbolic systems of conservation laws, Springer 1995.

[16] J.-F. Hass, L. Schwaerdele, G. Jourdan and L. Houas, Hot wire measurements of the local Reynolds number in a turbulent mixing zone induced by the Richtmyer-Meshkov instability in a shock tube, in proceedings of the Second international Conference on Inertial Fusion and Apllications, (2001) Kyoto.

[17] A. Harten, On a class of high resolution total-variation-stable finite-difference schemes, SIAM Jour. of Numer. Anal., 21 (1), (1984) 1–23.

[18] A. Harten, High resolution schemes for hyperbolic conservation laws, J. Comp. Phys., 49, (1983) 357–393.

[19] C. W. Hirt and B. D. Nichols, Volume-of-fluid (VOF) method for the dynamics of free boundaries, J. Comp. Phys., 39, 201–225, 1981.

[20] S. Karni, Multicomponent flow calculations by a consistent primitive algorithm, Journal of Computational Physics, 112, 31–43, 1993.

[21] S. Karni, Hybrid multifluid algorithms, SIAM J. Sci. Comp., 17 (5), 1019–1039, 1996.

[22] F. Lagoutière, Modélisation mathématique et résolution numérique de problèmes de fluides compressibles à plusieurs constituants, PhD dissertation, University Paris-VI, 2000.

[23] X. D. Liu and R. P. Fedkiw and S. Osher, A quasi-conservative approach to the multiphase Euler equations without spurious oscillations, Comp. And Applied Math., 11, UCLA, 1998.

[24] W. F. Noh and P. R. Woodward, SLIC (Simple Line Interface Calculation), Springer Lecture Notes in Physics, 25:330–339, 1976.

[25] S. Osher and J. A. Sethian, Front propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations, Journal of Computational Physics, 79, 12–49, 1988.

[26] R. Saurel and R. Abgrall, A simple method for compressible fluid flows, Journal of Computational Physics, 1997.

[27] R. Saurel and R. Abgrall, A multiphase Godunov method for compressible multifluid and multiphase flows, Journal of Computational Physics, 1999.

[28] P. K. Sweby, High resolution schemes using flux limiters for hyperbolic conservation laws, SIAM J. Num. Anal., 21, (1984), 995—1011.

[29] E. F. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics, (Springer-Verlag 1997).

[30] Z. Xu and C.-W. Shu, Anti-diffusive corrections for high-order finite difference WENO schemes, preprint.

[31] S. T. Zalesak, Fully multidimensional flux-corrected transport algorithms for fluids, Journal of Computational Physics, 31, 335–362, 1979.