Abstract

In the context of multicomponent flows, we are faced with PDE systems solutions combining waves whose speeds are several orders of magnitude apart. Of these waves, only the slow kinematic ones that represent transport phenomena are of concern to us. The fast acoustic ones, although uninteresting for the physical application considered, nevertheless impose a prohibitively small time-step (via the classical CFL restriction) if treated explicitly. This is why we propose to use a hybrid finite-volume scheme in which fast waves are handled by a linearized implicit formulation and slow waves remain explicitly solved. To further decrease the computational cost, mostly due to the complexity of nonlinear thermodynamical laws, we combine this method with a fully adaptive multiresolution scheme. At each time step, a multiscale analysis followed by the thresholding of small details enables us to discretize the solution over a time-varying adaptive grid, based on the smoothness of the relevant phenomenon. Particular attention is given to the extension of the reference scheme to non uniform grid and to the prediction strategy of the adaptive grid from one time-step to another. Finally, efficiency in terms of computing time requirements is studied in conjunction with the accuracy performances.

Résumé

Dans le contexte des écoulements multiphasiques, les solutions des systèmes d’EDP font intervenir des ondes dont les vitesses peuvent être d’ordres de grandeur très différent. Parmi ces ondes, seule celle qui représente le phénomène de transport, de cinétique lente, est véritablement intéressante. Les ondes acoustiques rapides, bien que n’intervenant pas du point de vue physique dans l’application considérée, imposent néanmoins un pas de temps excessivement petit (en raison de la condition de stabilité CFL classique) si elles sont traitées de manière explicite. C’est pourquoi nous proposons d’utiliser un schéma volumes-finis hybride dans lequel les ondes rapides sont traitées par une formulation implicite linéarisée et où seules les ondes lentes sont résolues par un schéma explicite. De manière à diminuer encore plus les coûts de calcul, dus principalement à la complexité des lois de fermeture thermodynamiques, nous combinons cette méthode avec un schéma de multirésolution adaptatif. À chaque pas de temps, une analyse multiéchelle suivie d’un seuillage des petits détails permet de discrétiser la solution sur une grille adaptative, variable en temps et basée sur la régularité du phénomène étudié. Un soin particulier est apporté à l’extension du schéma de référence à une grille non uniforme et à la stratégie de prédiction de la grille adaptative d’un pas de temps à l’autre. Finalement l’efficacité en terme de temps de calcul est étudiée en regard de la précision du schéma.
1 Introduction

In this paper, we address the numerical simulation of a 1-D system of conservation laws representing a class of physical problems that involve different waves with speeds separated by several orders of magnitude. Typically, a mixture of gas and oil coming from a pumping system and moving along a pipeline will give birth to two types of waves: fast acoustic (pressure) waves and slow kinematic (transport) waves. From the standpoint of oil production, the slower waves are the only ones of concern, since these correspond to the front displacement of the gas mass-fraction in the mixture. As a matter of fact, their accurate numerical simulation should allow better understanding and prediction of terrain- or severe-slugging, a phenomenon likely to cause serious damages to various industrial facilities [4, 25]. On the other hand, while the fast waves are not interesting in this kind of simulations, they impose a severe stability condition on the time-step, especially in the case of fully explicit schemes.

The first answer to this numerical difficulty consists in using an explicit-implicit time integration. Roughly speaking, this amounts to writing the unknowns in a basis where the actions of slow and fast waves can be somehow easily decoupled. The fast components of the solution will be evolved in time with an implicit scheme, thus eliminating the most severe part of the CFL condition. The slow components will be treated with an explicit scheme, therefore ensuring a high degree of accuracy required for the computation of contact waves. This idea can be proved to be entirely valid and consistent in the linear case. Its extension to the case of nonlinear hyperbolic systems has been extensively and successfully studied, by Faille and Heintze [16] or Baudin et al. [1, 3] for instance. In particular, it meets the very important requirement of conservativity dictated by the physics of the problem.

A realistic computation in a pipeline will consist in modeling the transport of, say, a gas mass-fraction discontinuity over ten kilometers, with an average speed of five meters per second. According to engineers, it is really necessary that, in the neighborhood of this discontinuity, a very small cell size, e.g., of at most one meter, be used in order to capture the motion with enough accuracy. If a uniform mesh were used, this would lead to a huge number of cells in the overall mesh, which is unfortunately not affordable. Now, everywhere else in the mesh, away from this discontinuity, the fluctuations in the solution are due to the supposedly uninteresting acoustic waves. Furthermore, since these acoustic waves are treated implicitly, their possible zones of stiff transitions are smoothed out from the very first time-steps on. Therefore, it is reasonable to think that their computation can be carried out over a much coarser grid.

This observation leads us to extend the multiresolution techniques that have been proposed for explicit schemes in [11, 12] to the specific case of our explicit-implicit scheme. The multiresolution ideas were actually introduced, in the context of systems of conservation laws, by Harten [19] in the early nineties. The basic idea is to perform a smoothness analysis of the solution using a wavelet decomposition and to use this information on the local smoothness to adapt locally the mesh and/or the numerical scheme. In this pioneering work [19], later expanded by [9], the discretization grid remains uniform and the local smoothness indicators defined by the wavelet analysis are used as flags to select between different numerical fluxes at cell interfaces: accurate (but expensive) flux evaluation will be performed in stiff variation areas, while in smooth areas, the fluxes will be roughly (but cheaply) estimated by interpolation of values computed on a coarser grid. In another direction, Cohen et al. [12] have developed a fully adaptive algorithm where the multiscale analysis of the solution is used to design an adaptive grid by selecting the correct level according to the local smoothness out of a hierarchy of nested grids. This non-uniform grid evolves with time thanks to a strategy based on the prediction of the displacement and creation of singularities in the solution. The wavelet basis used to perform the multiscale analysis enables to reconstruct the solution at any time back to the finest level of discretization, within an error tolerance controlled by a thresholding parameter. In theory, the mathematical analysis is complete only in the case of a scalar conservation law, but in practice, the method
has been successfully applied to several PDE systems (see, e.g., [24, 6, 14, 13, 15, 21]). It has recently been upgraded by Müller [22, 23] to become a time-adaptive version, where the solution on the adaptive grid evolves with a locally adapted time-step.

In the present study, we have drawn inspiration from the adaptive algorithm elaborated in [12]. The solution is discretized on a varying grid adapted to its local smoothness. At each time step the multiscale analysis is used as indicators to coarsen or refine the cells in order to follow the evolution of singularities. Time evolution of the solution is performed with a uniform time step ruled by the stability condition on the finest grid. The novelty of our work lies in the way adaptive mesh refinement is coupled with the explicit-implicit time integration scheme. The success of such a coupling is based, once again, on the conservative decomposition of the solution into slow and fast components.

We start, in section 2, by sketching out the principles of the explicit-implicit time integration scheme over a uniform grid, in the general case of a system of hyperbolic conservation laws. In this Roe-like method, the waves are approximated by the eigenvectors of the Jacobian matrix of a linearized problem. Naturally, we assume that the physical setting leads to a clear separation—by several orders of magnitude—of the eigenvalues, or wave speeds. Then, in section 3, we work out the multiresolution method for this generic case. Of course, we will recall the basics of multiscale analysis and indicate the evolution strategy for the adaptive grid. Section 4 is devoted to the specific problem of 1-D two-phase flow without drift, that is, when liquid and gas phases move at the same speed. Implementation of the semi-implicit multiresolution algorithm is first illustrated in section 4 with a Cauchy problem with simple Neumann boundary conditions. Further tests corresponding to more realistic experiments mimicking specific inflows at one end of the pipeline and friction term are dealt with in section 5.

2 Explicit-implicit scheme for conservation laws

This section briefly addresses the design principle of the explicit-implicit time integration scheme (also referred to as semi-implicit scheme from now on) to be used hereafter. Our purpose is to highlight the interest of such a scheme for the sake of a better accuracy regarding slow waves. The details are to be found in [1, 3].

2.1 Setting and notations

Let us consider the nonlinear system of conservation laws

\[ \partial_t U + \partial_x F(U) = 0, \quad x \in \mathbb{R}, \ t \geq 0, \]

where the unknown \( U \) belongs to an open convex subset \( \Omega \subset \mathbb{R}^M \) with \( M \geq 1 \). Here, the Jacobian matrix \( \nabla F(U) \) of the smooth flux function \( F: \Omega \to \mathbb{R}^M \) is assumed to be \( \mathbb{R} \)-diagonalizable for all state vectors \( U \in \Omega \). In other words, the system (1) is hyperbolic. In agreement with our main motivation, the associated eigenvalues \( \lambda_i(U) \), \( 1 \leq i \leq M \), are supposed to be split into two sets of real numbers of several orders of magnitude apart. Up to a relabeling of subscripts, this is expressed by

\[ |\lambda_1(U)| \leq \ldots \leq |\lambda_p(U)| < |\lambda_{p+1}(U)| \leq \ldots \leq |\lambda_M(U)|, \]

for all the states \( U \) under consideration. The first set \( \mathcal{I}_s = \{\lambda_1, \ldots, \lambda_p\} \) contains the slow waves, while the second one \( \mathcal{I}_f \) contains the fast waves.

The physical domain is discretized using a sequence of non-overlapping bounded cells

\[ [x_{j-1/2}, x_{j+1/2}] \] with \( x_{j-1/2} < x_{j+1/2} \) such that \( \mathbb{R} = \bigcup_{j \in \mathbb{Z}} [x_{j-1/2}, x_{j+1/2}] \).

3
Let $\delta x_j = x_{j+1/2} - x_{j-1/2}$ be the length of the $j$th cell and $x_j = x_{j-1/2} + \frac{1}{2} \delta x_j$ its center. Likewise, the time axis is divided into a sequence of non-overlapping intervals

$$[t_n, t_{n+1}] \text{ with } t_n < t_{n+1} \text{ such that } \mathbb{R}^+ = \bigcup_{n \in \mathbb{N}} [t_n, t_{n+1}].$$

The difference $\delta t = t_{n+1} - t_n$ is called the time-step. The finite-volume solution is then defined as the piecewise constant function

$$\tilde{U}(x, t_n) = U^n_j \text{ for } x \in [x_{j-1/2}, x_{j+1/2}],$$

where $U^n_j$ approximates the mean value of $U$ over the $j$th cell at time $t_n$, i.e.,

$$U^n_j \simeq \frac{1}{\delta x_j} \int_{x_{j-1/2}}^{x_{j+1/2}} U(x, t_n) dx.$$

Updating $U^n_j$ for increasing $n$ requires the definition of a numerical flux function $F : \Omega \times \Omega \to \mathbb{R}^M$, so that the scheme reads

$$\frac{1}{\delta t} [U^{n+1}_j - U^n_j] = \left\{ \begin{array}{ll}
-\frac{1}{\delta x_j} [F(U^n_j, U_{j+1}^n) - F(U^n_{j-1}, U^n_j)] & \text{if fully explicit;} \\
\frac{1}{\delta x_j} [F(U^{n+1}_j, U_{j+1}^{n+1}) - F(U^{n+1}_{j-1}, U^{n+1}_j)] & \text{if fully implicit.}
\end{array} \right.$$

### 2.2 Explicit-implicit time integration scheme for a linear system

At this stage, in order to motivate the explicit-implicit time integration scheme, we study in some details the particular case of a linear system $F(U) = AU$. After a change of basis, we have $A = R \Delta R^{-1}$, with $\Delta = \text{diag}(\lambda_1, \ldots, \lambda_M)$. The simplest first-order method is the standard upwind scheme, which is known to be associated with the numerical flux

$$F_{j+1/2} = F(U_j, U_{j+1}) = \frac{1}{2} \{ AU_j + AU_{j+1} - |A|((U_{j+1} - U_j)),$$

with $|A| = R |\Delta| R^{-1}$ and $|\Delta_{j,j}| = |\lambda_j|$. In (8), we have not specified the time superscripts, since this relation is to hold true for both explicit and implicit schemes. In the eigenvectors basis, the waves satisfy a system of decoupled equations

$$\partial_t \alpha^m + \lambda_m \partial_x \alpha^m = 0, \text{ with } (\alpha^1, \alpha^2, \ldots, \alpha^M)^T = R^{-1} U$$

For sake of simplicity, we momentarily assume constant space and time steps $\delta x$ and $\delta t$. Dropping out the index $m$ in the solution, we denote by $\alpha^n(x)$ the approximation of wave $\alpha^m(x, t)$ at time $t_n = n \delta t$.

First, let us look at the slow waves $(m = 1, \ldots, p)$, that we wish to discretize explicitly. Using a first-order upwind scheme, we have

$$\alpha^{n+1}(j \delta x) = \alpha^n(j \delta x) - \nu \lambda^+_m \alpha^n(j \delta x) - \alpha^n((j - 1) \delta x) - \nu \lambda^-_m \alpha^n((j + 1) \delta x),$$

where $\nu = \frac{\delta t}{\delta x}$ and $\lambda^+_m$ (resp. $\lambda^-_m$) is the positive (resp. negative) part of $\lambda_m$. It is easy to check that if the discretization steps are chosen so that $\nu |\lambda_m| = 1$, the wave $\alpha_m$ is discretized exactly by this scheme. In order to study the stability of (9) in the $L^2$ sense [18], we take the Fourier transform of equation (9). Once the amplification factor has been defined as

$$A(k) = \left| \frac{\hat{\alpha}^{n+1}(k)}{\hat{\alpha}^n(k)} \right|, \text{ with } \alpha^n(k) = \int_{-\infty}^{+\infty} e^{ikx} \alpha(x) dx,$$

with $\hat{\alpha}^n(k) =$
it can be shown, by a straightforward calculation, to be equal to
\[ A_{\text{exp}}(k) = |1 - \nu |\lambda_m| + \nu \lambda_m^+ e^{-ik\delta x} - \nu \lambda_m^- e^{ik\delta x}|. \] (11)

The classical CFL condition \( \nu |\lambda_m| \leq 1 \) ensures that \( A_{\text{exp}}(k) \leq 1 \) for all \( k \). The graph of this amplification factor is depicted on left panel of Figure 1 in the variable \( k/\delta x \) in the range \((0, \pi)\). Indeed notice that \( \pi/\delta x \) is about the highest frequency that can be captured by a grid with mesh step \( \delta x \). The constant amplification factor \( A_{\text{exp}} = 1 \) for \( jm = 1 \) reflects the exactness of the scheme while the attenuation of high frequencies observed for smaller values \( \nu |\lambda_m| < 1 \) corresponds to the smearing of discontinuous solution.

Next, let us look at the fast waves \( I \). If we wish to discretize implicitly. Using a first-order upwind scheme, we have
\[
\alpha^{n+1}(j\delta x) = \alpha^n(j\delta x) - \nu \lambda_m^+ [\alpha^{n+1}(j\delta x) - \alpha^n((j+1)\delta x)] - \nu \lambda_m^- [\alpha^{n+1}((j+1)\delta x) - \alpha^{n+1}(j\delta x)],
\] (12)
with the same definition for \( \nu \). In this case, the amplification factor is
\[ A_{\text{imp}}(k) = |1 + \nu |\lambda_m| - \nu \lambda_m^+ e^{-ik\delta x} + \nu \lambda_m^- e^{ik\delta x}|^{-1}. \] (13)

This scheme is unconditionally stable since \( A_{\text{imp}}(k) < 1 \) for all values of \( \nu |\lambda_m| \neq 0 \). On the graph of \( A_{\text{imp}}(k) \), depicted in the right panel of Figure 1, we see that even for \( \nu |\lambda_m| = 1 \), the amplification factor \( A_{\text{imp}}(k) \) is equal to 1 only for \( k = 0 \) which corresponds to the fact that the solution is instantaneously smeared out.

To sum it up, these considerations clearly advocate the use of an explicit time integration for the slow waves with a CFL number controlled by the slow waves velocity to limit the diffusion, and thus necessarily an implicit time integration for the fast waves.

\begin{figure}[h]
\centering
\includegraphics[width=\textwidth]{figure1.png}
\caption{Amplification factors as a function of \( k/\delta x \) in \([0, \pi]\) for different values of \( \nu |\lambda| \).}
\end{figure}

We are now going to combine these two types of time integration into a single compact formulation. The physical domain is restricted to a slab of length \( L \) by imposing Neumann boundary conditions at both ends and the unknowns are the discretized mean values \( U^n = (U^n_j) \), with \( j = 0, \ldots, N_x \). Direct calculations show that the vector of time increments \( \delta U^n = U^{n+1} - U^n \) corresponding to (9) for slow waves and (12) for fast waves is, in reality, the solution of the linear system
\[
\begin{pmatrix}
E_j & D_j & F_j \\
\vdots & \vdots & \vdots \\
E_j & D_j & F_j \\
\end{pmatrix}
\begin{pmatrix}
\delta U^n_0 \\
\delta U^n_{j-1} \\
\delta U^n_j \\
\delta U^n_{j+1} \\
\vdots \\
\delta U^n_{N_x} \\
\end{pmatrix}
= -\begin{pmatrix}
R^n_0 \\
R^n_1 \\
\vdots \\
\end{pmatrix}
\] (14)
2.3 Explicit-implicit scheme for the general nonlinear case

where, after introducing $\nu_j = \frac{\delta t}{\delta x_j}$, the tridiagonal block matrices are

$$E_j = -\nu_j \tilde{A}^+, \quad D_j = \text{Id} + \nu_j |\tilde{A}|, \quad F_j = \nu_j \tilde{A}^-.$$  (15)

The most remarkable feature of (15) lies in the new matrix

$$\tilde{A} = R\tilde{\Delta}R^{-1}, \quad \text{with} \quad \tilde{\Delta} = \text{diag}(0, \ldots, 0, \lambda_{p+1}, \ldots, \lambda_M),$$  (16)

whose role is to decide whether or not implicitation on a given wave component should be activated. Indeed, since the right-hand side is the block vector of explicit residuals

$$R_j^n = \nu_j [F(U_j^n, U_{j+1}^n) - F(U_{j-1}^n, U_j^n)],$$  (17)

it can be easily seen that if $\tilde{A} = 0$ (resp. $\tilde{A}$) instead of (16), then the system (14) degenerates to the fully explicit (resp. implicit) scheme.

2.3 Explicit-implicit scheme for the general nonlinear case

It is possible to extend this method to the nonlinear case in a conservative way, provided that a Roe-type numerical flux function is used for the reference scheme. To emphasize the similarity with the linear case we denote by $\mathcal{A} : \Omega \times \Omega \rightarrow \text{Mat}(\mathbb{R}^M)$ this associated Roe linearization, which must satisfy the standard conditions [18]

$$\begin{cases}
\mathcal{A}(U, U) = \nabla F(U), \\
\mathcal{A}(U_L, U_R) = (R\Delta R^{-1})(U_L, U_R) \text{ is } \mathbb{R}\text{-diagonalizable}, \\
\mathcal{F}(U_R) - \mathcal{F}(U_L) = \mathcal{A}(U_L, U_R)(U_R - U_L),
\end{cases}$$  (18)

for all states $U, U_L$ and $U_R$ in $\Omega$. Denoting by $\mathcal{A}_{j+1/2}$ the Roe matrix $\mathcal{A}(U_j, U_{j+1})$, the numerical flux function entering in (7) is then

$$\mathcal{F}_{j+1/2} = \mathcal{F}(U_j, U_{j+1}) = \frac{1}{2} \{ \mathcal{F}(U_j) + \mathcal{F}(U_{j+1}) - |\mathcal{A}_{j+1/2}|(U_{j+1} - U_j) \},$$

where the absolute value of the Roe matrix is defined by

$$|\mathcal{A}| = \mathcal{A}^+ - \mathcal{A}^- = R|\Delta| R^{-1}, \quad \text{with} \quad |\Delta| = \text{diag}(|\lambda_1|, \ldots, |\lambda_M|).$$  (19)

Following the standard approach described in [20] for instance, we consider an approximation of the fully implicit scheme

$$U_j^{n+1} = U_j^n - \nu_j [\mathcal{F}_{j+1/2}^{n+1} - \mathcal{F}_{j-1/2}^{n+1}],$$  (20)

by first freezing the Roe matrix to its $n$th time step value in the definition of $\mathcal{F}_{j+1/2}^{n+1}$, and then approaching the flux $\mathcal{F}(U_j^{n+1})$ at time step $n+1$ by its first order Taylor expansion around $U_j^n$. Denoting $U_j^{n+1} - U_j^n$ by $\delta U_j^n$, this gives for the approximate numerical flux

$$\mathcal{F}_{j+1/2} = \mathcal{F}_{j+1/2}^{n} + \frac{1}{2} (\nabla_{U_j^n} \mathcal{F}(U_j^n) + |\mathcal{A}_{j+1/2}^n|) \delta U_j^n _n + \frac{1}{2} (\nabla_{U_{j+1}^n} \mathcal{F}(U_{j+1}^n) - |\mathcal{A}_{j+1/2}^n|) \delta U_{j+1}^n.$$  (21)

This linearized implicit scheme can put under a tridiagonal block linear system similar to (14), in which the block matrices are now

$$\begin{align*}
E_j &= -\nu_j (\Delta_{j+1} U_j^n) + |\mathcal{A}_{j-1/2}^n|, \\
D_j &= \text{Id} + \nu_j |\mathcal{A}_{j-1/2}^n| + |\mathcal{A}_{j+1/2}^n|, \\
F_j &= \nu_j (\Delta_{j+1} U_{j+1}^n) - |\mathcal{A}_{j+1/2}^n|.
\end{align*}$$  (22)
and the right-hand side is none other than the explicit residuals (21) as in the linear case (17). In its diagonal form, the Roe matrix exhibits the characteristic wave speeds in the diagonal matrix and we propose now to treat the slow waves explicitly and only the fast acoustic waves with the implicit scheme above, by analogy with the linear case. In view of (14), not only $A_{j+1/2}$ but also the matrices $\nabla F(U_j^n)$ must be changed by first diagonalizing

$$A_{j+1/2} = R_{j+1/2}\Delta_{j+1/2}R_{j+1/2}^{-1}, \quad \Delta_{j+1/2} = \text{diag}(\lambda_{j+1/2}^1, \ldots, \lambda_{j+1/2}^M),$$

(23)

then by canceling the first $p$ entries of the diagonal matrices $\Delta_{j+1/2}$ and $\Gamma_j$, and finally returning back to the original basis. This procedure works fine in practice [16, 3] as shall be demonstrated in sections 4 and 5.

Before closing this section, it is necessary to discuss about the actual of various CFL ratios involved. The study of the amplification factors has evidenced the fact that, on one hand, the discretization parameters $\Delta t$ and $\Delta x$ should respect a stability condition based on the largest “slow” wave, i.e.,

$$\delta t \max_j \frac{|\lambda_p(U_j)|}{\Delta x_j} < \text{CFL}_{exp} \leq 1.$$  

(24)

On the other hand, the damping undergone by the fast waves has to be somewhat controlled. To this effect, we impose the additional restriction

$$\delta t \max_j \frac{|\lambda_M(U_j)|}{\Delta x_j} < \text{CFL}_{imp} \simeq 20.$$  

(25)

### 3 Multiscale analysis of the explicit-implicit scheme

In [12], a multiresolution analysis was proposed defined for a system of hyperbolic conservation laws treated with an explicit scheme. We are now going to extend it to the case of the explicit-implicit scheme previously introduced.

#### 3.1 Basics of multiresolution analysis

We first recall a few fundamental notions about multiscale analysis. Let us be given some uniform mesh, considered to be the finest level. Starting from this finest discretization, we define a hierarchy of $K$ nested grids by dyadic coarsening. We attribute the label 0 to the coarsest level with cells of length $\Delta x = h$, as shown in Figure 2. Let $U$ be a scalar function. Initially, $U$ is defined on the finest grid with cell width $\Delta x = 2^{-K} h$, and thus, is represented by the sequence of real scalars $U^K = (u^K_j)$. Since the numerical solution will be obtained with a finite-volume scheme, it is natural to consider the given discrete function $u^K_j$ as mean values on the cells of the finest grid $K$.

We now need to define operators in order to switch from one level to another. The coarsening operator $P_{k-1}^k$ will consist in cell averaging from one grid to the coarser one, i.e.,

$$U^{k-1} = P_{k-1}^k U^k \quad \text{with} \quad u_{j,k-1} = \frac{1}{2}(u_{2j}^k + u_{2j+1}^k).$$  

(26)

The inverse operator consists in recovering the mean values on grid level $k$, given the mean values on the coarser level $k-1$. This involves an approximation—or prediction—operator $P_{k-1}^k$. In contrast to the coarsening operator, there is an infinite number of choices for the definition of $P$. However, it should obey some rules of locality and consistency (e.g., $P_{k-1}^k \circ P_{k-1}^k = \text{Id}$) in
We define the prediction error $E^k$ at a given level $k$ as the difference between the solution on level $k$ and its reconstruction $\hat{U}^k$ using the solution at level $k - 1$. Thanks to the consistency property $e_{2j} = -e_{2j+1}$, we define the detail vector $D^{k-1}$ with $d_{j}^{k-1} = u_{2j}^{k} - \hat{u}_{2j}^{k}$ and use it along with $U^{k-1}$ to entirely recover $U^k$.

The two vectors $U^k$ and $(U^{k-1}, D^{k-1})$ are of same length. Iterating this encoding operation from the finest level down to the coarsest provides the multiscale representation $M = (U^0, D^0, \ldots, D^{K-1})$. Using the local structure of the operators $P_{k-1}^{k}$ and $P_{k}^{k}$, the multiscale transformation

$$
M : U^K \mapsto M^K, 
$$

and its inverse $M^{-1}$ can be implemented with an optimal $O(N_K)$ complexity, where $N_K$ represents the dimension of the finest grid $\nabla^K$. The interest of the multiscale representation lies in the fact that thanks to the consistency of the prediction operator, the local regularity of the function is reflected by the size of its details. We can use this property to compress the function in the multiscale domain by dropping all details below a given level-dependent threshold. To clarify this idea, we first define a thresholding operator $T_{\Lambda}$ acting on the multiscale representation $M^K$, depending on a subset $\Lambda \subset \nabla^K$ of indices $\lambda = (j, k)$, by

$$
T_{\Lambda}(d_{\lambda}) = \begin{cases} 
0 & \text{if } \lambda \in \Lambda \\
d_{\lambda} & \text{otherwise}
\end{cases} 
$$

Given level-dependent threshold values $\varepsilon = (\varepsilon_k)_k$, we introduce the subset $\Lambda_\varepsilon$ as

$$
\Lambda_\varepsilon = \Lambda(\varepsilon_0, \varepsilon_1, \ldots, \varepsilon_K) := \{ \lambda \text{ s.t. } |d_{\lambda}| \geq \varepsilon_{|\lambda|} \}. 
$$

This completes the definition of the thresholding operator $T_{\varepsilon} := T_{\Lambda_\varepsilon}$, and gives rise to an approximating operator $\mathcal{A}_\varepsilon$ acting on the physical domain representation

$$
\mathcal{A}_\varepsilon := M^{-1}T_{\varepsilon}M. 
$$

In practice, we take advantage of the fact that the remaining fine-scale details will be concentrated near singularities. This is not such a trivial result because the operator $\mathcal{A}_\varepsilon$ is nonlinear.
since $\Lambda_\varepsilon$ depending on $U^K$ by (30). We refer to [10] for a thorough investigation of nonlinear approximation and the proof of the main result

$$|U^K - A_\varepsilon U^K|_{L^1} < C\varepsilon_K$$

(32)

that we will be using here, for when $\varepsilon = 2^{-K}\varepsilon_K$. The estimation (32) is used in several areas where data compression is a crucial issue such as image processing. In those applications, the zeroed details are simply not encoded. For our problem, it can be used to define an adaptive grid where the local size of the cell will be the grid step corresponding to the finest non-zero detail.

The representation of $U_\varepsilon = A_\varepsilon U^K$ on this adaptive grid is intermediate between the physical representation $U^K$ on the finest grid and the encoded multiscale representation $M^K$. Note, in particular, that the representation by its mean value $u^k_j$ on an intermediate level $k$ does not mean that the function is locally constant on this cell of width $2^k h$, but simply that its mean values on the finest grid in this area can be recovered — within the $\varepsilon$ accuracy — using the mean values on this intermediate level and the reconstruction operators $P_{l-1}^l$ for $l = k + 1, \ldots, K$. In the sequel, we will call partial decoding the algorithm that computes $U_\varepsilon$ on this adaptive grid from $M^K$, and partial encoding the reverse transformation. In order to perform these operations with the minimum complexity, the tree $\Lambda_\varepsilon$ giving the indices of relevant details will be inflated with all indices corresponding to cells in the reconstruction stencils, to become what we call a graded tree.

### 3.2 Prediction strategy for the tree

So far, we have considered a discrete function $U^K$ independent of time. Before addressing our real problem (with a semi-implicit scheme, as stated in section 2), we briefly recall the basics of the adaptive scheme as was proposed in [12] for an underlying explicit scheme on the finest uniform grid $\nabla^K$. Formally, the update of the function $U$ reads

$$U^{n+1}_K = U^n_K + \nu B^K_n.$$  

(33)

Its singularities will move with time, which means that the tree $\Lambda_\varepsilon$ depends on time and that its computation using (30) must be performed at each time step. Of course, we wish to compute this time-dependent tree without having to decode the solution back to the finest grid at each time-step, since this would destroy all the benefits of the adaptive computation. This is possible, thanks to Harten’s heuristic [19] that we briefly summarize here. This intuitive idea relies on the hyperbolic nature of the PDE’s system, which ensures that the singularities of the solution move at finite speed. More specifically, if we denote by $\Lambda^n_\varepsilon$ the graded tree obtained by applying $A_\varepsilon$ to $U^n_K$, then $\Lambda^n_\varepsilon$ can be inflated into $\tilde{\Lambda}_\varepsilon^n$ containing $\Lambda^{n+1}_\varepsilon$ as well as $\Lambda^n_\varepsilon$, ensuring that both estimations

$$\|U^n_K - A_{\tilde{\Lambda}^{n+1}_\varepsilon} U^n_K\| \leq C\varepsilon_K \quad \text{and} \quad \|U^{n+1}_K - A_{\tilde{\Lambda}_\varepsilon^{n+1}} U^{n+1}_K\| \leq C\varepsilon_K$$

(34)

are satisfied. Setting $\tilde{\Lambda}^{n+1}_\varepsilon$ to $S^K$ does the trick but it is not very interesting in practice. The inflated tree $\Lambda^{n+1}_\varepsilon$ should be as small as possible. The inflation strategy proposed by Harten consists in adding immediate neighbors of cells where the detail is above the level-dependent threshold and the two subdivisions of cells where the detail is more than twice this threshold. This scheme works fine in practice, although a more refined strategy is required in order to obtain an error estimate. Such an analysis is performed in [12] in the case of an explicit finite volume scheme for a scalar hyperbolic equation with a CFL condition ensuring that a discontinuity cannot move further that one cell away within one time step.

Harten’s heuristic can be extended to the case of an explicit-implicit scheme for vector hyperbolic system of conservation laws. Below is the actual adaptive algorithm we implemented.
1. Initialization: encoding of the initial solution and definition of $\Lambda^0$

2. Loop over time-steps $n = 0, \ldots, N - 1$:
   - Prediction of $\Lambda^{n+1}$ and partial decoding of $\Lambda^n$
   - Evolution of $\Lambda^n$ to $\Lambda^{n+1}$ on the adaptive grid $\Lambda^{n+1}$
   - Encoding of $\Lambda^{n+1}$ and definition of $\Lambda^{n+1}$

3. Decoding of $\Lambda^N$ on the finest grid

An important point in this algorithm is the evaluation of the numerical fluxes between adjacent cells of the adaptive grid, which must be performed in order to update the solution $\Lambda^n$ into $\Lambda^{n+1}$. In the case the underlying uniform scheme is of first-order, the numerical flux is a function of the mean values on each side of the interface. If the adaptive grid data is used straightforwardly the local order of the scheme is of order one with respect to the local grid size which can be quite large. Experience shows that it is in fact necessary to locally reconstruct the mean values on the finest grid on each side of the interface. This can be done in $\mathcal{O}(N_A)$ operations, with $N_A$ the number of cells in the adaptive grid, thanks to the linearity and uniformity of the reconstruction (27) and this is implemented in the numerical simulations presented in this paper. When the underlying uniform scheme is of higher order, the local reconstruction of the solution on the finest grid near the interfaces is less crucial since in the smooth regions the high order of numerical scheme will be able to compensate for the coarseness of the grid. The alternative consisting in applying the underlying high order scheme directly on the adaptive grid solution can be used, up to some modification of the high order non linear reconstruction, to take into account situations where interfaces can separate two cells of length in a ratio of two [23].

The novelty here is that the underlying uniform scheme is not fully explicit in time but the semi-implicit scheme described in paragraph 2. The stability conditions (24)–(25) on the discretization parameters ensure that singularities moving with the slow waves speed will propagate at most one cell away within one time step, but those carried away by the fast waves, typically the acoustic phenomena which can actually contain high frequency structures, do not verify this assumption. Therefore the predicted tree $\Lambda^n$ may not be adapted to their representation. On the other hand, since there is at least one order of magnitude between the slow and fast waves speed, enlarging the tree $\Lambda^n$ taking into account the much weaker condition of fast waves (25) would require so much refinement of the solution that all computational and complexity benefits would disappear. In fact, thanks to the implicit treatment of the fast waves, it turns out that the original refinement strategy —based on the explicit scheme stability condition (24)— can still be used, for heuristic reasons that we will now illustrate on the ground of simple considerations in the linear case.

### 3.3 Heuristical justification

In section 2, we have already pointed out that the implicit scheme attenuates high frequencies of the solution —and quite dramatically so, as illustrated in Figure 1— when the CFL number is in the range of 20, which is relevant for our physical problem. This means that the high frequency content in the fast waves, requiring to be encoded on the finest levels of multiresolution, will be severely damped by the time implicit scheme and that most of it will be adequately represented on coarser levels. Considering that the adaptive tree is graded, the refinement rule consisting in adding to the tree the immediate neighbors of cells with non negligible details requires in consequence the recursive addition of the cells at coarser levels belonging to the prediction stencil of these neighbors. The width of the prediction stencil for the quadratic reconstruction that we use in practice is six times the width of the small cell of size $2^{-k}$ where the solution is to be predicted.
At this point it is interesting to observe that events in the solution are transported by the implicit scheme at different speeds that depend on their frequency content. This is well quantified in the linear transport case by the group velocity $d\omega(k)/dk$ which is the propagation velocity of the energy, as put forward by Trefethen in [26]. For the linear implicit scheme (12), assuming $\lambda > 0$, the phase velocity is

$$\omega(k) = \Re \left\{ \frac{I\lambda}{\nu k \delta x} \log \left[ \frac{1}{1 + \nu \delta x \delta t} \right] \right\}.$$  

As for the group velocity, it is represented in Figure 3 as a function of $k \delta x \in [0, \pi]$ (what is actually displayed is the group velocity divided by the eigenvalue $\lambda$). The figure shows that the energy associated to high frequencies in the signal are transported at a much slower speed that the nominal wave speed. The discrepancy between the speeds is all the greater as the CFL number $\nu \lambda$ increases. In addition, it clearly indicates that the estimate

$$\nu \left| \frac{d\omega(k)}{dk} \right| \leq C \nu \lambda \leq 1,$$

holds true for frequencies $k \geq k_c(\nu \lambda)$, where the cut-off frequency $k_c$ decreases rapidly with the CFL number $\nu \lambda$. The above inequality just expresses that the CFL number based on the group velocity is readily less than one for $\nu \lambda = 20$ so that the energy associated with the high frequencies do not propagate more than one cell during one time step, hence the heuristic to keep the tree prediction strategy unchanged.

![Figure 3: Implicit scheme group velocity as a function of $k/\delta x$ in $[0, \pi]$ for different values of $\nu \lambda$.](image)

To further assess this last remark we perform the following numerical experiments: the semi-implicit scheme is applied to a linear system of two waves, the slow one with speed $\lambda_1 = 1$, the other fast one with speed $\lambda_2 = 20$. For the sake of comparison, computations with the uniform scheme are performed in each case, along with the multiresolution scheme using a hierarchy of five levels and a threshold parameter equal to 0.001. A first set of computations is designed on a slab of length $L = 5.12$ with 512 cells on the finest level (32 on the coarsest one). A first computation is performed with a CFL condition $\lambda_1 \delta t/\delta x = 1$ - imposed to ensure the stability and in fact the exactness of the explicit scheme used for the slow wave. This choice of discretization parameters ($\delta t = \delta x = 1$) leads to $\lambda_2 \delta t/\delta x = 20$ which verifies the condition (25) recommended at paragraph 2. Another computation is performed with a time step twenty times smaller - corresponding to a CFL condition $\lambda_2 \delta t/\delta x = 1$.

In Figure 4, the solutions at $t = 0.1$ are represented. In the left panel, the fast wave computed by the uniform scheme and by the multiresolution scheme for the two time steps are displayed along with the initial condition and the exact solution (the latter is obtained using the explicit
scheme with $\lambda_2 \delta t / \delta x = 1$). The shapes are quite different and the group velocity displayed in Figure 3 can somewhat enlighten their interpretation. The implicit solution obtained with $\lambda_2 \delta t / \delta x = 1$ is dissipated symmetrically around the exact profile, while the solution obtained with $\lambda_2 \delta t / \delta x = 20$ exhibits little resemblance with the exact profile. In particular, the top of the initial discontinuity has almost not moved at all: it is high frequency and the group velocity graph in Figure 3 predicts that high frequency components of the signal have a group velocity reduced by a factor of ten. In the right panel of Figure 4, which represents the slow wave obtained with the same parameters, we note that the results corresponding to $\lambda_1 \delta t / \delta x = 1$ are exact, while those obtained with $\lambda_2 \delta t / \delta x = 0.05$ are slightly diffused almost symmetrically around the exact one, and that there are no differences between the multiresolution and the uniform solution.

Figure 4: Uniform and multiresolution solutions after one time step. Fast wave (left panel) and slow wave (right panel).

The corresponding adaptive trees are represented in Figure 5. Several interesting features can be noticed, which corroborates the remarks made above: the tree at $t = 0.1$ when using the implicit scheme with $\lambda_2 \delta t / \delta x = 1$ is not well adapted to the exact solution for the fast wave, since this one has moved with speed 20 when the tree has moved with the slow wave speed equal to 1. It is nevertheless sufficient to represent the high frequency content of the fast wave, because as was seen in the previous Figure 4, the initial discontinuity has moved very slowly. The faster modes correspond to smoother part of the solution and are reasonably represented on coarser levels of discretization, which accounts for the good agreement between the uniform

Figure 5: Adaptive trees at time $t = 0.1$. Implicit solution with $\nu \lambda_2 = 1$ and $\nu \lambda_2 = 20$ (left panel). Explicit solution with $\nu \lambda_2 = 1$ and initial condition (right panel).
and multiresolution solution.

Figure 6: Solutions (left panel) and adaptive trees (right panel) at time $t = 0.3$. Implicit solution with $\nu \lambda_2 = 20$ with uniform and multiresolution scheme.

In Figure 6, we display the solutions at time $t = 0.3$ obtained with discretization parameters $\lambda_2 \delta t / \delta x = 20$, that is, after three time steps. The interesting points are the very good agreement between the uniform and adaptive solutions and the disappearance of the higher level of multiresolution in the fast wave representation. Figures 7 and 8 illustrate the long time behavior of the semi-implicit scheme. Simulations have been performed on a slab of length $L = 40.96$ discretized with 4096 cells on the finest level, and the waves have propagated for 100 time steps, until $t = 10$, corresponding again to $\lambda_2 \delta t / \delta x = 20$. At this stage, the fast wave has been completely smoothed out and the coarsest grid is sufficient to represent it as accurately as the uniform solution computed on the finest grid.

Figure 7: Long time behavior: initial condition (left panel) and adaptive trees at times $t = 0$ and $t = 10$ (right panel).

4 Application to multi-component flows

In order to illustrate the effectiveness of the multiresolution method proposed for the implicit-explicit time integration, we now address the example of a system of conservation laws coming from the modeling of two-phase flows in pipelines.
Figure 8: Long time behavior of the fast wave (left panel) and slow wave (right panel) at time $t = 10$. Implicit solution with $\nu \lambda_2 = 20$ with uniform and multiresolution scheme.

4.1 Drift-flux model and its relaxed version

In this model, both phases are supposed to move at the same velocity, which makes it a handy case of a more general and more sophisticated class of models known as Drift-Flux Model. From the physical point of view, it is considered as quite a good model for horizontal flows. Let us consider the PDE system

$$
\begin{align*}
\partial_t (\rho) + \partial_x (\rho v) &= 0, \\
\partial_t (\rho Y) + \partial_x (\rho v Y) &= 0, \\
\partial_t (\rho v) + \partial_x (\rho v^2 + P) &= 0,
\end{align*}
$$

(35)

in which, $\rho$ denotes the density of a mixture of liquid (oil) and gas, $v$ is the averaged velocity while $Y$ stands for the gas mass fraction in the mixture. The pressure law $P$ is given as the thermodynamic closure law

$$
P(\rho, \rho Y) = a_g^2 \frac{\rho Y}{1 - \frac{\rho Y}{\rho_l} (1 - Y)},
$$

(36)

obtained when assuming the liquid phase to be incompressible (e.g. with a constant density $\rho_l > 0$, typically $\rho_l = 1000 \text{kg} \cdot \text{m}^{-3}$) and the gas phase to obey an isothermal law (e.g. with a constant sound speed $a_g > 0$, in the applications $a_g = 100 \text{m} \cdot \text{s}^{-1}$). The natural state space associated with (35)--(36) is

$$
\Omega = \left\{ \mathbf{U} = (\rho, \rho Y, \rho v)^T \in \mathbb{R}^3 \mid 0 < \frac{\rho}{\rho_l} < \frac{1}{1 - Y}, \; v \in \mathbb{R}, \; 0 < Y < 1 \right\}.
$$

(37)

The PDE system (35)--(36) is easily seen to be hyperbolic over $\Omega$ and equipped with the three distinct eigenvalues

$$
v - c(\mathbf{U}) < v < v + c(\mathbf{U}), \quad \text{with} \; c(\mathbf{U}) = a_g \frac{\sqrt{Y}}{1 - \frac{\rho Y}{\rho_l} (1 - Y)} > 0.
$$

(38)

The intermediate eigenvalue can be shown to correspond to a linearly degenerate field (the so-called transport wave for the gas mass fraction $Y$) while the two others are associated with genuinely nonlinear fields (the so-called acoustic waves). We shall be concerned with solutions of (35)--(36) exhibiting small values of the dimensionless ratio $|v|/c(\mathbf{U})$, so that the extreme waves will generically evolve with speeds much larger than the intermediate one.
In a related setting for petroleum mixtures, Baudin et al. [2] have shown how to conveniently approximate the solutions of (35)–(36) by those of an extended system with singular perturbations. Such a relaxation procedure is virtually insensitive to the precise form of the exact pressure law since it precisely intends to relax the nonlinearities in \( P(\rho, \rho Y) \). Moreover, it satisfies several stability properties [2] and in turn, allows for the derivation of a Roe type linearization which is mandatory in the implementation of our time integration technique. Following [2] (see also Bouchut [5], Chalons-Coquel [7] and the references therein), we consider the relaxation model

\[
\begin{align*}
\frac{\partial}{\partial t} (\rho)^k + \frac{\partial}{\partial x} (\rho v)^k &= 0, \\
\frac{\partial}{\partial t} (\rho Y)^k + \frac{\partial}{\partial x} (\rho Y v)^k &= 0, \\
\frac{\partial}{\partial t} (\rho v)^k + \frac{\partial}{\partial x} (\rho v^2 + \Pi)^k &= 0, \\
\frac{\partial}{\partial t} (\rho \Pi)^k + \frac{\partial}{\partial x} (\rho \Pi v + a^2 v)^k &= k \rho^k (P(\rho^k, \rho Y^k) - \Pi^k),
\end{align*}
\]

where the additional unknown \( \Pi \) is easily seen to formally coincide with the exact pressure law \( P(\rho, \rho Y) \) in the limit of an infinite relaxation parameter \( k > 0 \). Hence and again formally, the solutions of (35)–(36) are expected to be recovered from those of (39) in such a limit. After Chen, Levermore and Liu [8], however, it is known that to prevent the relaxation approximation procedure from instabilities in the regime of a large relaxation parameter, some sub-characteristic conditions must be met. In the present setting, these stability conditions are fulfilled provided that the real parameter \( a > 0 \) entering (39) is prescribed so as to meet (see [5], [7], [2]):

\[
a > \rho c(\mathbf{U}) = a g \frac{\rho v Y}{1 - \frac{a}{\rho} (1 - Y)} > 0,
\]

for all states under consideration. Besides this stability condition, the first-order underlying system in (39) is easily seen to be hyperbolic for all \( a > 0 \) and to admit the following four eigenvalues

\[
v - \frac{a}{\rho} < v, \quad v < v + \frac{a}{\rho},
\]

with the important property that all the fields are linearly degenerate. Notice that within the set of solutions of interest here and in view of the sub-characteristic condition (40), the two extreme contact waves in (39) will naturally play the role of fast waves by contrast to the intermediate ones. To shorten the notations, let us introduce

\[
\mathbb{V} = (\rho , \rho Y, \rho v , \rho \Pi)^T, \quad \mathbb{G} = (\rho v, \rho Y v, \rho v + \Pi, \rho \Pi v + a^2 v)^T,
\]

and the associated natural phase space

\[
\Omega_{\mathbb{V}} = \{ \mathbb{V} = (\mathbf{U}, \rho \Pi) \in \Omega \times \mathbb{R} \}.
\]

4.2 Reference numerical scheme

Let us then briefly state the numerical procedure for approximating the weak solutions of (35)–(36) on the basis of the relaxation model (39)–(40). Given some approximation of the equilibrium solution at time \( t^n \), say,

\[
\mathbb{U}^n(x) = (\rho^n, (\rho Y)^n, (\rho v)^n)^T(x),
\]

this one is evolved at the next time level \( t^{n+1} = t^n + \delta t \) into two steps
1. **Evolution in time.** For times $t' \in (0, \delta t)$, we solve the Cauchy problem for the relaxation system (39)--(40) when setting the relaxation parameter $k$ to zero

\[
\begin{align*}
\frac{\partial_t (\rho)}{\partial_t (\rho v)} &= 0, \\
\frac{\partial_t (\rho Y)}{\partial_t (\rho v^2 + \Pi)} &= 0,
\end{align*}
\]

with an initial data prescribed according to the equilibrium condition

\[V(x, t' = 0) = V^n(x) = (\rho^n, (\rho Y)^n, (\rho v)^n, (\rho \Pi)^n = \rho P(\rho, \rho Y)^n)^T(x).\]  

(44)

2. **Relaxation.** At time $t' = \delta t$, we solve the ODE problem

\[
\begin{align*}
\frac{\partial_t (\rho)}{\partial_t (\rho Y)} &= 0, \\
\frac{\partial_t (\rho v)}{\partial_t (\rho \Pi)} &= k\rho(P(\rho, \rho Y) - \Pi^k),
\end{align*}
\]

with initial data $V(x, t' = \delta t)$ while sending the relaxation parameter $k$ to $+\infty$. This second step in the method clearly amounts to set

\[
\begin{align*}
(\rho)^{n+1}(x) &= (\rho)(x, t' = \delta t), \\
(\rho Y)^{n+1}(x) &= (\rho Y)(x, t' = \delta t), \\
(\rho v)^{n+1}(x) &= (\rho v)(x, t' = \delta t), \\
(\rho \Pi)^{n+1}(x) &= \rho P(\rho, \rho Y)(x, t' = \delta t).
\end{align*}
\]

(47)

The above numerical relaxation procedure can be given a numerical flux function $G : \Omega \times \Omega \rightarrow \mathbb{R}^4$ defined for any given two states $V_j, V_{j+1}$ by

\[G(V_j, V_{j+1}) = G(\omega(0^+, V_j, V_{j+1})),\]

(48)

where $\omega(\cdot, V_j, V_{j+1})$ denotes the solution of the Riemann problem for (44) with initial data

\[V = \begin{cases} 
V_j & \text{if } x < 0, \\
V_{j+1} & \text{if } x > 0.
\end{cases}\]

This self-similar solution reads (see [2] for the details)

\[\omega(\frac{x}{t}, V_j, V_{j+1}) = \begin{cases} 
V_j & \text{if } x/t \leq v_j - a/\rho_j, \\
v_j^* & \text{if } v_j - a/\rho_j < x/t \leq v^*, \\
V_{j+1} & \text{if } v^* < x/t \leq v_j + 1/\rho_j + a/\rho_j, \\
v_j + 1/\rho_j + a/\rho_j < x/t, 
\end{cases}\]

where the velocity $v^*$ is given by

\[v^* = v_j^* = v_{j+1}^* = \frac{1}{2}(v_j + v_{j+1}) + \frac{1}{2a}(\Pi_j - \Pi_{j+1}).\]

(49)

As for the two intermediate states $V_j^*$ and $V_{j+1}^*$, these are recovered from

\[
\begin{align*}
\rho_j^* &= \frac{a^2 \rho_j}{\rho_j (\Pi_j - \Pi^*) + a^2}, \\
\rho_{j+1}^* &= \frac{a^2 \rho_{j+1}}{\rho_{j+1} (\Pi_{j+1} - \Pi^*) + a^2}, \\
Y_j^* &= Y_j, \\
\Pi_j^* &= \Pi^*,
\end{align*}
\]

(50)

16
with

$$\Pi^* = \frac{1}{2}(\Pi_j + \Pi_{j+1}) + \frac{1}{2}a(v_j - v_{j+1}).$$  \hspace{1cm} (51)$$

As observed in [2], the linear degeneracy property of all the fields in (39) allows to re-express equivalently the Godunov numerical flux function (48) in terms of a Roe type numerical flux function

$$G(V_j, V_{j+1}) = \frac{1}{2} \{G(V_j) + G(V_{j+1}) - |A(V_j, V_{j+1})|(V_{j+1} - V_j)\},$$  \hspace{1cm} (52)$$

where the Roe linearization $A(V_j, V_{j+1}) : \Omega_V \times \Omega_V \rightarrow \text{Mat}(\mathbb{R}^4)$ is defined in its diagonalized form $A(V_j, V_{j+1}) = A_{j+1/2} = R_{j+1/2} \Delta_{j+1/2} L_{j+1/2}$ by

$$\Delta_{j+1/2} = \text{diag}(v_j - a/\rho_j, v^*, v^*, v_{j+1} + a/\rho_{j+1}),$$

$$R_{j+1/2} = \begin{pmatrix} 1 & 0 & 1 & v_j - a/\rho_j & 0 & v^* & v_{j+1} + a/\rho_{j+1} \\ Y_j & 1 & 0 & Y_{j+1} & v_j - a/\rho_j & 0 & v^* \\ 0 & 0 & Y_j & Y_{j+1} & v_j - a/\rho_j & 0 & v^* \\ 0 & 0 & 0 & Y_j & Y_{j+1} & v_j - a/\rho_j & 0 & v^* \\ \Pi_j + a^2/\rho_j & 0 & \Pi^*_{j+1} + a^2/\rho_{j+1} & 0 & \Pi_j + a^2/\rho_j & 0 & \Pi^*_{j+1} + a^2/\rho_{j+1} \end{pmatrix},$$  \hspace{1cm} (53)$$

$$L_{j+1/2} = (L^1_{j+1/2} L^2_{j+1/2} L^3_{j+1/2} L^4_{j+1/2}) = R_{j+1/2}^{-1},$$

with

$$L^1_{j+1/2} = \begin{pmatrix} \rho^*Y_j - \Pi^* \\ \rho^*Y_{j+1} - \rho^*_{j+1}Y_j + \rho^*_{j+1}Y_{j+1} + \rho^* Y_{j+1} - \rho^*_{j+1}Y_j + \rho^*_{j+1}Y_{j+1} \\ 0 \\ 0 \end{pmatrix}, \quad L^2_{j+1/2} = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix},$$

$$L^3_{j+1/2} = \begin{pmatrix} \rho^*_{j+1} - \rho^* \\ \rho^*_{j+1} - \rho^*_{j+1} \\ \rho^*_{j+1} - \rho^*_{j+1} \\ \rho^*_{j+1} - \rho^*_{j+1} \end{pmatrix}, \quad \text{and} \quad L^4_{j+1/2} = \begin{pmatrix} -\rho^*_{j+1}Y_j + \rho^*_{j+1}Y_{j+1} - \rho^*_{j+1}Y_j + \rho^*_{j+1}Y_{j+1} - \rho^*_{j+1}Y_j + \rho^*_{j+1}Y_{j+1} - \rho^*_{j+1}Y_j + \rho^*_{j+1}Y_{j+1} \\ -\rho^*_{j+1}Y_j + \rho^*_{j+1}Y_{j+1} - \rho^*_{j+1}Y_j + \rho^*_{j+1}Y_{j+1} - \rho^*_{j+1}Y_j + \rho^*_{j+1}Y_{j+1} - \rho^*_{j+1}Y_j + \rho^*_{j+1}Y_{j+1} \end{pmatrix}.$$

On the ground of this Roe type linearization, we are now in a position to apply the time implicit-explicit algorithm, described in part 2, in the state variable $V$. Let us recall that the extreme waves in this Roe linearization, propagating respectively at speed $v_j - a/\rho_j$ and $v_{j+1} + a/\rho_{j+1}$, are actually the fast waves while the two other intermediate ones are the slow waves. This distinction, dictated by the forthcoming applications, leads us to introduce $\hat{A}_{j+1/2} = R_{j+1/2}\hat{\Delta}_{j+1/2} R_{j+1/2}^{-1}$ with $\hat{\Delta}$ restricted to the large eigenvalues

$$\hat{\Delta}_{j+1/2} = \text{diag}(v_j - a/\rho_j, 0, 0, v_{j+1} + a/\rho_{j+1}).$$

Likewise, the Jacobian matrix of the system (44) must also be diagonalized as

$$\nabla G(V_j) = \{P_j \mid \Gamma_j \mid P^{-1}_j\}(V_j),$$

with $\mid \Gamma_j \mid (V_j) = \text{diag}(|v_j - a/\rho_j|, |v_j|, |v_j|, |v_j + a/\rho_j|).$
where the eigenvector matrices $\mathcal{P}_j$ and $\mathcal{P}_{j+1}$ are easily recovered from the definition of the flux function $\mathcal{G}$ in (42). Here again, we replace $\Gamma_j(\mathcal{V}_j)$ by

$$\tilde{\Gamma}_j(\mathcal{V}_j) = \text{diag}(v_j - a/\rho_j, 0, 0, v_j + a/\rho_j).$$

Let us mention that Baudin et al. in [2] have shown how to infer from the flux formula (52) expressed in the state variable $\mathcal{V}$ a reduced numerical problem in the $\mathcal{U}$ equilibrium variable. The technical details in order stay, however, beyond the scope of the present paper and the interested reader is referred to [2].

We now have defined the numerical scheme in the generic context introduced in section 2. Its coupling with the multiresolution as described in section 3 consists in performing the two steps (44) and (46) on the time dependent adaptive grid $\Lambda^n$. We have seen in section 3 that the definition of $\Lambda^n$ and prediction of $\Lambda^n$ is performed by comparing the size of the coefficient of the solution in the multiresolution basis with a threshold parameter $\varepsilon$. Here, the solution is a four-component vector $(\rho, \rho Y, \rho u, \rho \Pi)$ whose amplitude range can be several order of magnitudes apart, therefore the detail component will be normalized before being compared to $\varepsilon$. Denoting by $d_{j,k}(u)$ the $u$-component of detail of $\mathcal{U}$, and by $|u|$ the supremum over all cells of the $u$-component in absolute value, the adaptive grid $\Lambda_{\varepsilon}$ is defined by

$$\Lambda_{\varepsilon} = \{(j, k); |d_{j,k}(\rho)| \geq \varepsilon_k |\rho| \text{ or } |d_{j,k}(\rho Y)| \geq \varepsilon_k |\rho Y| \text{ or } |d_{j,k}(\rho u)| \geq \varepsilon_k |\rho u| \}$$

(54)

The first three components must be tested, since an initial condition with a single singularity, on the $Y$ component for instance, can evolve and create singularities on the other components. At the time where the multiscale analysis of the solution is performed, the fourth component $\rho \Pi$ contains the equilibrium value of the pressure and its singularities are all contained in the other three.

In numerical simulations, we used a second-order enhancement in space of the numerical flux described above, which consists in using on both sides of the interface a linear reconstruction of the primitive unknowns $\rho$, $u$, $Y$ and $\Pi$, stabilized by a minmod limiter. In fact, in order to preserve the positivity of the density, only $u$, $Y$ and $\Pi$ are reconstructed and $\rho$ is then deduced of the reconstructed values of $Y$ and $\Pi$ using the thermodynamic equilibrium law (36).

### 4.3 Numerical validation

The first set of numerical simulations is meant to validate the method on a simple problem with Neumann boundary conditions. The pipeline length is $L = 1000$ m for a diameter of 0.146 m and a single discontinuity is initially located at $x = 500$ m. The density of the liquid is $\rho_L = 1000$ Kg/m$^3$ and the sound speed in the gas is equal to $a_g = 100$ m/s.

The solution is initialized with the following left and right states

$$\begin{pmatrix} \rho_1 \\ Y_1 \\ v_1 \\ \Pi_1 \end{pmatrix} = \begin{pmatrix} 500 \\ 0.2 \\ 10 \\ P(\rho_1, (\rho Y)_1) \end{pmatrix} \quad \text{and} \quad \begin{pmatrix} \rho_2 \\ Y_2 \\ v_2 \\ \Pi_2 \end{pmatrix} = \begin{pmatrix} 400 \\ 0.4 \\ -10.4261 \\ P(\rho_2, (\rho Y)_2) \end{pmatrix}.$$

It evolves into a three-waves solution with the speeds $\sigma^- = -77.7$ m/s, $\sigma^0 = -4.62$ m/s and $\sigma^+ = 76.7$ m/s. We represent on figure 9 the components $\rho$, $Y$ and $u$ of the solution along with the time adaptive grid, at two different times of the simulation $t = 1$ and 3 seconds. We see on the $\rho$ component the original contact discontinuity at $x = 500$ meters moving slowly to the left and two other discontinuities interact moving fast in opposite directions which correspond to the shock discontinuities. On the $Y$ component only the original discontinuity is slowly transported to the left and on the $u$ component it splits itself into the two acoustic waves.
Figure 9: Solution and adaptive grid at different times.
On each figure we display the initial condition, the solution obtained with the time explicit scheme on the uniform grid (for the sake of comparison) and the solution obtained with the time semi-implicit multiresolution scheme. The explicit scheme is second order in space with limited linear reconstruction and in time with a Heun quadrature. The uniform grid corresponds to the finest level of the multiresolution hierarchy with 2048 cells. The semi-implicit scheme is second order in space and first order in time. The three discontinuities are well captured by the multiresolution algorithm as illustrated by the adaptive tree displayed on the bottom line. Because the intermediate discontinuity is treated explicitly it is accurately resolved and therefore uses the finest level of resolution until the end of the simulation, while the two other waves being implicitly treated, are as expected smeared and thus their description requires only intermediate levels of resolution. The solution is displayed after full reconstruction on the finest level of discretization at the time of the snapshot. It cannot then be distinguished from the solution computed on the same grid with the uniform semi-implicit scheme. The stability condition is ruled by the fast wave in the explicit case (see 24), with a CFL number equal to 0.5, and the time step is then of the order of 0.0025. In the semi-implicit case the time step is dictated by the slow transport wave and is about ten times larger.

This first simulation illustrates first that slow and fast waves can actually be properly handled by the semi implicit scheme in conservative form, and also that the multiresolution scheme is able to capture and follow singularities.

The second test consists in comparing the explicit, implicit and semi-implicit schemes on a large time experiment, in their uniform and multiresolution versions. We start the simulation with a piecewise constant initial solution taking the values \( U_1 \) for \( x \) between 200 and 300 meters and the values \( U_2 \) elsewhere, with

\[
U_1 = \begin{pmatrix}
\rho_1 \\
Y_1 \\
v_1 \\
\Pi_1
\end{pmatrix} = \begin{pmatrix}
357.143 \\
0.2 \\
1.3141 \\
10^6
\end{pmatrix} \quad \text{and} \quad U_2 = \begin{pmatrix}
\rho_2 \\
Y_2 \\
v_2 \\
\Pi_2
\end{pmatrix} = \begin{pmatrix}
918.182 \\
0.01 \\
357.1431 \\
10^6
\end{pmatrix}.
\]

The slow wave transports the \( Y \) profile to the right during 300 seconds, with an average time step equal to 0.0022 when using the explicit scheme and 0.0876 when using the semi-implicit scheme. On figure 10 we display a zoom of the \( Y \) component of the solution at final time, along with the multiresolution tree. As expected, the time semi-implicit method achieves much sharper fronts than the fully implicit scheme and also the fully explicit scheme. In both the semi-implicit and the explicit schemes the transport wave is handled explicitly, but with a much smaller time step in the later case, since the CFL is based on the fast acoustic wave stability condition, which is also treated explicitly. The ratio between the time steps of the two schemes is almost forty, which accounts for the more diffusive character of the explicit solution.

The second set of figures related to this test, labeled 11, illustrates the performances of the multiresolution scheme in terms of computing time, grid size and precision. In these curves the parameter \( \varepsilon \) corresponds to the threshold value on the finest level of discretization, with 2048 cells over the 1km long pipeline. Three multiscale computations are compared with 5, 6 or 7 levels of coarsening. The accuracy of the computation is measured by estimating the \( L^1 \) error between the multiscale solution reconstructed on the finest level and the uniform scheme solution at final time \( t = 300 \). The first set of curves representing the error as a function of \( \varepsilon \) exhibits the expected \( O(\varepsilon) \) behavior, although it is not proved mathematically in this case. The next two figures represent the variations of the computing time and the size of the adaptive grid as a function of the threshold parameter. As expected, these two parameters decrease when the threshold becomes less severe. An interesting point is that the curves corresponding to 6 and 7 levels of resolution are quite close together compared to the 5 levels one. The reason is that the solution is seldom smooth enough to enable the coarsening down to the 7th level and the performances of this simulation are therefore quite similar to the one allowing coarsening...
Figure 10: $Y$ component of solution and adaptive grid at final time.
only down to the 6th level. The last figure (bottom right) representing the variations of the $L^1$ error as a function of the computing time is the most interesting one, from the practical point of view, since it tells what we can really expect to gain, and at which cost in accuracy. All these curves must be seen in view of the uniform simulation performances which requires 323 seconds to compute the solution on the 2048 cells on the finest level. All the multi-resolution simulations are roughly 5 to 6 times faster, and require ten times less grid points. Because we use log scale on the vertical axis (for the error), the points corresponding to the multiscale computation with $\epsilon = 0$, are not represented on the curves. They produce the same solution as the uniform computation since no details are thresholded but require more computing time (roughly 20%) because of the overhead costs induced by the coding, thresholding and decoding at each time step.

![Graphs showing L1 error, memory, computing time, and error versus threshold.]

Figure 11: Performances of the multi-resolution.

5 Simulation of flows in pipelines

We now turn to more realistic simulations taking into account operating conditions at both ends of the pipeline.

5.1 Boundary conditions and initial data

At the inlet, we wish to prescribe the gas and total mass flow rates, namely,

$$\bar{q}(t) = (\rho v)(x = 0, t) \quad \text{and} \quad \bar{q}_G(t) = (\rho Y v)(x = 0, t).$$

At the outlet, we wish to monitor the pressure, as well as a no liquid backdrop condition, that is,

$$\bar{P}(t) = P(x = L, t) \quad \text{and} \quad v(x = L, t) < 0 \Rightarrow Y(x = L, t) = 1.$$
These boundary conditions are implemented by introducing one ghost cell at each end of the pipeline as described in [17]. The physical state on the ghost cell is computed so that the solution of the Riemann problem between this ghost cell and the first cell of the grid yields the operating inlet or outlet conditions. The length of the pipeline is $L = 8\text{km}$. The liquid has a density $\rho_l = 1000\text{kg}\cdot\text{m}^{-3}$ and the sound speed in the gas is $a_g = 200\text{m}\cdot\text{s}^{-1}$. The inlet flow rates (55) are set in such a manner that: (i) the corresponding gas mass fraction have the double-bell shape

$$Y(t) = \frac{\bar{q}_G(t)}{q(t)} = \begin{cases} 0.1 & \text{for } t < 110, \\ 0.1 + 0.6 \frac{t-100}{10} & \text{for } 100 \leq t < 110, \\ 0.7 & \text{for } 110 \leq t < 600, \\ 0.7 - 0.4 \frac{t-600}{10} & \text{for } 600 \leq t < 610, \\ 0.3 & \text{for } 610 \leq t < 700, \\ 0.3 + 0.5 \frac{t-700}{10} & \text{for } 700 \leq t < 710, \\ 0.8 & \text{for } 710 \leq t < 1200, \\ 0.8 - 0.6 \frac{t-1200}{10} & \text{for } 1200 \leq t < 1210, \\ 0.2 & \text{for } 1210 \leq t, \end{cases}$$

(57)

as depicted in the left panel of Figure 12; (ii) the total flow rate follows a similar double-bell pattern, as shown in the middle panel of Figure 12. As far as the outlet pressure is concerned, it is monitored according to the time variations displayed in the right panel of the same figure.

Realistic operating conditions should also take into account friction, which leads us to add the source term

$$S(\rho, pv) = -\frac{2C_f pv|pv|}{D\rho}$$

(58)

to the right-hand side of the momentum balance equation of the model. Here, $C_f = 0.005$ is the friction coefficient and $D = 0.146\text{m}$ is the pipeline diameter. The drift-flux model (35), which should be more appropriately called zero-drift model here, becomes

$$\begin{aligned} &\partial_t(\rho) + \partial_x(\rho v) = 0, \\
&\partial_t(\rho Y) + \partial_x(\rho Y v) = 0, \\
&\partial_t(pv) + \partial_x(pv^2 + P) = S. \end{aligned}$$

(59)

The initial data $\bar{U}(x) = \bar{U}(x, t = 0)$ for this experiment is set as the steady-state solution of (59) with the boundary conditions taken at time $t = 0$, that is,

$$\begin{aligned} &\bar{d}_x(\bar{\rho}) = 0 \\
&\bar{d}_x(\bar{\rho} Y) = 0 \\
&\bar{d}_x(\bar{\rho} v^2 + P) = \bar{S}, \end{aligned}$$

(60)

with

$$\begin{aligned} &\bar{\rho} \bar{v}(x = 0) = \bar{q}(t = 0), \\
&\bar{\rho} \bar{Y} \bar{v}(x = 0) = \bar{q}_G(t = 0), \\
&P(x = L) = P(t = 0). \end{aligned}$$

(61)

### 5.2 Error analysis

Several numerical experiments are performed on this test case. First of all, we check the convergence of the uniform scheme assuming that the exact solution is well approximated by a reference solution obtained with a very fine discretization (8192 points). The error is estimated on the gas mass fraction by

$$E_1(\delta x) = \frac{\sum_{i=1}^{N_x} |Y_i - Y_{i}^{\text{ref}}|}{\sum_{i=1}^{N_x} |Y_{i}^{\text{ref}}|},$$

(62)
where $Y_{i}^{ref}$ is the gas mass fraction of the reference solution averaged on the corresponding 8192/$N_x$ cells. The trend of the relative $L^1$-error as a function of $\delta x = L/N_x$ is represented in figure 13. It is not too different of the order 1 trend which is represented by a straight line for reference. Simultaneously, the computing time varies in $N_x^2$, since the CFL ratio has been maintained constant.

Figure 13: $L^1$ norm of the relative error on gas mass fraction as a function of $\delta x$ (left) and zoom of the gas mass fraction for different discretizations (right).

We next turn to the multiresolution experiments. In theory, multiscale analysis in time is required in order to capture variations in the experimental boundary conditions and introduce small cells accordingly. This feature has not been implemented yet, and at the present time, we must impose the boundary conditions on the uniform grid. The finest grid is used at both ends of the pipeline at all times, even when the solution is very smooth there. In figure 14, we represent a snapshot of the gas mass fraction and of the velocity field at time $t = 3000$ s. The solution obtained with a uniform grid of 1024 cells and the multiresolution solution obtained on a hierarchy of six levels from 32 to 1024 cells are almost identical; for the chosen threshold value $\varepsilon = 10^{-4}$, the relative $L^1$-error between the two solutions is less than $8 \times 10^{-4}$. Superimposed to the solution are the active cells at this time. The finest levels of discretization are used only in the vicinity of the sharp variations of $Y$. Towards the end of the pipeline, intermediate levels of discretization are used, even though the $Y$ field is almost constant, due to a sharper variation of the velocity field in this region.

We check the good behavior of the scheme by computing the error between the multiresolution solution reconstructed on the finest grid at the final time and the uniform scheme solution on the same grid. Here again, we select the gas mass fraction as the most interesting component,
and define the error

\[ E_1(\varepsilon, \delta x) = \frac{\sum_{i=1}^{N_x} |Y_i^\varepsilon - Y_i^{\text{uni}}|}{\sum_{i=1}^{N_x} |Y_i^{\text{uni}}|} \]  \hspace{1cm} (63)

We try several values of the threshold \( \varepsilon \) and of the number of coarsening levels, and record along with the corresponding error the maximum number of cells used by the multiresolution during the computation. In Figure 15, we represent the variation of the relative error on the gas mass fraction versus the CPU gain (on the left hand side) and versus the memory gain (on the right hand side). The CPU gain is the ratio of the computing time using the uniform scheme on the finest grid over the computing time using the multiresolution scheme. The memory gain is the ratio between the number of cells on the finest grid and the maximum number of cells used at any time step during the multiresolution simulation. Two sets of results corresponding to a hierarchy of five (\( \times \) symbols) or six (\( + \) symbols) levels are displayed, with 1024 cells on the finest grid in both cases. Note that if the memory gain is always larger than one, the CPU gain on the opposite can be less than one, i.e., the multiresolution can be disadvantaging if the threshold parameter is set to a very small value, thus enforcing the used of the finest level of discretization almost everywhere. In that case, the overhead costs due to the encoding and decoding of the solution in the multiscale analysis render it prohibitive.

In Figure 16, we compare performances in term of error versus CPU and memory gain for simulations done on a hierarchy of eight levels up to 4096 cells on the finest grid (with \( \times \) symbols) and on a hierarchy of six levels up to 1024 cells on the finest grid (with \( + \) symbols). From these figures, it is clear that the advantage of the multiresolution increases as the uniform grid of reference is more refined.

6 Conclusion

As announced in the abstract, the numerical method proposed in this paper successfully combines two existing schemes that were already implemented in slightly different but still related contexts. The explicit-implicit time integration scheme applied by Baudin et al. to two-phase flow models allows to use a time-step dictated by the slow transport wave velocity, instead of the prohibitively small time-step that should have been imposed by the fast acoustic waves, had these been treated explicitly. Our special design to the case of the zero-drift model with a source term to take friction into account behaves correctly. A multiresolution analysis of solution is performed at each time step to monitor the adaptive grid. A typical simulation as shown in
Figure 15: Performances in term of relative error versus computing time (left) and number of active cells (right) for different numbers of levels.

Figure 16: Performances in term of relative error versus computing time (left) and number of active cells (right) for different finest discretizations.
the numerical section of this paper is performed on a hierarchy of six levels of resolution and
achieves a CPU gain of five to six compared to the same simulation on the finest uniform grid.
The robustness of the adaptive scheme has been given special attention, with respect to the pre-
diction of the adaptive grid from one time-step to the next, and it has been shown and explained
—albeit somewhat heuristically—that even though the fast acoustic wave evolve with a CFL
condition twenty times larger than the transport wave, the enlargement strategy designed for
explicit schemes with a CFL < 1 can still be used.

The future developments will be devoted to more complicated models involving drift, non
horizontal—crooked or even vertical—pipelines. Multiresolution with adaptive time stepping
as proposed by Müller et al. [22] is also currently investigated.

References

[1] M. Baudin. Méthode de relaxation pour la simulation des écoulements polyphasiques dans
[9] G. Chiavassa, R. Donat, and S. Muller. Multiresolution-based adaptive schemes for hyper-
conservation laws. In F. Benkhaldoun R. Vielsmeier and D. Hänel, editors, *Finite Volumes
gas dynamics. In *Hyperbolic problems: theory, numerics, applications. Vol. I, II (Magde-


