3D instationary supersonic gas dynamics problems, has been demonstrated by Berger et al. in [4].

Recently, multiresolution (MR) techniques have become popular for hyperbolic conservation laws, going back to the seminal work of Harten [36] in the context of finite volume schemes and cell-average MR analysis. Starting point is a finite volume scheme for hyperbolic conservation laws on a regular grid. Subsequently a discrete multiresolution analysis is used to avoid expensive flux computations in smooth regions, first without reducing memory requirements, e. g. for 1D hyperbolic conservation laws (Harten [36]), 1D conservation laws with viscosity (Bihari [8]), 2D hyperbolic conservations laws (Bihari and Harten [9]), 2D compressible Euler equations (Chiavassa and Donat [14]), 2D hyperbolic conservation laws with curvilinear patches (Dahmen et al. [19]) and unstructured meshes (Abgrall and Harten [1], Cohen et al. [16]). A fully adaptive version, still in the context of 1D and 2D hyperbolic conservation laws, has been developed to reduce also memory requirements (Gottschlich-Müller and Müller [35], Kaibara and Gomes [40], Cohen et al. [17]). This algorithm has been extended to the 3D case and to parabolic PDEs (Roussel et al. [48], Roussel and Schneider [47]), and more recently to self-adaptive global and local time-steppings (Müller and Stiriba [45], Domingues et al. [23–25]). Therewith the solution is represented and computed on a dynamically evolving automatically adapted space-time grid. Different strategies have been proposed to evaluate the flux without requiring a full knowledge of fine grid cell-average values. Applications to shock waves in compressible flows addressing the issue of shock resolution have been presented in [5], and extensions to the Navier–Stokes equations in the weakly compressible regime can be found in [49]. Adaptive MR methods with operator splitting have been proposed for multiscale reaction fronts with stiff source terms in [27, 28]. The numerical analysis of the above higher order operator splitting techniques has been performed in [20, 29] and it was shown that the splitting time step can be even larger than the times scales involved in the PDEs. Adaptive MR computations using the above method with complex chemistry and including detailed transport can be found in [21], further applications involving various stiffness levels have been presented in [30, 31].

The MR approach has also been used in other contexts. For instance, the Sparse Point Representation (SPR) method was the first fully adaptive MR scheme, introduced by Hölmstrom [38] in the context of finite differences and point-value MR analysis, leading to both CPU time and memory reduction. In the SPR method, the wavelet coefficients are used as regularity indicators to create locally refined grids, on which the numerical solution is represented and the finite difference discretization of the operators is performed. Applications of the SPR method have been published in [22, 46]. Discontinuous Galerkin methods have been applied to hyperbolic conservation laws in [13] using Haar wavelet indicators to decide where to refine or to coarsen the meshes. These publications reveal that the multiresolution concept has been applied by several groups with success to different stiff problems. For comprehensive literature about the subject, we refer to the books of Cohen [15] and Müller [44].

The objective of the paper is the extension of the adaptive multiresolution method [24, 48] to the numerical simulation of detonation waves. As model we use here a one-step chemical reaction involving two chemical species only. Since the chemical source term is stiff, a time splitting has to be made between the convective and source terms and each term needs to be computed with a different time step and a different time integration method. An error-controled time step based on a Runge–Kutta–Fehlberg approach is used for the source term integration. The application concerns Chapman-Jouguet detonations in one dimension with different stiffness values and instabilities of detonations waves due to an interaction with a pocket of partially burnt gases in two dimensions. These detonation problems motivate the use of the Euler equations, although an extension of the method to compute viscous flows using the Navier–Stokes equations is straightforward, as proposed in [49].

The outline of the paper is the following: first, we present the set of reactive Euler equations for a simplified detonation model. Then we describe the finite volume discretization. A Strang splitting technique is utilized to account for temporal scales in the source term that do not influence the hydrodynamics. We also briefly summarize the multiresolution strategy. Finally, we show the numerical results for the test problems in one and two space dimensions and we set the conclusions together with some perspectives for future work.

2 Governing equations

For modelling the combustion process, we use the reactive Euler equations, as described in [18, 34]. The simplest description of a chemically reacting gas flow assumes that the gas mixture is made only of two chemical species, the burnt gas, denoted with subscript b and the unburnt gas, denoted with subscript u. The unburnt gas is converted to burnt gas via a single irreversible reaction. We represent the mixture state by a single scalar variable Z corresponding the mass fraction of the unburnt gas. We also assume gases in the mixture to be ideal polytropic gases with equal specific heat ratio γ and specific gas constant r. The system of equations in two dimensions, which has been non-dimensionalized in a suitable way, may be written as

$$\frac{\partial Q}{\partial t}+\frac{\partial F}{\partial x}+\frac{\partial G}{\partial y}=S, \tag{1}$$

where $Q=(\rho,\rho v_x,\rho v_y,\rho e,\rho Z)^T$ and

$$F=\begin{pmatrix}\rho v_x\\ \rho v_x^2+p\\ \rho v_x v_y\\ (\rho e+p)v_x\\ \rho v_x Z\end{pmatrix},\ G=\begin{pmatrix}\rho v_y\\ \rho v_x v_y\\ \rho v_y^2+p\\ (\rho e+p)v_y\\ \rho v_y Z\end{pmatrix},\ S=\begin{pmatrix}0\\0\\0\\0\\-k(T)\rho Z\end{pmatrix} \tag{2}$$

Here ρ denotes the mixture density, $V=(v_x,v_y)^T$ the mixture velocity, e the mixture total energy per unit of mass, p the pressure, T the temperature and k the chemical reaction rate. The two equations of state completing the model are

$$p=\rho rT \tag{3}$$

and

$$e=\frac{p}{\rho(\gamma-1)}+\frac{V^2}{2}+Q_0Z \tag{4}$$

where Q_0 denotes the amount of heat per unit of mass released in the chemical reaction.

The reaction rate $k(T)$ of the irreversible chemical reaction is expressed in Arrhenius form as

$$k(T)=A\exp\left(-\frac{T_A}{T}\right) \tag{5}$$

where the pre-exponential coefficient A and the activation temperature T_A are empirical constants. When the reaction source term is stiff, however, the reaction rate may be simplified by adopting the so-called ignition temperature kinetic model, i. e.

$$k(T)=\begin{cases}\frac{1}{\tau} & \text{if} \quad T\geq T_i\\ 0 & \text{if} \quad T<T_i\end{cases} \tag{6}$$

where T_i denotes the ignition temperature and τ the characteristic time of the chemical reaction, which determines the stiffness of the problem. This formulation has been chosen in the applications presented in the numerical results section. However, the numerical method is not limited to this simplified model and its extension to the Arrhenius law or even more complex chemical reactions is possible.

3 Numerical method

In this section, we first describe the classical Strang splitting and the space discretization of the convective terms. Subsequently, the time integration is discussed and, first, for the convective terms, a time step depending on the CFL is chosen, then, for the source term, we split the convective time step into error-controlled time steps using an explicit Dormand-Prince method. A different choice has been proposed by Duarte et al. in [28], based on implicit and explicit Runge-Kutta methods and a posteriori error estimators. We also briefly recall the multiresolution method, previously published in [23, 48].

3.1 Strang splitting

We denote by $C(Q)$ the operator of the convective terms. Equation (1) becomes

$$\frac{\partial Q}{\partial t} = C(Q) + S(Q). \tag{7}$$

Discretizing explicitly with first order in time, we get

$$Q^{n+1} = Q^n + \Delta t \left[C(Q^n) + S(Q^n)\right] \tag{8}$$

where n denotes the time instant and Δt the convective time step.

The splitting relies on the separation of the convective and source terms operators. To get a first-order discretization, it writes

$$Q^\star = Q^n + \Delta t\ S(Q^n) \tag{9}$$

$$Q^{n+1} = Q^\star + \Delta t\ C(Q^\star) \tag{10}$$

i. e.

$$Q^{n+1} = C^{o1}_{\Delta t}\ S^{o1}_{\Delta t}\ Q^n \tag{11}$$

where $S^{o1}_{\Delta t}$ denotes the first-order accurate source term operator with time step Δt and $C^{o1}_{\Delta t}$ the first-order accurate convection operator with time step Δt.

To get second-order accuracy, which corresponds to the classical Strang, the following procedure can be applied

$$Q^{n+1} = S^{o2}_{\Delta t/2}\ C^{o2}_{\Delta t}\ S^{o2}_{\Delta t/2}\ Q^n \tag{12}$$

where $S^{o2}_{\Delta t/2}$ denotes the second-order accurate source term operator with time step $\Delta t/2$ and $C^{o2}_{\Delta t}$ the second-order accurate convection operator with time step Δt.

We note that the splitting time step in the above method is fixed. Techniques to introduce adaptive splitting time steps based on a posteriori error estimators have been introduced in [20, 27] allowing automatic error control.

3.2 Space discretization of the convective terms

Discretization in space is made using a finite volume method, to ensure conservative flux computations. Convective terms are discretized using the AUSM+ scheme [43]. In this procedure, pressure terms are computed separately. The Euler flux F writes

$$F = \begin{pmatrix} \rho v_x \\ \rho v_x^2 + p \\ \rho v_x v_y \\ (\rho e + p) v_x \\ \rho v_x Z \end{pmatrix} = M\ c \begin{pmatrix} \rho \\ \rho v_x \\ \rho v_y \\ \rho h \\ \rho Z \end{pmatrix} + \begin{pmatrix} 0 \\ p \\ 0 \\ 0 \\ 0 \end{pmatrix} \tag{13}$$

where M denotes the Mach number, c the speed of sound and h the enthalpy per unit of mass. Denoting by Φ the purely convective term and by Π the pressure term, we discretize the Euler flux F in space following

$$F_{i+\frac{1}{2}} = M_{i+\frac{1}{2}}\ c_{i+\frac{1}{2}}\ \Phi_{i+\frac{1}{2}} + \Pi_{i+\frac{1}{2}} \tag{14}$$

where the indices refer to nodal values.

The interface speed of sound is $c_{i+\frac{1}{2}} = \sqrt{c_i c_{i+1}}$ and the interface convective term is

$$\Phi_{i+\frac{1}{2}} = \begin{cases} \Phi_i & \text{if } M_{i+\frac{1}{2}} \geq 0 \\ \Phi_{i+1} & \text{otherwise} \end{cases} \tag{15}$$

The terms $M_{i+\frac{1}{2}}$ and $p_{i+\frac{1}{2}}$ follow

$$M_{i+\frac{1}{2}} = M_i^+ + M_{i+1}^- \tag{16}$$

$$p_{i+\frac{1}{2}} = P_i^+\ p_i + P_{i+1}^-\ p_{i+1} \tag{17}$$

where

$$M_i^{\pm} = \begin{cases} \frac{1}{2}\left(M_i \pm |M_i|\right) & \text{if } |M_i| \geq 1 \\ \pm\frac{1}{2}\left(M_i \pm 1\right)^2 \pm \frac{1}{8}\left(M_i^2 - 1\right)^2 & \text{otherwise} \end{cases} \tag{18}$$

and

$$P_i^{\pm} = \begin{cases} \frac{1}{2}\left(1 \pm \operatorname{sign}(M_i)\right) & \text{if } |M_i| \geq 1 \\ \frac{1}{4}\left(M_i \pm 1\right)^2\left(2 \mp M_i\right) \pm \frac{3}{16} M_i \left(M_i^2 - 1\right)^2 & \text{otherwise} \end{cases} \tag{19}$$

Third-order accuracy of the spatial discretization far from discontinuities is obtained using a MUSCL interpolation [51], together with a Koren slope limiter [41]. Denoting by q one of the conservative quantities (i. e. density, momentum, energy, partial mass of the unburnt gas), the corrected value via a MUSCL interpolation is

$$q_i' = q_i + \frac{1}{6}\phi\left(r_i\right)\left(q_i - q_{i-1}\right) + \frac{1}{3}\phi\left(\frac{1}{r_i}\right)\left(q_{i+1} - q_i\right) \tag{20}$$

and

$$q'_{i+1} = q_{i+1} - \frac{1}{3}\phi\left(r_{i+1}\right)\left(q_{i+1} - q_i\right) - \frac{1}{6}\phi\left(\frac{1}{r_{i+1}}\right)\left(q_{i+2} - q_{i+1}\right) \tag{21}$$

where

$$r_i = \frac{q_{i+1} - q_i}{q_i - q_{i-1}} \tag{22}$$

and

$$\phi(r) = \max\left[0, \min\left(2r, \frac{1+2r}{3}, 2\right)\right] \tag{23}$$

Nevertheless, since we use a second-order accurate Strang splitting in time, the global accuracy of the scheme remains second-order.

3.3 Time integration of the convective terms

Time integration of the convective term is made using a classical third-order TVD Runge-Kutta scheme, i. e.

$$Q^\star = Q^n + \Delta t \; C\left(Q^n\right) \tag{24}$$

$$Q^{\star\star} = \frac{1}{4}\left[3\;Q^n + Q^\star + \Delta t\; C\left(Q^\star\right)\right] \tag{25}$$

$$Q^{n+1} = \frac{1}{3}\left[Q^n + 2\;Q^{\star\star} + 2\;\Delta t\; C\left(Q^{\star\star}\right)\right] \tag{26}$$

The corresponding Runge-Kutta tableau is given in Table 1.

The convective time step Δt is chosen to satisfy the Courant-Friedrichs-Levy (CFL) condition. In the computations, $CFL = 0.5$ is used.

3.4 Time integration of the stiff source terms

Due to the stiffness of the chemical source terms in case of detonation computations, a high order time integration is chosen to ensure the numerical stability. As the chemical time scale is much smaller than the convective one we apply a much smaller time step Δt_d than for the convection terms. However, this is only required in a small area of the computational domain, i. e. in the reaction zone. In the rest of the computational domain, the source term is almost equal to zero.

For this reason, we introduce an adaptive local time step $\Delta t_d = \frac{\Delta t}{2N}$, N being the number of substeps required by the source term computation. In order to adapt this local time step with time, a fourth-fifth order embedded Runge-Kutta method is used. The computational error between the fourth-order and the fifth-order method enables to decide wether the local time step needs to be increased

Table 1: Butcher tableau corresponding to the compact TVD third-order Runge-Kutta method.

0			
1	1		
$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	0
	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{2}{3}$

Table 2: Butcher tableau corresponding to the Dormand-Prince method. The first row of *b* coefficients gives the fourth-order accurate solution, and the second row yields order five.

0							
$\frac{1}{5}$	$\frac{1}{5}$						
$\frac{3}{10}$	$\frac{3}{40}$	$\frac{9}{40}$					
$\frac{4}{5}$	$\frac{44}{45}$	$-\frac{56}{15}$	$\frac{32}{9}$				
$\frac{8}{9}$	$\frac{19372}{6561}$	$-\frac{25360}{2187}$	$\frac{64448}{6561}$	$-\frac{212}{729}$			
1	$\frac{9017}{3168}$	$-\frac{355}{33}$	$\frac{46732}{5247}$	$\frac{49}{176}$	$-\frac{5103}{18656}$		
1	$\frac{35}{384}$	0	$\frac{500}{1113}$	$\frac{125}{192}$	$-\frac{2187}{6784}$	$\frac{11}{84}$	
	$\frac{5179}{57600}$	0	$\frac{7571}{16695}$	$\frac{393}{640}$	$-\frac{92097}{339200}$	$\frac{187}{2100}$	$\frac{1}{40}$
	$\frac{35}{384}$	0	$\frac{500}{1113}$	$\frac{125}{192}$	$-\frac{2187}{6784}$	$\frac{11}{84}$	0

or decreased, as proposed by Fehlberg [32].

$$Q_{o4}^{n+1} = Q^n + \Delta t_d \sum_{i=1}^{s} b_{i,o4} k_i \tag{27}$$

$$Q_{o5}^{n+1} = Q^n + \Delta t_d \sum_{i=1}^{s} b_{i,o5} k_i \tag{28}$$

where

$$k_i = S\left(t^n + c_i \Delta t_d, Q^n + \Delta t_d \sum_{i=1}^{s} \sum_{j=1}^{s} a_{i,j} k_j \right) \tag{29}$$

In this article, an embedded explicit Dormand-Prince method [26] was chosen. The coefficients are computed in order to minimize the error of the fifth-order solution. This is the main difference with the Fehlberg method, which was constructed so that the fourth-order solution has a small error. For this reason, the Dormand-Prince method is more suitable when the higher-order solution is used to continue the time integration. The coefficients are given in Table 2.

The local relative error of the quantity q is denoted by e and the accepted tolerance by ε. The optimal time step $\sigma\ \Delta t_d$ can be determined by multiplying the scalar σ times the current time step Δt_d. The scalar σ is given by

$$\sigma = \left(\frac{\varepsilon\ \Delta t_d}{2\ e\ \Delta t_{max}}\right)^{\frac{1}{4}} = \left(\frac{\varepsilon\ \Delta t_d}{e\ \Delta t}\right)^{\frac{1}{4}} \tag{30}$$

In practice, however, the new time step $\Delta t'_d$ is determined in function of σ, following

$$\Delta t'_d = \begin{cases} 2\ \Delta t_d & \text{if } \sigma > 2 \\ \frac{\Delta t_d}{2} & \text{if } \sigma < 1 \\ \Delta t_d & \text{otherwise} \end{cases} \tag{31}$$

3.5 Multiresolution method

The principle in the multiresolution (MR) setting is to represent a set of function cell averages as values on a coarser grid plus a series of differences at different levels of nested grids. The differences contain the information of the function when going from a coarse to a finer grid.

A tree data structure is an efficient way to store the reduced MR dataas it allows to reduce the memory with respect to a finite volume (FV) scheme on the finest level. This representation could also increase the speed-up during the time evolution because it reduces the time-searching of the elements.

In the following we consider a hierarchy of regular grids in 2D, Ω_ℓ, $0 \le \ell \le L$. The root cell is $\Omega_{0,0,0} = \Omega$ and corresponds to a rectangle with side lengths h_x and h_y. The different node cells at a level $\ell > 0$ forming Ω_ℓ are denoted by $\Omega_{\ell,i,j}$ where $(i,j) \in \Lambda_\ell$. The ensemble of indices of the existing node cells on the level ℓ is Λ_ℓ. Note that $\Omega_{\ell,i,j}$ are rectangles with side lengths $h_{x,\ell} = 2^{-\ell} h_x$ and $h_{y,\ell} = 2^{-\ell} h_y$. In the tree terminology, the refinement of a parent node cell $\Omega_{\ell,i,j}$ at level ℓ produces four children nodes $\Omega_{\ell+1,2i,2j}$, $\Omega_{\ell+1,2i,2j+1}$, $\Omega_{\ell+1,2i+1,2j}$ and $\Omega_{\ell+1,2i+1,2j+1}$ at level $\ell+1$, as illustrated in Figure 1.

The cell-average value of the quantity u on the cell $\Omega_{\ell,i,j}$ is given by $\bar{u}_{\ell,i,j} = \frac{1}{|\Omega_{\ell,i,j}|}\int_{\Omega_{\ell,i,j}} u(x,y)\ dx\ dy$ and correspondingly the ensemble of the existing cell-

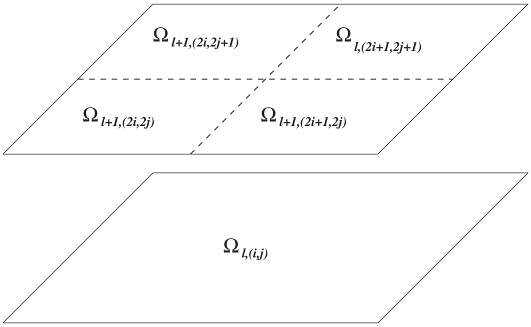

Figure 1: Dyadic grid refinement in 2D.

average values at level ℓ by $\bar{U}_\ell = (\bar{u}_{\ell,i,j})_{(i,j)\in\Lambda_\ell}$. The projection (or restriction) operator

$$P_{\ell+1\to l} \,:\, \bar{U}_{\ell+1} \,\mapsto\, \bar{U}_\ell.$$

estimates the cell-averages of a level ℓ from the ones of the level $\ell + 1$. The parent cell-average is the weighted average of the children cell-averages

$$\bar{u}_{\ell,i,j} = \frac{1}{4}(\bar{u}_{\ell+1,2i,2j} + \bar{u}_{\ell+1,2i,2j+1} + \bar{u}_{\ell+1,2i+1,2j} + \bar{u}_{\ell+1,2i+1,2j+1})$$

and thus the projection operator is exact and unique. To predict the cell-averages of a level $\ell + 1$ from the ones of the level ℓ, we use a prediction (or interpolation) operator

$$P_{\ell\to l+1} \,:\, \bar{U}_\ell \,\mapsto\, \widetilde{U}_{\ell+1}.$$

This operator yields an approximation $\widetilde{U}_{\ell+1}$ of $\bar{U}_{\ell+1}$ at the level $\ell + 1$. In this paper, we use third order interpolation given by a tensor product approach [9]. For $n, p \in \{0, 1\}$, we define

$$\begin{aligned}\widetilde{u}_{\ell+1,2i+n,2j+p} = \bar{u}_{\ell,i,j} &+ \frac{1}{8}(-1)^n\left(\bar{u}_{\ell,i+1,j} - \bar{u}_{\ell,i-1,j}\right)\\ &+ \frac{1}{8}(-1)^p\left(\bar{u}_{\ell,i,j+1} - \bar{u}_{\ell,i,j-1}\right)\\ &+ \frac{1}{64}(-1)^{np}\left[\left(\bar{u}_{\ell,i+1,j+1} - \bar{u}_{\ell,i+1,j-1}\right) - \left(\bar{u}_{\ell,i-1,j+1} - \bar{u}_{\ell,i-1,j-1}\right)\right].\end{aligned} \tag{32}$$

First, this prediction is local, since it is made from the cell average $\bar{u}_{\ell,i,j}$ and the eight nearest uncles $\bar{u}_{\ell,i\pm1,j\pm1}$. Second, it is consistent with the projection, i. e. $P_{\ell+1\to\ell} \circ P_{\ell\to\ell+1} = \mathrm{Id}$. The difference between the exact and the predicted values at three children cells yields the wavelet (or detail) coefficients. The sum of the four details in the children cells is equal to zero [9]

$$\bar{d}_{\ell+1,2i,2j+1} = \bar{u}_{\ell+1,2i,2j+1} - \widetilde{u}_{\ell+1,2i,2j+1}$$

$$\begin{aligned} \bar{d}_{\ell+1,2i+1,2j} &= \bar{u}_{\ell+1,2i+1,2j} - \tilde{u}_{\ell+1,2i+1,2j} \\ \bar{d}_{\ell+1,2i+1,2j+1} &= \bar{u}_{\ell+1,2i+1,2j+1} - \tilde{u}_{\ell+1,2i+1,2j+1}. \end{aligned} \tag{33}$$

This implies that the knowledge of the cell-average values on the children $\bar{U}_{\ell+1}$ is equivalent to the knowledge of the cell-average values on the parents $\bar{U}_\ell$ and the wavelet coefficients $\bar{D}_{\ell+1} = (\bar{d}_{\ell+1,2i,2j+1}, \bar{d}_{\ell+1,2i+1,2j}, \bar{d}_{\ell+1,2i+1,2j+1})_{(i,j)\in\Lambda_\ell}$. The so-called multiresolution transform on the cell-average values is obtained by repeating this operation recursively on L levels [36],

$$\bar{U}_L \longleftrightarrow (\bar{D}_L, \bar{D}_{L-1}, \ldots, \bar{D}_1, \bar{U}_0).$$

In conclusion, knowing the cell-average values of all the leaves $\bar{U}_L$ is equivalent to knowing the cell-average value of the root $\bar{U}_0$ and the details of all the other nodes of the tree structure.

In the MR scheme, instead of using the representation on the full uniform grid Ω_L, the numerical solution $\bar{U}^n_{MR} = \bar{U}^n_{L,MR}$ is formed by cell averages on an adaptive sparse grid $\Gamma^n = \Gamma^n_L$. Grid adaptivity in the MR scheme is related with an incomplete tree structure, where cell refinement may be interrupted at intermediate scale levels. This means that Γ^n is formed by *leaf cells* $\Omega_{\ell,i,j}$, $0 \leq \ell \leq L$, $(i, j) \in \mathcal{L}(\Lambda_\ell)$, which are cells without children. Here $\mathcal{L}(\Lambda_\ell)$ denotes the ensemble of indices for the existing leaf cells of the level ℓ.

Three basic steps are undertaken to evolve the solution from $\bar{U}^n_{MR}$ to $\bar{U}^{n+1}_{MR}$,

Refinement: $\bar{U}^{n+}_{MR} \leftarrow \mathbf{R}\bar{U}^n_{MR}$

Evolution: $\check{\bar{U}}^{n+1}_{MR} \leftarrow \mathbf{E}_{MR}\bar{U}^{n+}_{MR}$

Coarsening: $\bar{U}^{n+1}_{MR} \leftarrow \mathbf{T}(\epsilon)\check{\bar{U}}^{n+1}_{MR}$

The refinement operator $\mathbf{R}$ is a precautionary measure to account for possible translation or creation of finer scales in the solution between two subsequent time steps. As the regions of smoothness or irregularities of the solution may change with time, the grid Γ^n may not be convenient anymore at the next time step t^{n+1}. Hence, before doing the time evolution, the representation of the solution should be extended onto a grid Γ^{n+}, which is expected to be a refinement of Γ^n, and to contain Γ^{n+1}. Then, the time evolution operator $\mathbf{E}_{MR} = \mathbf{E}_{MR}(\Delta t)$ is applied. The subscript MR in E_{MR} means that only the cell-averages on the leaves of the computational grid Γ^{n+} are evolved in time, and that an adaptive flux computation $F_{MR}(\bar{U}^{n+}_{MR})$ is adopted at interfaces of cells of different scale levels. Finally, a thresholding operation $\mathbf{T}(\epsilon)$ (coarsening) is applied in order to unrefine those cells in Γ^{n+} that are unnecessary for an accurate representation of $\check{\bar{U}}^{n+1}_{MR}$. The choice of the threshold value is motivated by an error analysis equilibrating the perturbation and discretization errors. For details we refer to [17, 48].

To compress data in an adaptive tree structure, while still being able to navigate through it, *gradedness* is required. For instance, for a given node in the dynamic tree structure we assume that:

- its parent and eight nearest uncles are in the tree (if not, we create them as nodes);
- for flux computations, if $\Omega_{\ell,i,j}$ is a leaf, its four nearest cousins $\Omega_{\ell,i\pm2,j}$ and $\Omega_{\ell,i,j\pm2}$ in each direction are in the tree (if not, we create them as virtual leaves);
- if a child is created, all its brothers are also created;

For more details of these procedures, we refer to [48].

In the tree structure, the thresholding operator $\mathbf{T}(\epsilon)$ is defined by removing leaves where details are smaller than a prescribed tolerance ϵ, while preserving the gradedness property, and the refinement operation $\mathbf{R}$ adds one more level as security zone, in order to forecast the evolution of the solution in the tree representation at the next time step. These two operations are performed by the following procedure.

We denote by Λ the ensemble of indices of the existing tree nodes in Γ^{n+}, by $\mathcal{L}(\Lambda)$ the restriction of Λ onto the leaves, and by Λ_ℓ the restriction of Λ to a level ℓ, $0 \leq \ell < L$. For the whole tree, from the leaves to the root:

- Compute the details on the nodes $\bar{d}_{\ell,i,j}$, $(i,j) \in \Lambda_{\ell-1}$, by the multiresolution transform;
- Define the deletable cells, if the details on the corresponding nodes and their brothers are smaller than the prescribed tolerance.

For the whole tree, from the leaves to the root:

- If a node and its children nodes are deletable, and the children nodes are simple leaves (without virtual children), then delete their children.
- If the node and its parents are not deletable, and it is not at the maximum level, then create the children for this node.

To illustrate the adaptive flux computation, we consider the leaf $\Omega_{\ell+1,2i+1,2j}$, sharing an interface with another leaf $\Omega_{\ell,i+1,j}$ at a lower scale level, as illustrated in Figure 2. For the calculation of the outgoing numerical flux on the right interface, we use the cell width in the x direction $h_{x,\ell+1}$ as step size. The required right neighboring stencils are obtained from the cousins $\Omega_{\ell+1,2i+2,2j}$ and $\Omega_{\ell+1,2i+3,2j+1}$, which are virtual cells. For conservation, the ingoing flux on the leaf $\Omega_{\ell,i+1,j}$ is set equal

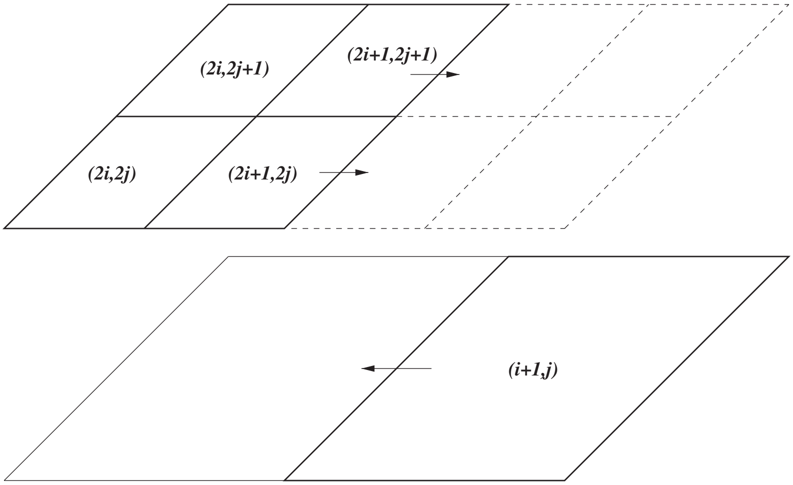

Figure 2: Adaptive numerical flux computation in 2D.

to the sum of the outgoing fluxes on the neighbour leaves of level $\ell + 1$. For more details on the implementation of this procedure we refer to [48].

4 Numerical results

In all numerical simulations, we consider an initially planar Chapman-Jouguet detonation moving with constant unitary speed through the unburnt gas to the right of the domain. Setting $\gamma = 1.4$, $r = 1$, $Q_0 = 1$ and $T_i = 0.22$ and assigning the following burnt gas values

$$\rho_b = \gamma \ , \ v_{x,b} = \delta \ , \ p_b = 1 \ , Z_b = 0 \, , \tag{34}$$

where

$$\delta = \sqrt{\frac{2(\gamma - 1)}{\gamma + 1}} \, , \tag{35}$$

the corresponding von Neumann state past the shock wave is [37, 50]

$$\rho_N = \frac{\gamma}{1 - \delta} \ , \ v_{x,N} = 2\delta \ , \ p_N = 1 + \gamma\delta \ , Z_N = 1 \, , \tag{36}$$

and the unburnt gas values are

$$\rho_u = \frac{\gamma}{1 + \delta} \ , \ v_{x,u} = 0 \ , \ p_u = 1 - \gamma\delta \ , Z_u = 1 \, . \tag{37}$$

Table 3: Chapman-Jouguet detonation front: CPU time, CPU compression and relative error on the von Neumann density for different scales and for the FV and MR methods, $\tau = 10^{-1}$.

Method	L	ϵ	CPU time	CPU compression	relative error
MR	9	10^{-2}	1 s	41.34%	1.099×10^{-1}
FV	9		3 s	100.00%	1.109×10^{-1}
MR	10	5×10^{-3}	4 s	34.28%	6.519×10^{-2}
FV	10		13 s	100.00%	6.601×10^{-2}
MR	11	2.5×10^{-3}	11 s	21.86%	3.554×10^{-2}
FV	11		54 s	100.00%	3.617×10^{-2}
MR	12	1.25×10^{-3}	40 s	18.75%	1.740×10^{-2}
FV	12		3 min 37 s	100.00%	1.783×10^{-2}

The resulting temperature of the unburnt gas $T_u = 0.215995$ is only slightly lower than the chosen ignition temperature T_i. The initial condition for an initial front at $x = x_0$ is

$$q(x,0) = \begin{cases} (q_N - q_b)\exp\left[a(x - x_0)\right] + q_B & \text{if} \quad x \le x_0 \\ q_u & \text{if} \quad x > x_0 \end{cases} \tag{38}$$

where a is set to $\frac{1}{\tau}$. Here q stands for ρ, v_x, p, and Z, while $v_y = 0$ everywhere. This way the initial condition is close to the classical Zeldovich-von Neumann-Dring (ZND) solution of the reaction Euler equations.

4.1 One-dimensional detonation

First we consider a one-dimensional setting. The computational domain is $\Omega = [-3, 1]$. In this case, the location of the initial front is set to $x_0 = -2$. In order to follow the detonation wave and perform longer computations, a change of variables is made for v_x. We set $v_x' = v_x - \delta$ and omit the superscript $'$ everywhere. Since the initial transition happens in the segment $[-3, -1]$, we only focus on the domain $[-1, 1]$. The dimensionless final time is $t = 2.2$.

4.1.1 Non-stiff case

In this first part, we consider a detonation with a time coefficient $\tau = 10^{-1}$. As observed in Figure 3, the detonation front propagates from left to right with constant maximal values of density, pressure and velocity, which correspond to the von Neumann state. The slopes for the density, pressure, and velocity, are moder-

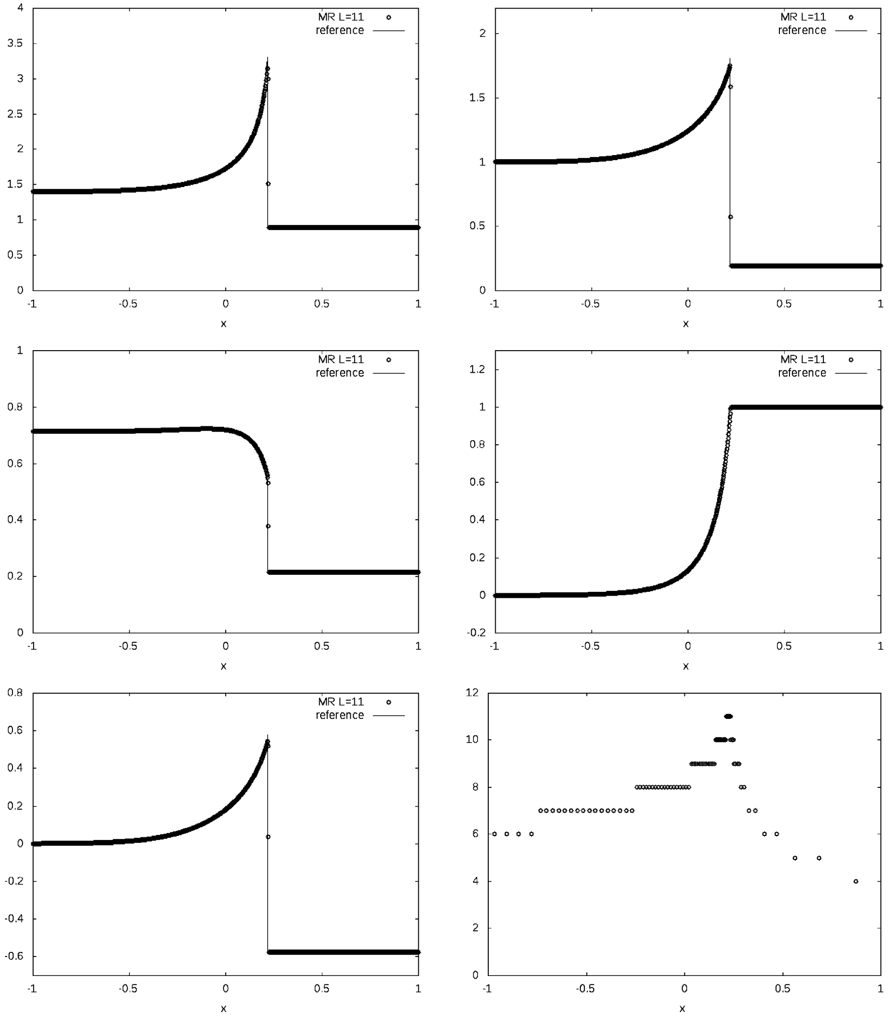

Figure 3: Chapman-Jouguet detonation front: density (*top, left*), pressure (*top, right*), temperature (*middle, left*), partial mass of the limiting reactant (*middle, right*), velocity (*bottom, left*) and adaptive mesh (*bottom, right*) at $t = 2.2$, $L = 11$ scales, $\epsilon = 2.5 \times 10^{-3}$, $\tau = 10^{-1}$.

ate on the left side of the shock, i. e. in the burnt gas zone, but become sharp on the right side, at the interface with the unburnt gases. For all the displayed quantities, the curves fit with the reference computation, which has been performed on a fine grid with $L = 14$ scales, i. e. with 16 384 grid points.

Table 3 assembles results for computations performed with different maximum number of scales for both the MR and the FV method. The relative error is

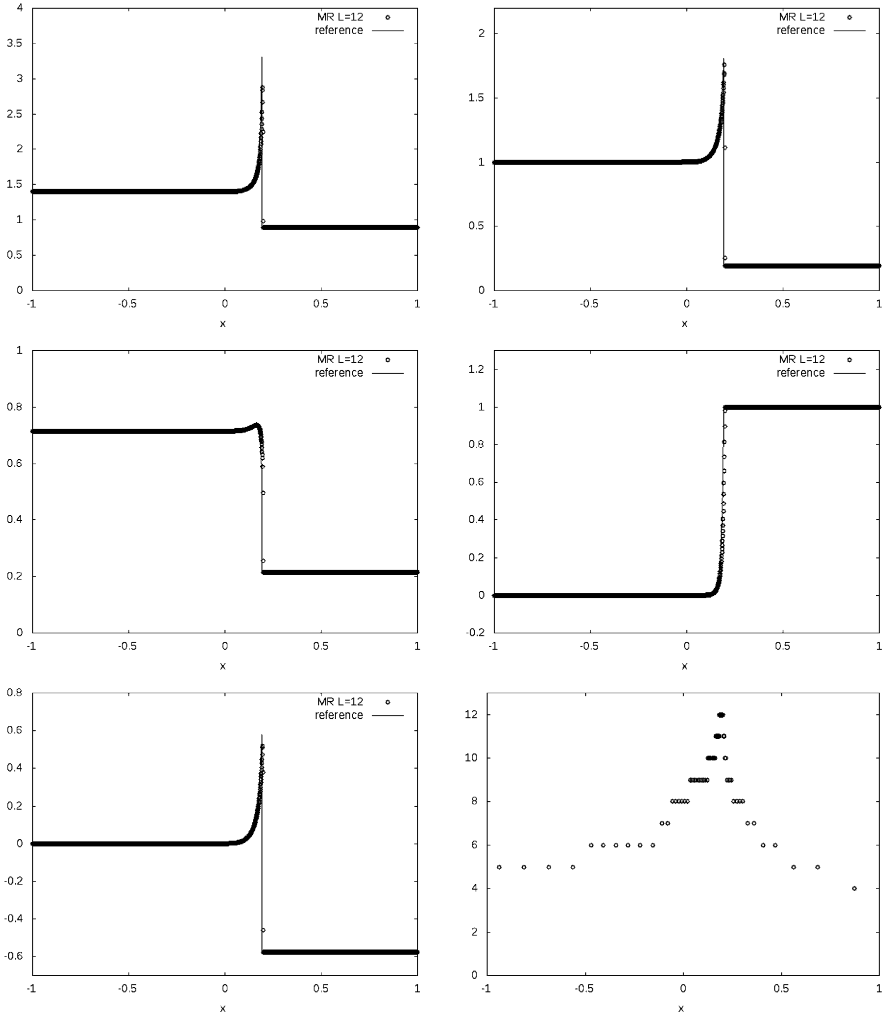

Figure 4: Chapman-Jouguet detonation front: density (*top, left*), pressure (*top, right*), temperature (*middle, left*), partial mass of the limiting reactant (*middle, right*), velocity (*bottom, left*) and adaptive mesh (*bottom, right*) at $t = 2.2$, $L = 12$ scales, $\epsilon = 1.25 \times 10^{-3}$, $\tau = 10^{-2}$.

computed on the von Neumann value of the density. The numerical value is averaged in time to damp its oscillations and is compared to the theoretical one. We observe for the MR computations a CPU time compression rate growing with increasing number of scales, while the relative error remains comparable with the one of the same computation on the regular fine grid. As expected, the error is reduced when increasing the number of scales L.

Table 4: Chapman-Jouguet detonation front: CPU time, CPU compression and relative error on the von Neumann density for different values of the stiffness coefficient and for the FV and MR methods.

Method	τ	L	ϵ	CPU time	CPU comp.	rel. error
MR	10^{-2}	12	1.25×10^{-3}	43 s	20.08%	1.336×10^{-1}
FV	10^{-2}	12		3 min 37 s	100.00%	1.342×10^{-1}
MR	10^{-3}	13	6.25×10^{-4}	2 min 03 s	7.56%	3.057×10^{-1}
FV	10^{-3}	13		27 min 13 s	100.00%	3.057×10^{-1}

4.1.2 Influence of the stiffness

In this part, we perform computations of stiff problems and set τ first to 10^{-2}, then to 10^{-3}. In order to resolve correctly the detonation front, 13 scales are required for the case $\tau = 10^{-3}$. The results are shown in Figures 4 and 5. We observe a much sharper and thinner reaction zone, reduced to a few computational points in the case $\tau = 10^{-3}$. We observe a good agreement with the reference computations, which were computed on a regular fine grid with $L = 15$ scales.

Table 4 gives the results of the computations for $\tau = 10^{-2}$ and $\tau = 10^{-3}$. As expected, a high compression rate is reached when 13 scales are used. But as it is seen in Figure 5, the error on the von Neumann density remains quite large, even if the detonation front is well tracked.

4.2 Two-dimensional detonation

Now we consider a two-dimensional problem. The interaction between a detonation wave and a pocket of unburnt gas is simulated. The computational domain is $[0, 4] \times [-1, 1]$, the initial planar detonation front is located at $x = 1.7$ and the initial spot of unburnt gas is centered in $(x_0, y_0) = (2, 0)$. The radius of the circular pocket is $r_0 = 0.1$. The parameters of the detonation front are the same as in the non-stiff case, i. e. we choose $\tau = 10^{-1}$. The dimensionless final time is $t = 1.3$.

In Figure 6, we observe the destabilization of the planar front by the pocket of unburnt gas. We remark circular structures due to the expansion of the pocket in the three directions and their interaction with the detonation wave going from left to right. Recirculations are observed in the center of the pocket, which are advected by the flow. We also observe reflexions of the waves on the free-slip boundaries in $y = -1$ and $y = 1$.

On the right side of Figure 6, we observe that the adaptive mesh follows well the structures of the different waves that interact in the flow. A zoom in the center

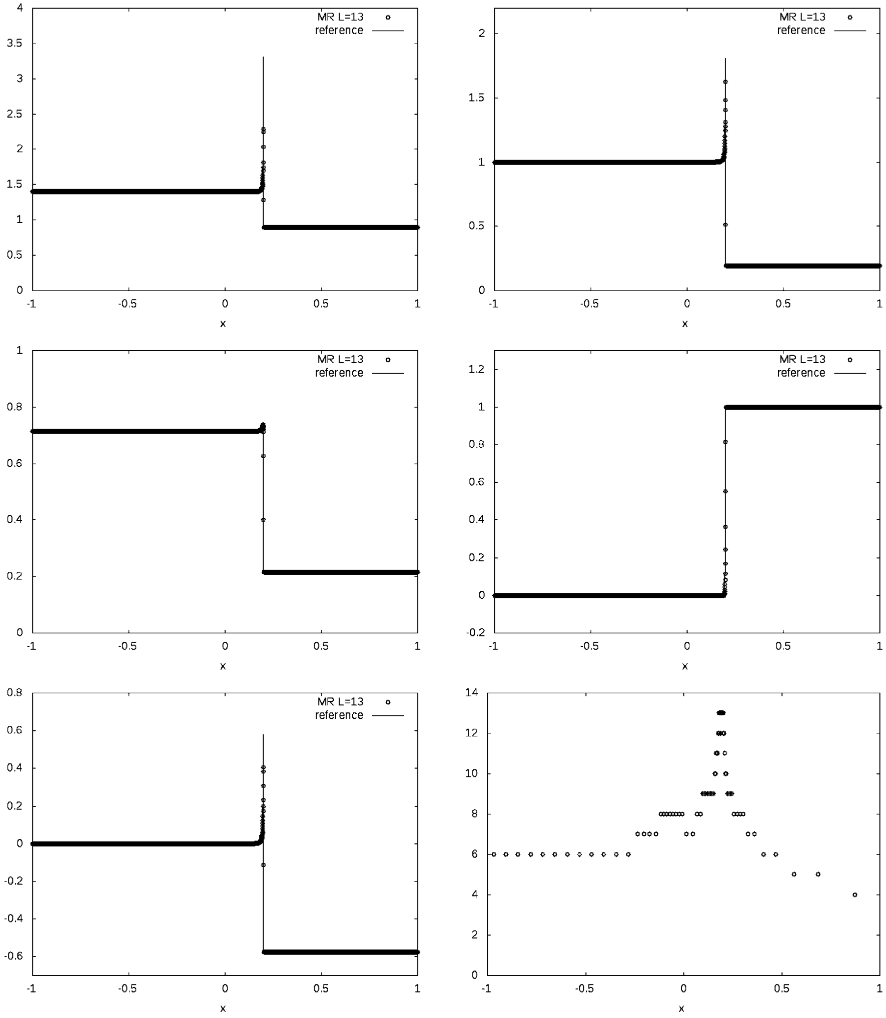

Figure 5: Chapman-Jouguet detonation front: density (*top, left*), pressure (*top, right*), temperature (*middle, left*), partial mass of the limiting reactant (*middle, right*), velocity (*bottom, left*) and adaptive mesh (*bottom, right*) at $t = 2.2$, $L = 13$ scales, $\epsilon = 6.25 \times 10^{-4}$, $\tau = 10^{-3}$.

of the mesh (Figure 7) enables to see how well the mesh adapts to all the structures of the flow. The computation, performed on $L = 10$ scales, i. e., a maximum of $2^{2L} = 1\,048\,576$ points, requires 2 h 59 min of CPU time on a 16 Core PC. The same computation on the regular fine grid lasts 14 h 35 min. Hence the CPU time compression is around 20 %.

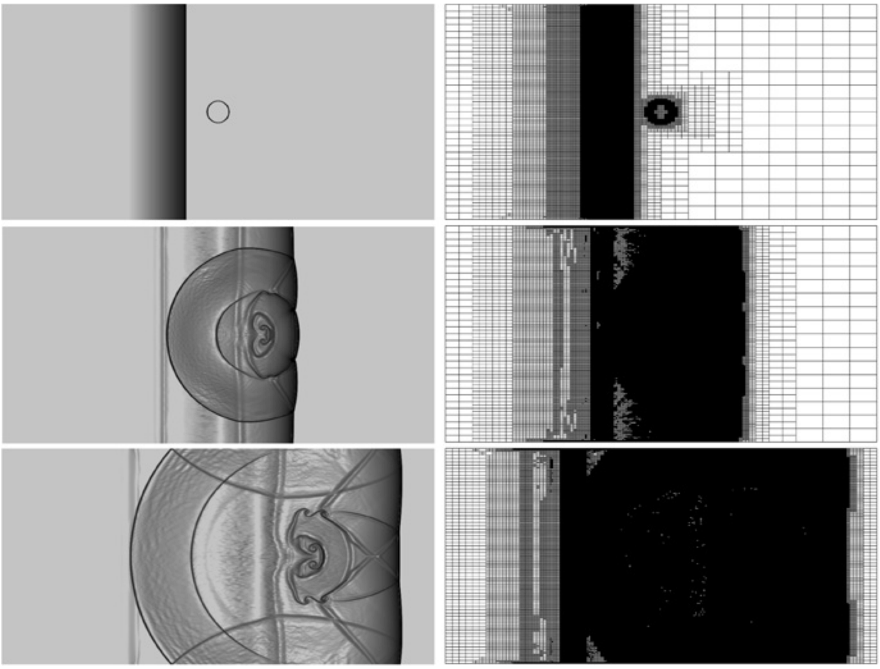

Figure 6: Time evolution of the interaction between a detonation wave an a pocket of unburnt gas computed with the adaptive MR method. Numerical Schlieren of the density gradient (*left*) and corresponding meshes (*right*) at $t = 0$ (*top*), $t = 0.65$ (*middle*) and $t = 1.3$ (*bottom*).

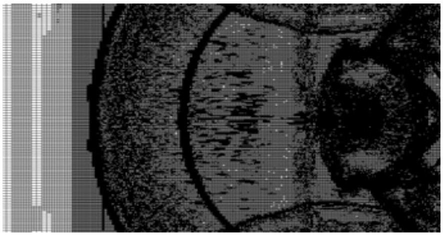

Figure 7: Adaptive MR computation of the interaction between a detonation wave an a pocket of unburnt gas: Zoom on the adaptive mesh at $t = 1.3$.

Figure 8 shows a cut of the density at $y = 0$. We observe an excellent agreement between the MR and FV computations with $L = 10$ scales, and a good agreement of both curves with the FV reference computation obtained with $L = 11$ scales, which validates the grid convergence of the computation. The total number of grid points in the FV computations is 2^{2L}.

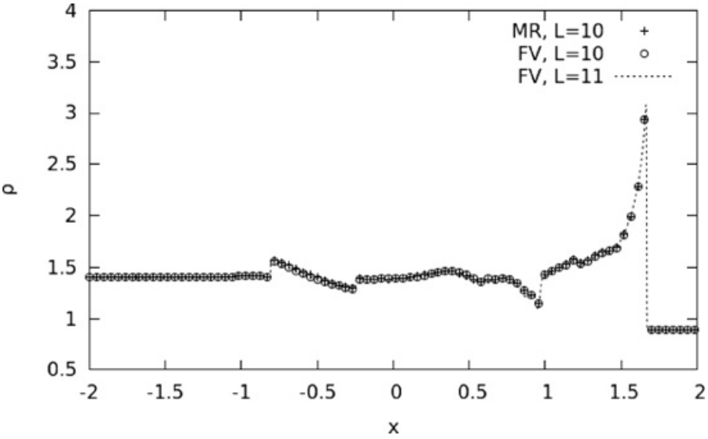

Figure 8: Interaction between a detonation wave an a pocket of unburnt gas. One-dimensional cuts of density for $y = 0$ at $t = 1.3$. Comparison of the adaptive MR and the FV computations for $L = 10$ with the FV reference computation for $L = 11$. The total number of grid points in the FV computations is 2^{2L}.

5 Conclusion and perspectives

We presented an extension of the adaptive multiresolution method for the reactive Euler equations, able to deal with fast chemical reactions. A finite volume scheme with second order shock capturing schemes is used for space discretization. Classical Strang splitting is applied to deal with the stiffness of the physical problem in time. Space and time adaptivity are then introduced using multiresolution analysis and Runge–Kutta–Fehlberg schemes, respectively. The implementation uses dynamic memory allocation with tree data structures.

Applications to detonation problems in one and two space dimensions validated the algorithm and illustrated the efficiency of the adaption strategy. The adaptive computations yield both speed-up of CPU time and memory compression with respect to uniform grid computations while the precision is automatically controlled using suitable thresholding techniques.

As perspective we plan the extension of the method to three space dimensions and also to include multi-species chemical reactions and viscous effects. The parallelization of the adaptive method using tree data structures to handle the adaptive grid remains a challenging task for the near future.

Acknowledgement: This article is dedicated to Professor Henning Bockhorn on the occasion of his 70th birthday, thanking him cordially for the great time we had in his group, first in Kaiserslautern and then in Karlsruhe. We also thank Margarete Domingues, Ralf Deiterding and Sonia Gomes for constructive discussions on the topic and fruitful interactions. KS thankfully acknowledges financial support from the ANR project SiCoMHD (ANR-Blanc 2011-045) and OR from the ANR project MAPIE (ANR-13-MONU-0002).

References

1. R. Abgrall and A. Harten, SIAM J. Numer. Anal. **35** (1998) 2128.
2. I. Babuska and W. C. Rheinboldt, Int. J. Numer. Meth. Engng. **12** (1978) 1597.
3. R. Becker and R. Rannacher, An optimal control approach to a-posteriori error estimation, in: *Acta Numerica*, edited by: A. Iserles, R. Becker, R. Rannacher, and P. G. Ciarlet, Cambridge University Press, Cambridge, UK, 1–102 (2001).
4. J. Bell, M. Berger, J. Saltzman, and M. Welcome, SIAM J. Sci. Comput. **15** (1994) 127.
5. L. Bentaleb, O. Roussel, and C. Tenaud, Adaptive multiresolution methods for the simulation of shocks/shear layer interaction in confined flows, in: *Numerical Mathematics and Advanced Applications*, Springer, Berlin, 761–769, (2006).
6. M. Berger and P. Colella, J. Comput. Phys. **82** (1988) 64.
7. M. Berger and J. Oliger, J. Comput. Phys. **53** (1984) 484.
8. B. L. Bihari, J. Comput. Phys. **123** (1997) 207.
9. B. L. Bihari and A. Harten, SIAM J. Sci. Comput. **18** (1996) 315.
10. M. Braack and Th. Richter, Int. J. Numer. Meth. Fl. **51** (2006) 981.
11. M. Braack and Th Richter, Solving multidimensional reactive flow problems with adaptive finite elements, in: *In Reactive Flows, Diffusion and Transport*, edited by: W. Jäger, R. Rannacher and J. Warnatz, Springer, Berlin, Heidelberg, 93–112 (2007).
12. A. Brandt, Math. Comp. **31** (1977) 333.
13. J. D. Calle, P. R. Devloo, and S. M. Gomes, Numer. Algor. **39** (2005) 143.
14. G. Chiavassa and R. Donat, SIAM J. Sci. Comput. **23** (2001) 805.
15. A. Cohen, *Wavelet methods in numerical analysis*. Elsevier, Amsterdam (2000).
16. A. Cohen, N. Dyn, S. M. Kaber, and M. Postel, J. Comput. Phys. **161** (2000) 264.
17. A. Cohen, S. M. Kaber, S. Müller, and M. Postel, Math. Comp. **72** (2003) 183.
18. R. Courant and K. O. Friedrichs, *Supersonic Flow and Shock Waves*, Springer-Verlag, New York (1985).
19. W. Dahmen, B. Gottschlich-Müller, and S. Müller, Numer. Math. **88** (2001) 399.
20. S. Descombes, M. Duarte, T. Dumont, V. Louvet, and M. Massot, Confluentes Mathematici, Dedicated to Michelle Schatzman **3** (2011) 413.
21. S. Descombes, M. Duarte, T. Dumont, F. Laurent, V. Louvet, and M. Massot, SIAM J. Numer. Anal. **52** (2014) 1311.
22. M. O. Domingues, S. M. Gomes, and L. M. A Diaz, Appl. Num. Math. **47** (2003) 421.
23. M. O. Domingues, S. M. Gomes, O. Roussel, and K. Schneider, J. Comput. Phys. **227** (2008) 3758.
24. M. O. Domingues, S. M. Gomes, O. Roussel, and K. Schneider, Appl. Num. Math. **59** (2009) 2303.
25. M. O. Domingues, O. Roussel, and K. Schneider, Int. J. Numer. Meth. Engng. **78** (2009) 652.

26. J. R. Dormand and P. J. Prince, J. Comput. Appl. Math. **6** (1980) 19.
27. M. Duarte, *Adaptive numerical methods in time and space for the simulation of multi-scale reaction fronts*, Ph.D. Thesis, Ecole Centrale Paris, Paris (2011).
28. M. Duarte, M. Massot, S. Descombes, C. Tenaud, T. Dumont, V. Louvet, and F. Laurent, SIAM J. Sci. Comput. **34** (2012) 76.
29. M. Duarte, M. Massot and S. Descombes, ESAIM-Math. Model. Num. **45** (2011) 825.
30. M. Duarte, S. Descombes, C. Tenaud, S. Candel and M. Massot, Combust. Flame **160** (2013) 1083.
31. M. Duarte, Z. Bonaventura, M. Massot, A. Bourdon, S. Descombes and T. Dumont, J. Comput. Phys. **231** (2012) 1002.
32. E. Fehlberg, Z. Angew. Math. Mech. **44** (1964) 17.
33. P. C. Franzone, P. Deuflhard, B. Erdmann, J. Lang, and L. F. Pavarino, SIAM J. Sci. Comput. **28** (2006) 942 .
34. E. Godlewski and P. A. Raviart, *Numerical Approximation of Hyperbolic Systems of Conservation Laws*, Springer-Verlag, New York (1995).
35. B. Gottschlich-Müller and S. Müller, Adaptive finite volume schemes for conservation laws based on local multiresolution techniques, in: *Hyperbolic Problems: Theory, Numerics, Applications*, edited by: R. Jeltsch and M. Frey, Vol. 129. ISNM, Inter. Ser. Numer. Math., Birkhäuser, Boston (1999).
36. A. Harten, Comm. Pure Appl. Math. **48** (1995) 1305.
37. C. Helzel, R. J. LeVeque, and G. Warnecke, SIAM J. Sci. Comput. **22** (2000) 1489.
38. M. Holmström, SIAM J. Sci. Comput. **21** (1999) 405.
39. J. M. Hyman, S. Li, and L. R. Petzold, Comput. Math. Appl. **46** (2003) 1511.
40. M. Kaibara and S. M. Gomes, A fully adaptive multiresolution scheme for shock computations, in: *Godunov Methods: Theory and Applications*, edited by: E. F. Toro, Klumer Academic/Plenum Publishers, New York, p. 497 (2001).
41. B. Koren, A robust upwind discretisation method for advection, diffusion and source terms, in: *Numerical Methods for Advection-Diffusion Problems*, edited by: C. B. Vreugdenhil and B. Koren, Vieweg, Braunschweig, p. 117, (1993).
42. J. Lang, Appl. Num. Math. **18** (1995) 223.
43. M. S. Liou, J. Comput.Phys. **129** (1996) 364.
44. S. Müller, *Adaptive multiscale schemes for conservation laws*, Vol. 27 of Lectures Notes in Computational Science and Engineering. Springer, Heidelberg (2003).
45. S. Müller and Y. Stiriba, J. Sci. Comput. **30** (2007) 493.
46. P. Pinho, M. O Domingues, P. J. Ferreira, S. M. Gomes, A. Gomide, and J. R. Pereira, IEEE T. Magn. **43** (2007) 1013.
47. O. Roussel and K. Schneider, Comp. Fluids **34** (2005) 817.
48. O. Roussel, K. Schneider, A. Tsigulin, and H. Bockhorn, J. Comput. Phys. **188** (2003) 493.
49. O.Roussel and K. Schneider, J. Comput. Phys. **229** (2010) 2267.
50. L. Tosatto and L. Vigevano, J. Comput. Phys. **227** (2008) 2317.
51. B. Van Leer, J. Comput. Phys. **32** (1979) 101.