Interpolated pressure law in two-fluid simulations and hyperbolicity

Philippe Helluy, Jonathan Jung

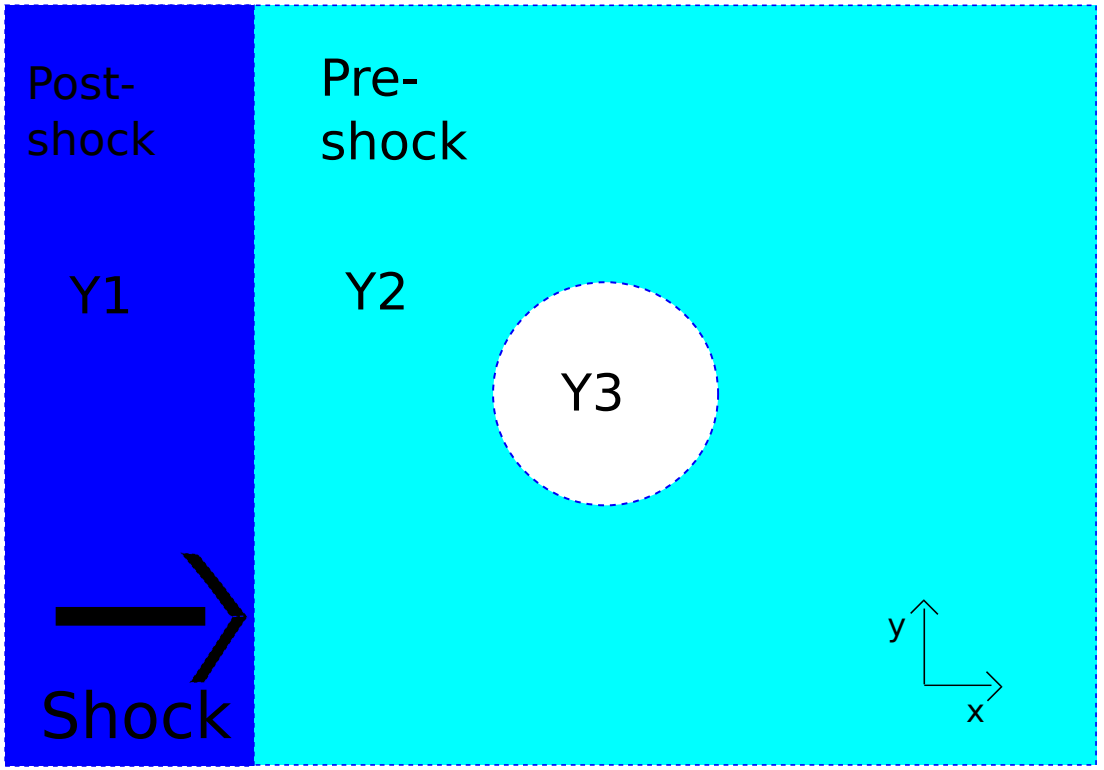
IRMA Strasbourg

DFG-CNRS workshop. February 2014

Outlines

Physical problem

Shock-bubble or shock-droplet interaction



Systems of conservation laws

The unknown is a vector $W(x,t) \in \mathbb{R}^m$ that depends on space $X \in \mathbb{R}^d$ $(d = 2)$ and time $t \in \mathbb{R}$ and satisfies

$$\partial_t W + \nabla_X \cdot F(W) = 0.$$

The flux F is supposed to be hyperbolic in all the directions $n \in \mathbb{R}^d$:

$$A(W,n) := \nabla_W F(W) \cdot n$$

is diagonalizable with real eigenvalues.

- The solutions are complex: shock waves, non uniqueness, turbulence, *etc.*
- Needs of precise and robust numerical methods.
- Algorithms must be adapted to multicore computers.

Compressible two-fluid model

Vector of conservative variables $W = (\rho, \rho u, \rho v, \rho Q, \rho \varphi)^T$, where

- ρ is the density,
- $U = (u, v)^T$ is the velocity,
- Q is the total energy,
- φ the color function ($\varphi = 0$ in the liquid and $\varphi = 1$ in the gas).
- The internal energy is $e = Q - u^2/2$.
- The pressure is defined by $p = p(\rho, e, \varphi)$.
- The flux is given by

$$F(w) \cdot n = (\rho U \cdot n, \rho (U \cdot n) U^T + p n^T, (\rho Q + p) U \cdot n, \rho \varphi U \cdot n)^T.$$

Diffusion of the color function

The color function is a weak solution of

$$\partial_t \varphi + U \cdot \nabla \varphi = 0,$$

thus

$$\forall (x,t) \quad \varphi(x,t) \in \{0,1\}.$$

However most numerical schemes will produce numerical diffusion and we have to interpolate the pressure law $p(\rho, e, \varphi)$ for

$$\varphi \in]0,1[.$$

Pressure law

For instance, we can consider a simple model made of two stiffened gases [AS99]. gas=(1), liquid=(2).

$$p(\rho, e, \varphi) = (\gamma(\varphi) - 1)\rho e - \gamma(\varphi)\pi(\varphi),$$
$$\frac{1}{\gamma(\varphi) - 1} = \varphi \frac{1}{\gamma_1 - 1} + (1 - \varphi)\frac{1}{\gamma_2 - 1},$$
$$\frac{\gamma(\varphi)\pi(\varphi)}{\gamma(\varphi) - 1} = \varphi \frac{\gamma_1 \pi_1}{\gamma_1 - 1} + (1 - \varphi)\frac{\gamma_2 \pi_2}{\gamma_2 - 1}.$$

$$\gamma_1 = 1.4, \quad \gamma_2 = 3, \quad \pi_1 = 0, \quad \pi_2 = 8500 \times 10^5 \text{ Pa}.$$

Hyperbolicity

Let

$$\mathscr{W}_{ad}(\varphi) = \left\{ W = (\rho, \rho u, \rho v, \rho Q, \rho \varphi)^T \in \mathbb{R}^m, \quad \rho > 0, \quad p + \pi(\varphi) > 0 \right\},$$

and

$$\mathscr{W}_{ad} = \bigcup_{\varphi \in [0,1]} \mathscr{W}_{ad}(\varphi).$$

The system is hyperbolic with eigenvalues $U \cdot n - c, U \cdot n, U \cdot n, U \cdot n, U \cdot n + c$ and $c = \sqrt{\gamma(p+\pi)/\rho}$, if $W \in \mathscr{W}_{ad}$.
For a given φ, the set $\mathscr{W}_{ad}(\varphi)$ is convex. But $\mathscr{W}_{ad}$ is not convex.

Riemann solver

First we consider the 1D framework $X = (x, y)^T$, $W = W(x, t)$, $n = (1, 0)^T$,

$$\partial_t W + \partial_x (F(W) \cdot n) = 0.$$

Let V_L and V_R be two constant states in $\mathscr{W}_{ad}$. We can prove that the Riemann problem

$$\partial_t V + \partial_x (F(V) \cdot n) = 0$$

$$V(x, 0) = \begin{cases} V_L \text{ if } x < 0, \\ V_R \text{ if } x \geq 0, \end{cases}$$

admits a unique global entropy solution [BHR04], which is denoted by

$$R(V_L, V_R, x/t) = V(x, t) \in \mathscr{W}_{ad}.$$

The function R is called the Riemann solver.

Mesh

- We consider a 1D mesh made, at the time step n, of cells $C_i^n =]x_{i-1/2}^n, x_{i+1/2}^n[$, $i \in \mathbb{Z}$. The size of the cell C_i^n is $h_i^n = x_{i+1/2}^n - x_{i-1/2}^n$.
- We also consider a time step τ_n satisfying a CFL condition and a sequence of times t_n satisfying $t_{n+1} = t_n + \tau_n$.
- The solution $W(x,t)$ is approximated in each cell by a constant value

$$W_i^n \simeq W(x, t_n), \quad x \in C_i^n.$$

Standard Godunov scheme

In the standard Godunov scheme the mesh is constant with respect to time: $x_{i+1/2}^n = x_{i+1/2}^{n+1}$. The scheme reads

$$h_i^n W_i^{n+1} - h_i^n W_i^n + \tau_n \left(F_{i+1/2}^n - F_{i-1/2}^n \right) = 0.$$

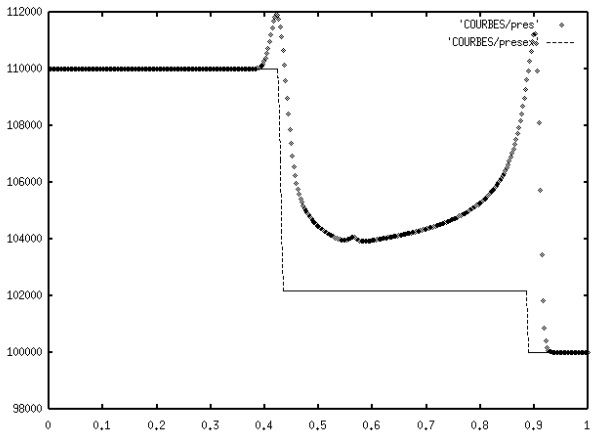
The numerical flux is defined from the Riemann solver $R(W_L, W_R, x/t)$ and

$$\begin{aligned} F_{i+1/2}^n &= F(W_{i+1/2}^n), \\ W_{i+1/2}^n &= R(W_i^n, W_{i+1}^n, u_{i+1/2}^n). \end{aligned}$$

Classic construction: (1) exact resolution of interface Riemann problems. (2) averaging over the cells. CFL condition and entropy stability follow from Jensen inequality.

Pressure "oscillations"

Problem: $\mathscr{W}_{ad}$ is generally no convex. The Godunov scheme is not stable and may fail after only one time step [MHB10].
Even when the computations are possible, the results are not accurate (spurious pressure "oscillations").



Better accuracy with the non-conservative scheme of Abgrall-Saurel [AS99], but with the same stability issue [Jung13].

Possible cures

We can:

1. Construct another pressure law that ensures convexity of $\mathscr{W}_{ad}$.
2. Construct another scheme that keeps W_i^n in $\mathscr{W}_{ad}$.

Lax entropy

Construction of the Lax entropies for a single fluid flow [Cro90, HLLM98] with general pressure ($\varphi = 1$ and $p = p(\rho, e, 1)$.)

- We consider a concave PH1 function $(V, E, M) \in \mathbb{R}^3 \mapsto S(V, E, M) \in \mathbb{R} \cup \{-\infty\}$: the extensive entropy. C^2 on its (convex) domain $\mathrm{dom} S = \{(V, E, M) \in \mathbb{R}^3, S(V, E, M) > -\infty\} \subset \{V, E, M \geq 0\}$.
- We define $\rho = M/V$, $\tau = V/M$, $e = E/M$ and the specific entropy $s(\tau, e) = S(\tau, e, 1)$.
- We define the temperature $T = \partial_e s(\tau, e)$, the pressure $p = T \partial_\tau s(\tau, e)$.
- Theorem: $(\rho, \rho U^T, \rho Q) \mapsto -\rho s(\tau, e)$ is a Lax entropy of the Euler equations.

Proof

- Lemma 1 (Croisille 91): $(x_0, x_1 \cdots x_n) \mapsto F(x_0, x_1 \cdots x_n)$ convex and PH1. $\mathrm{sgn} F'' = (0, 1, n-1)$ iff $(x_1 \cdots x_n) \mapsto F(1, x_1 \cdots x_n)$ is strictly convex.
- Lemma 2: if $Tds = de + pd\tau$ then Euler $\Rightarrow$additional conservation law $\partial_t(\rho s) + \nabla \cdot (\rho s) = 0$.
- $S(V, E, M) = VS(1, \rho e, \rho) = MS(\tau, e, 1) \Rightarrow \rho s(\tau, e) = S(1, \rho e, \rho)$ thus ρs is concave with respect to $(\rho, \rho e)$.
- Lemma 3: $(\rho, \rho U, \rho Q) \mapsto -\rho s(\tau, e)$ is strictly convex (if $T > 0$) and thus a Lax entropy.
- Mock's theorem $\Rightarrow$the Euler equations are hyperbolic on the convex domain of the Lax entropy.

Generalization

Two-fluid model with a general pressure law $p = p(\rho, e, \varphi)$

- We consider a concave PH1 function $(V, E, M, M_1) \in \mathbb{R}^4 \mapsto S(V, E, M, M_1) \in \mathbb{R} \cup \{-\infty\}$: the extensive entropy. C^2 on its (convex) domain $\mathrm{dom} S = \{(V, E, M, M_1) \in \mathbb{R}^4, S(V, E, M, M_1) > -\infty\}$.
- We define $\rho = M/V$, $\tau = V/M$, $e = E/M$, $\varphi = M_1/M$ and the specific entropy $s(\tau, e, \varphi) = S(\tau, e, 1, \varphi)$.
- We define the temperature $T = \partial_e s$, the pressure $p = T\partial_\tau s$ and the potential $\lambda = T\partial_\varphi s$.
- Theorem: $(\rho, \rho U^T, \rho Q, \rho\varphi) \mapsto -\rho s(\tau, e, \varphi)$ is a Lax entropy of the two-fluid model.

Proof

- Lemma 2': if $Tds = de + pd\tau + \lambda d\varphi$ then the two-fluid model $\Rightarrow$additional conservation law $\partial_t \rho s + \nabla \cdot (\rho s) = 0$.
- $S(V, E, M, M_1) = VS(1, \rho e, \rho, \rho\varphi) = MS(\tau, e, 1, \varphi) \Rightarrow \rho s(\tau, e, \varphi) = S(1, \rho e, \rho, \rho\varphi)$ thus ρs is concave with respect to $(\rho, \rho e, \rho\varphi)$.
- Lemma 3': $(\rho, \rho U, \rho Q, \rho\varphi) \mapsto -\rho s(\tau, e, \varphi)$ is strictly convex and thus a Lax entropy (if $T > 0$).
- Mock's theorem $\Rightarrow$the two-fluid model is hyperbolic on the convex domain of the Lax entropy.

Mixture pressure law

How to construct $S(V,E,M,M_1)$? Of course: entropy optimization [HM11]!

$$S(V,E,M,M_1) = \sup_{V_1,E_1} S_1(V_1,E_1,M_1) + S_2(V-V_1,E-E_1,M-M_1).$$

- by construction, S is concave and PH1.
- no optimization with respect to M_1: no phase transition.
- what happens with a mixture of a perfect gas and a stiffened gas ?

$$S_1(\tau,e,1) = (\gamma_1 - 1)\ln\tau + \chi_1 \ln e,$$

$$S_2(\tau,e,1) = (\gamma_2 - 1)\ln\tau + \chi_2 \ln(e - \pi_2\tau),$$

S_1 and S_2 are extended by $-\infty$ for non-positive arguments of the logarithms.

Mixture pressure law

We introduce

$$\chi = \chi_1\varphi + (1-\varphi)\chi_2, \quad \zeta = \frac{\chi_1\varphi}{\chi_1\varphi + (1-\varphi)\chi_2}, \quad \gamma = \zeta\gamma_1 + (1-\zeta)\gamma_2,$$

$$\delta = -\gamma_2\pi_2, \quad r = (\delta + (\gamma-1)\rho e)^2 - 4\delta(\gamma_1 - 1)\zeta\rho e,$$

$$\alpha = \frac{\delta + (\gamma-1)\rho e - \sqrt{r}}{2\delta}.$$

Then

$$p = (\gamma-1)\rho e - \gamma(1-\alpha)\pi_2.$$

Pure phases

Pure gas $\varphi = 1$ then everything is OK

$$p = (\gamma_1 - 1)\rho e.$$

But when $\varphi = 0$ then the liquid pressure can degenerate

$$p = \max((\gamma_2 - 1)\rho e - \gamma_2 \pi_2, 0).$$

Pressureless gas for

$$\rho e \leq \frac{\gamma_2 \pi_2}{(\gamma_2 - 1)}.$$

We recover the stability of the Godunov scheme, but spurious oscillations are still here. We need another strategy.

Lagrange and remap schemes

Each time step of a Lagrange plus remap scheme is made of two stages. In the first stage, we approximate the solution with a Lagrange scheme

$$h_i^{n+1,-} W_i^{n+1,-} - h_i^n W_i^n + \tau_n \left(F_{i+1/2}^n - F_{i-1/2}^n \right) = 0.$$

The Lagrange flux is defined from a Riemann solver $R(W_L, W_R, x/t)$ and

$$\begin{aligned} F_{i+1/2}^n &= F(W_{i+1/2}^n) - u_{i+1/2}^n W_{i+1/2}^n, \\ W_{i+1/2}^n &= R(W_i^n, W_{i+1}^n, u_{i+1/2}^n), \end{aligned}$$

where the cell boundary $x_{i+1/2}^n$ moves at the (yet unknown) velocity $u_{i+1/2}^n$

$$x_{i+1/2}^{n+1,-} = x_{i+1/2}^n + \tau_n u_{i+1/2}^n.$$

$$h_i^{n+1,-} = x_{i+1/2}^{n+1,-} - x_{i-1/2}^{n+1,-} = h_i^n + \tau_n (u_{i+1/2}^n - u_{i-1/2}^n).$$

Conservative remap

The classical remap step consists in returning to the Euler grid with conservative averaging. We obtain

$$W_i^{n+1} = W_i^{n+1,-} - \frac{\tau_n}{h_i}\Big(\max(u_{i-1/2}^n, 0)(W_i^{n+1,-} - W_{i-1}^{n+1,-}) + \min(u_{i+1/2}^n, 0)(W_{i+1}^{n+1,-} - W_i^{n+1,-})\Big).$$

And we go back to the initial Euler grid

$$h_i^{n+1} = h_i^n, \quad x_i^{n+1} = x_i^n.$$

The numerical results are similar to those obtained with the conservative scheme.

Non-conservative remap

Instead of averaging $\rho\varphi$, the last component of w, we average φ [BHR04], which leads to

$$\varphi_i^{n+1} = \varphi_i^{n+1,-} - \frac{\tau_n}{h_i}\Big(\max(u_{i-1/2}^n, 0)(\varphi_i^{n+1,-} - \varphi_{i-1}^{n+1,-}) + \min(u_{i+1/2}^n, 0)(\varphi_{i+1}^{n+1,-} - \varphi_i^{n+1,-})\Big).$$

The resulting scheme is non-conservative. It preserves constant (u, p) states. The results are very similar to those obtained with the Abgrall-Saurel approach [AS99]. In the sequel, this scheme is called the BHRJ scheme.

Glimm remap (I)

We construct a sequence of random or pseudo-random numbers $\omega_n \in [0,1[$. According to this number we take [CC10]

$$W_i^{n+1} = W_{i-1}^{n+1,-} \text{ if } \omega_n < \frac{\tau_n}{h_i} \max(u_{i-1/2}^n, 0),$$

$$W_i^{n+1} = W_{i+1}^{n+1,-} \text{ if } \omega_n > 1 + \frac{\tau_n}{h_i} \min(u_{i+1/2}^n, 0),$$

$$W_i^n = W_i^{n+1,-} \text{ if } \frac{\tau_n}{h_i} \max(u_{i-1/2}^n, 0) \leq \omega_n \leq 1 + \frac{\tau_n}{h_i} \min(u_{i+1/2}^n, 0).$$

Glimm remap (II)

A good choice for the pseudo-random sequence ω_n is the (k_1, k_2) van der Corput sequence, computed by the following C algorithm

```
float corput(int n,int k1,int k2){
  float corput=0;
  float s=1;
  while(n>0){
    s/=k1;
    corput+=(k2*n%k1)%k1*s;
    n/=k1;}
  return corput;
}
```

In this algorithm, k_1 and k_2 are two relatively prime numbers and $k_1 > k_2 > 0$. In practice, we consider the $(5, 3)$ van der Corput sequence.

Glimm remap (III)

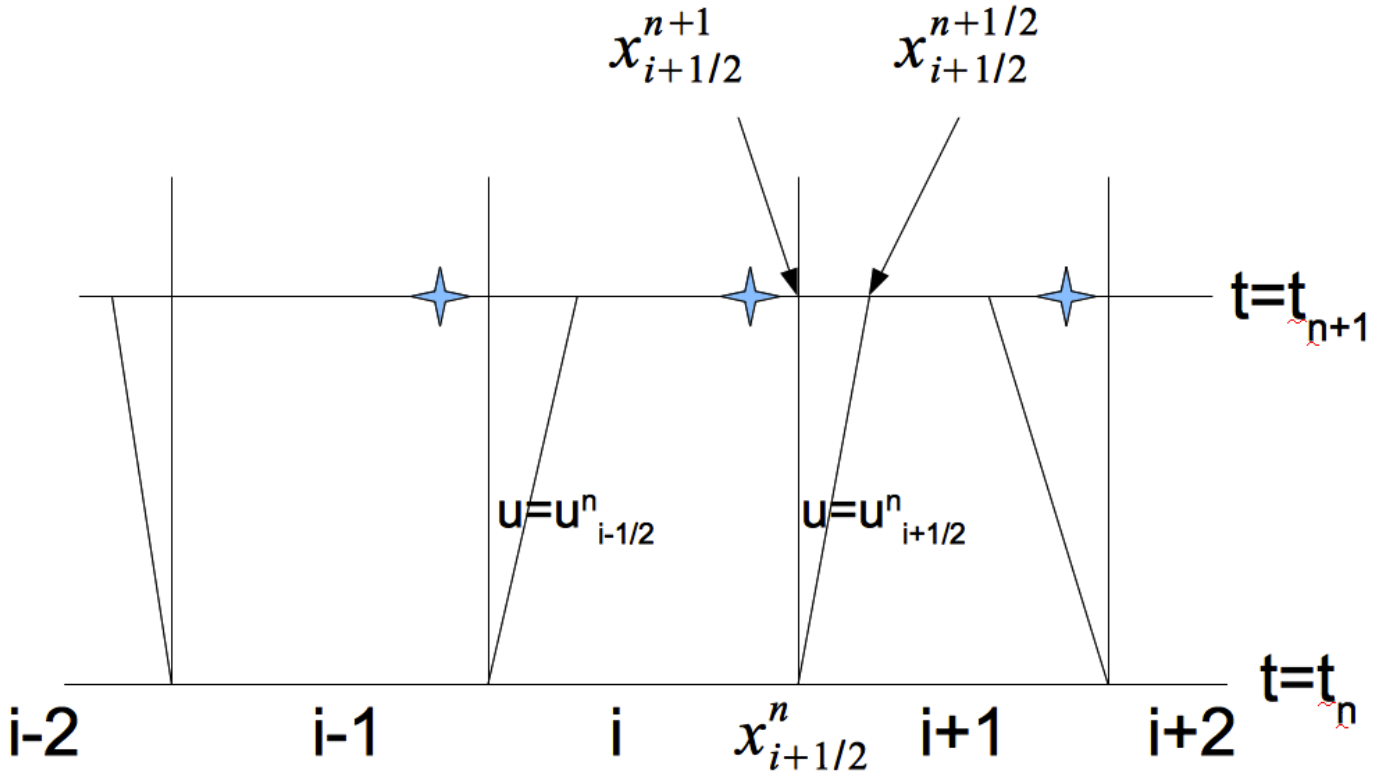


Figure : Example of Glimm remap. The stars correspond to the sampling points. In cells $i-1$ and i, we keep the values of the Lagrange cells. In cell $i+1$, we take the right values.

Lagrange interface velocity

We have to provide the interface velocities $u^n_{i+1/2}$.
In the resolution of the Riemann problem $R(W^n_i, W^n_{i+1}, x/t)$ we find four waves. The characteristic fields 2 and 3 are linearly degenerated and $\lambda_2(w) = \lambda_3(w) = u$, thus the velocity is constant across these waves. It corresponds to the interface velocity, which we denote by $u^*(W_i, W_{i+1})$. It is then natural to take

$$u^n_{i+1/2} = u^*(W^n_i, W^n_{i+1}).$$

Relaxation Riemann solver

We use an approximate Riemann solver based on relaxation techniques.

- it is positive and handles vacuum.
- entropy dissipative.

Properties

- The constant (u, p) states are exactly preserved.
- The gas fraction is not smeared at all.
- It is possible to use any approximate Riemann solver in the Lagrange step.
- Statistically conservative.
- Convergence ?

Weak shock

The first test consists in a two-fluid shock tube. The stiffened gas parameter are

$$\gamma_W = 2, \quad \pi_W = 1,$$
$$\gamma_A = 1.4, \quad \pi_A = 0.$$

We take for the left and right initial data

$$(\rho_L, u_L, p_L, \varphi_L) = (2, 1/2, 2, 1),$$
$$(\rho_R, u_R, p_R, \varphi_R) = (1, 1/2, 1, 0).$$

We compare the non-conservative remap and the Glimm remap. The Riemann solver is the approximate VFRoe solver in the (ρ, u, p, φ) variables.

Convergence study

The convergence rate is approximately 0.6.

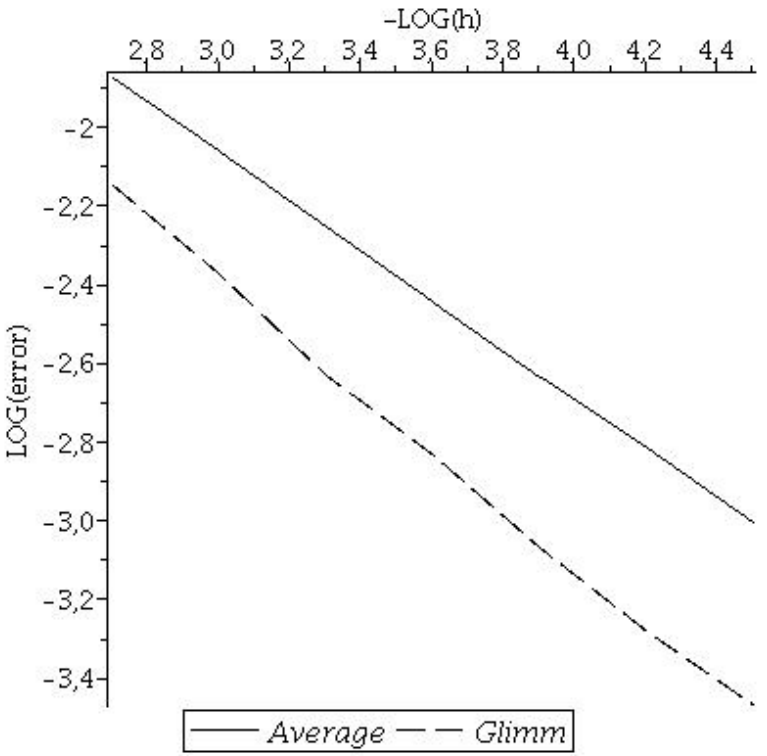


Figure : Convergence study: Glimm remap versus non-conservative averaging remap, weak shock.

Strong shock

Interaction between a shock of velocity $\sigma = 4$ and a contact of velocity $v = -1$.
The initial positions of the contact and the shock are chosen in such way that they meet together at the abscissa $x = 0$ at time $t = 1$. The EOS parameters are $\gamma_1 = 1.4$, $\pi_1 = 0, \gamma_2 = 2$, $\pi_2 = 7$.
The initial data are

$$\begin{array}{rcll} (\rho_L, u_L, p_L, \varphi_L) & = & (3.4884, 1.1333, 23.333, 1), & x < -4, \\ (\rho_M, u_M, p_M, \varphi_M) & = & (2, -1, 2, 1), & -4 \leq x \leq 1, \\ (\rho_R, u_R, p_R, \varphi_R) & = & (1, -1, 2, 0), & x > 1. \end{array}$$

After the interaction at time $t = 1$, the solution is simply given by the resolution of a two-fluid Riemann problem between states (L) and (R).

Strong shock

Similar phenomena in [Col82]

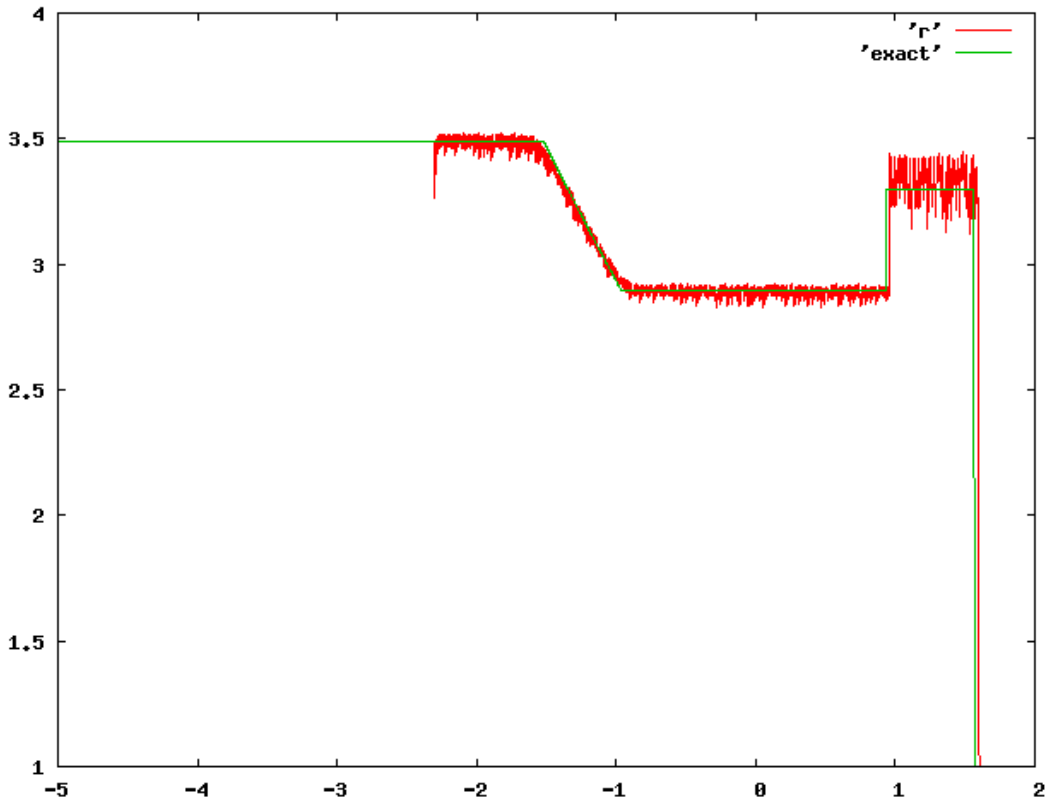


Figure : Glimm approach, density plot. BV explosion due to wall-heating effect propagation, strong shock.

Modified interface velocity

Simple remark: if one takes $u^n_{i+1/2} = 0$, we fall back on the classical Godunov scheme, which solves correctly the shock waves.
The idea is to use the Glimm approach only at the interface, thus we propose the choice

$$u^n_{i+1/2} = \begin{cases} u^*(W^n_i, W^n_{i+1}) \text{ if } \varphi^n_i \neq \varphi^n_{i+1}, \\ 0 \text{ if } \varphi^n_i = \varphi^n_{i+1}. \end{cases}$$

The scheme has the same properties as before but the "BV explosion" is removed in strong shocks.

Numerical results

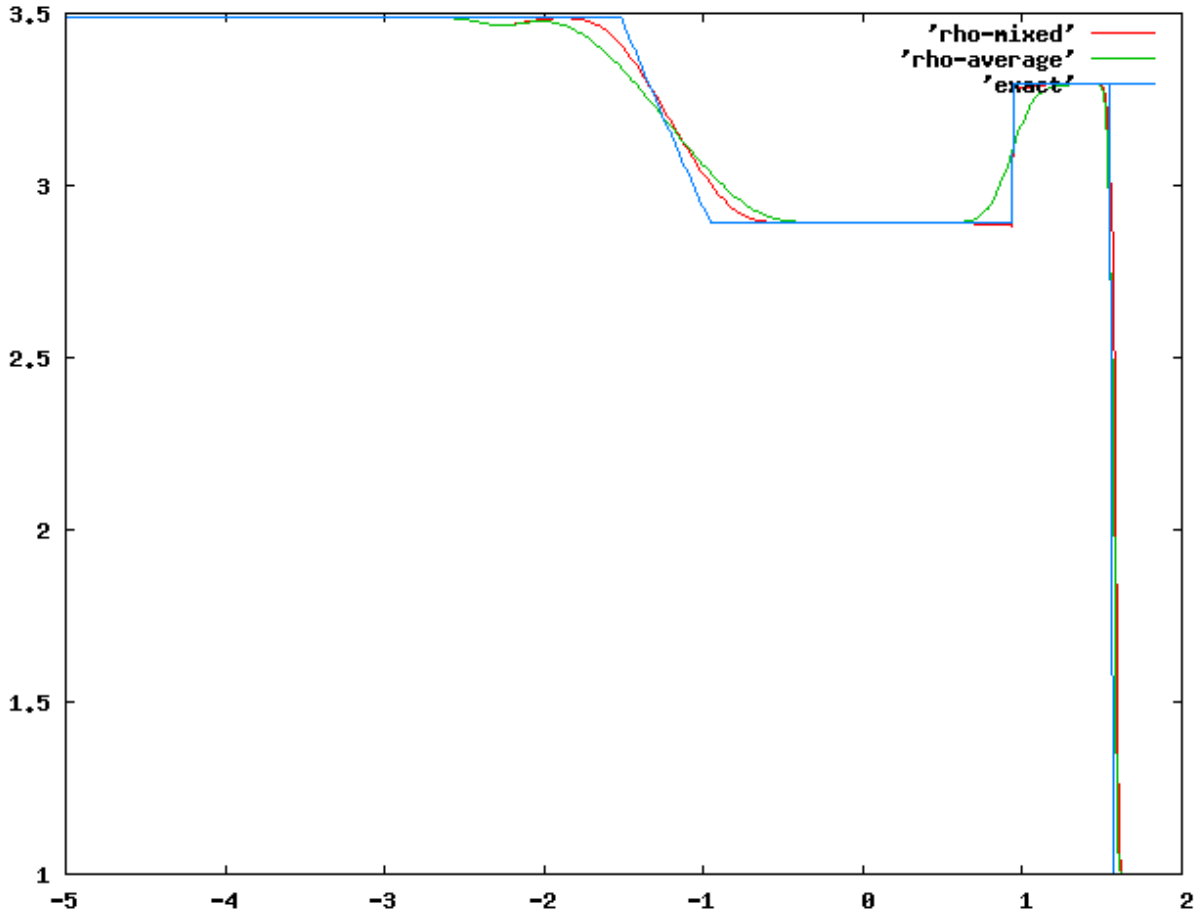


Figure : Density. Comparison of the modified Glimm and averaging remap schemes

Convergence study

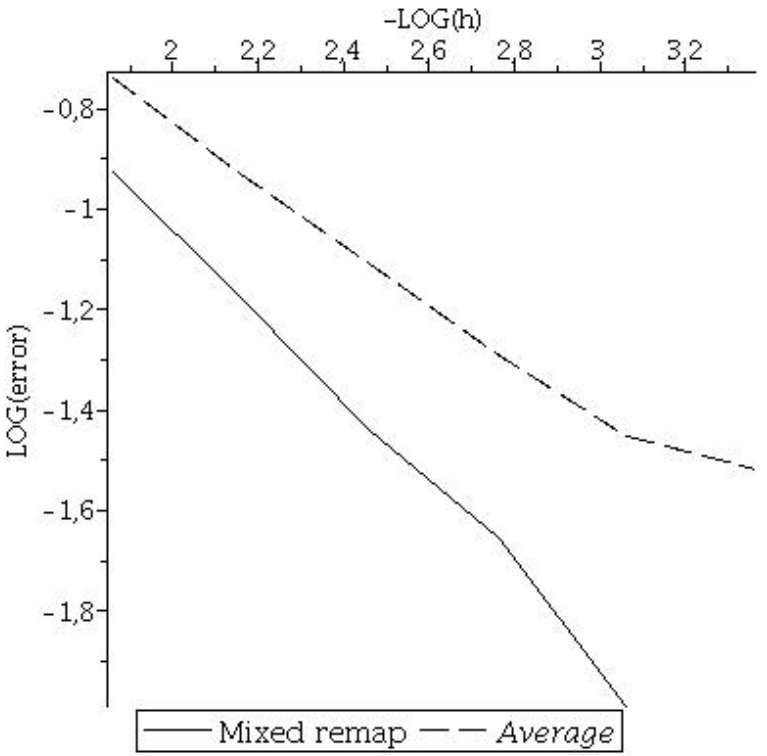


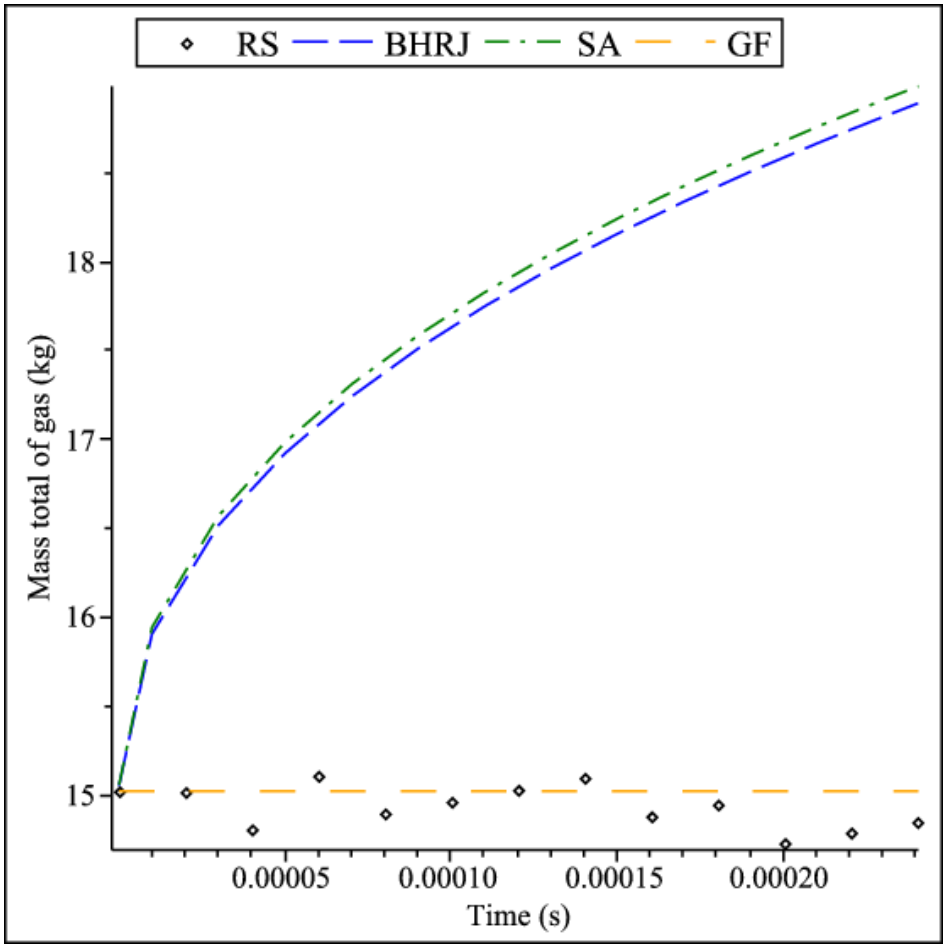
Figure : strong shock-interface interaction. Convergence study. modified Glimm remap and averaging remap.

Conservation (I)

The numerical mass transfer between the two fluids should be zero. We compare it for different schemes:

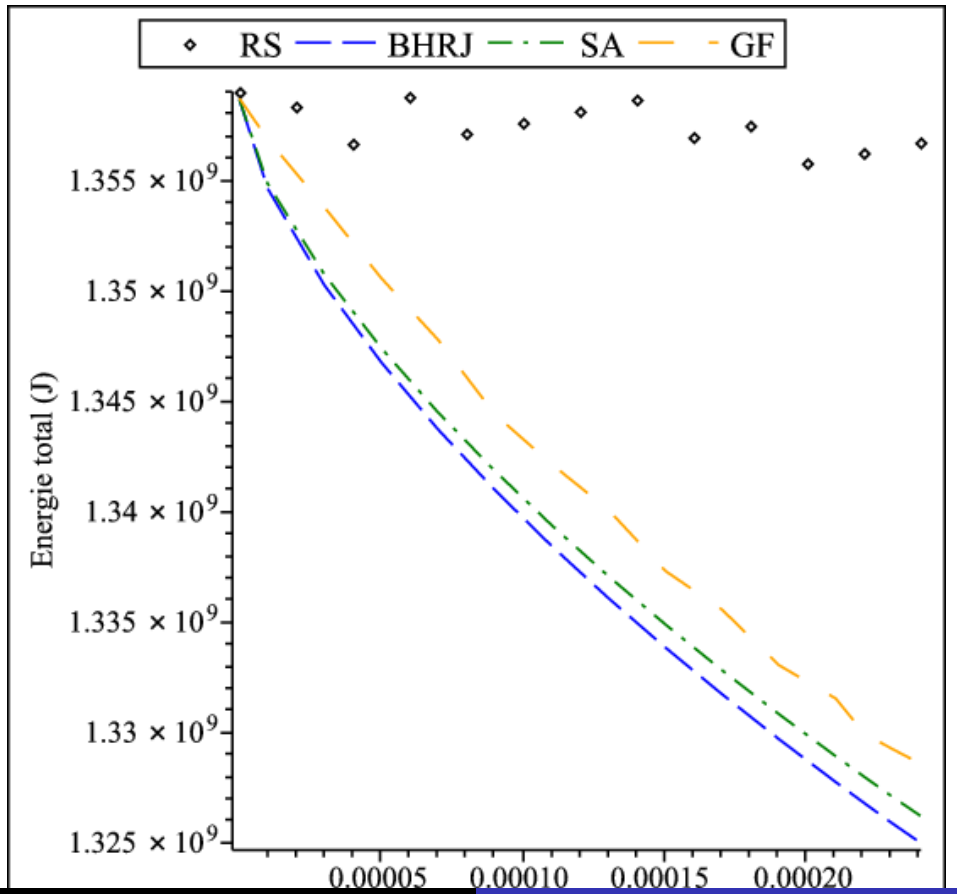
- Saurel-Abgrall scheme (SA) [AS99];
- A Lagrange and remap version of Saurel-Abgrall (BHRJ) [BHR04];
- The "Ghost Fluid for the poor" (GF) scheme [AK11];
- The random scheme (RS).

Conservation (II)



Conservation (III)

The total energy should be exactly conserved. We compare the energy conservation property of the four same schemes



Dimensional splitting

In order to perform 2D computations, we use dimensional splitting.
For advancing a time step τ, we first numerically solve

$$\frac{W^* - W^n}{\tau} + \partial_x F^1(W^n) = 0,$$

and then

$$\frac{W^{n+1} - W^*}{\tau} + \partial_y F^2(W) = 0,$$

with the Lagrange and remap scheme.
For performance reasons, we implement the algorithm on a GPU.

GPU (I)

A modern Graphics Processing Unit (GPU) is made of:

- Global memory (typically 1 Gb)
- Compute units (typically 27).

Each compute unit is made of:

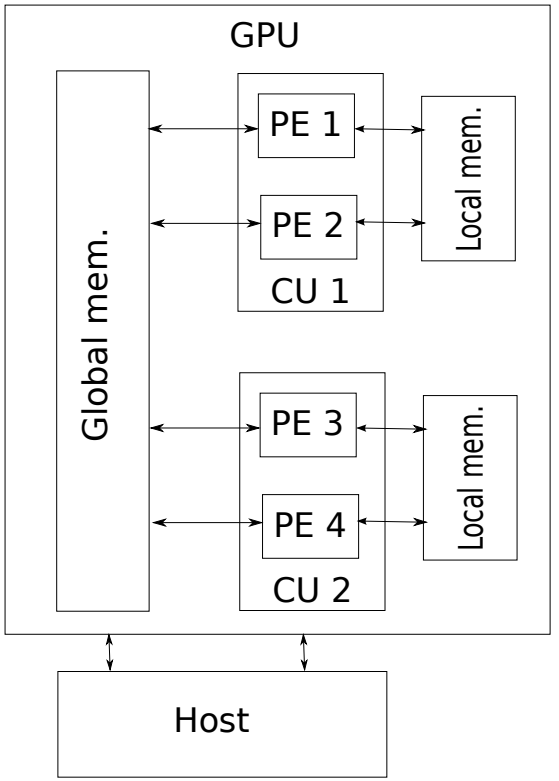
- Processing elements (typically 8).
- Local memory (typically 16 kb)

The same program can be executed on all the processing elements at the same time.

- All the processing elements have access to the global memory.
- The processing elements have only access to the local memory of their compute unit.
- If two processing elements write at the same location at the same time, only one wins...
- The access to the global memory is slow while the access to the local memory is fast.

GPU (II)

A (virtual) GPU with 2 Compute Units and 4 Processing Elements



OpenCL

- OpenCL means "Open Computing Language". It includes:
  - A library of C functions, called from the host, in order to drive the GPU.
  - A C-like language for writing the kernels that will be executed on the processing elements.
- Practically available since september 2009. The specification is managed by the Khronos Group (OpenGL).
- Virtually, it allows to have as many compute units (work-groups) and processing elements (work-items) as needed.
- The threads are sent to the GPU thanks to a mechanism of command queues on the real compute units and processing elements.
- Portable: the same program can run on a multicore CPU or a GPU.

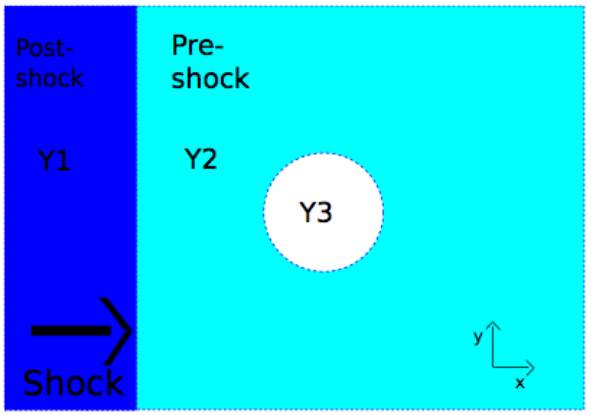
Implementation of the splitting scheme

We organize the data in a (x, y) grid and for each time step:

- we associate a processor to each cell of the grid.
- we compute the fluxes balance in the x-direction for each cell of each row of the grid. A row (or a part of the row) is associated to one compute unit and one cell to one processor.
- subdomain strategy in order to retain data into the local cache memory. Covering of two cells between the subdomain (for the correctness of the boundary values).
- we transpose the grid (exchange x and y) with an optimized memory transfer algorithm.
- we compute the fluxes balance in the y-direction for each row of the transposed grid. Memory access are optimal.
- we transpose again the grid.

Shock-bubble interaction

We consider a shock that comes to a bubble at velocity $\sigma = 415 m.s^{-1}$ (see [KL10]).



The initial datas are:

Quantities	Y1	Y2	Y3
$\rho(kg.m^{-3})$	1.69	1.22	3.86
$u(m.s^{-1})$	113.5	0	0
$v(m.s^{-1})$	0	0	0
$p(Pa)$	1.6e5	1.0e5	1.0e5
φ	0	0	1
γ	1.4	1.4	1.249
π	0	0	0

Speedup

	time (s)
AMD Phenom II x4 945 (1 core)	192
AMD Phenom II x4 945 (4 cores)	59
AMD Radeon HD5850	1.43
NVIDIA GTX 460	2.48
NVIDIA Geforce GTX470	0.93

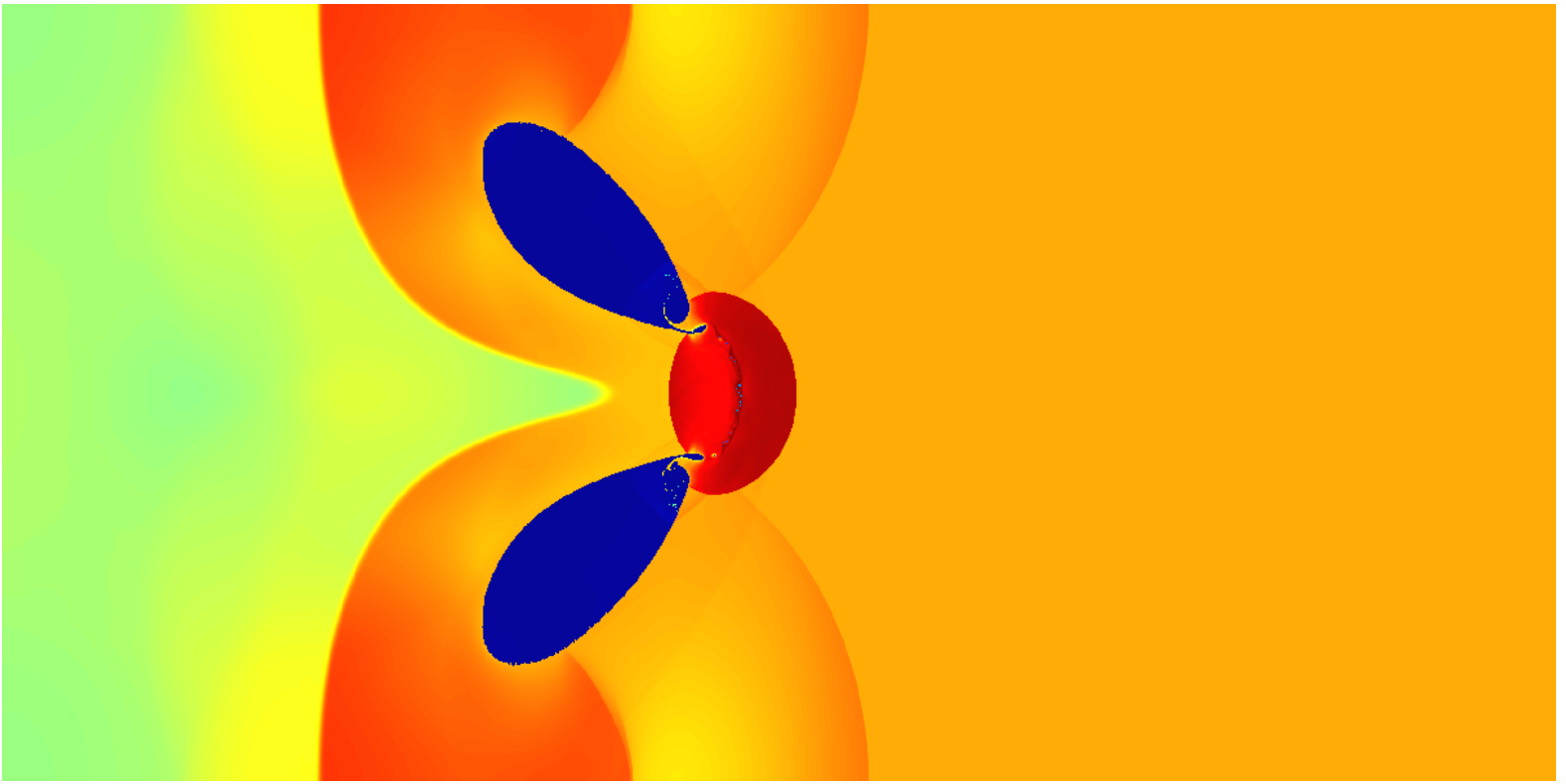
Animation

http://www.youtube.com/watch?v=c8hcqihJzbw

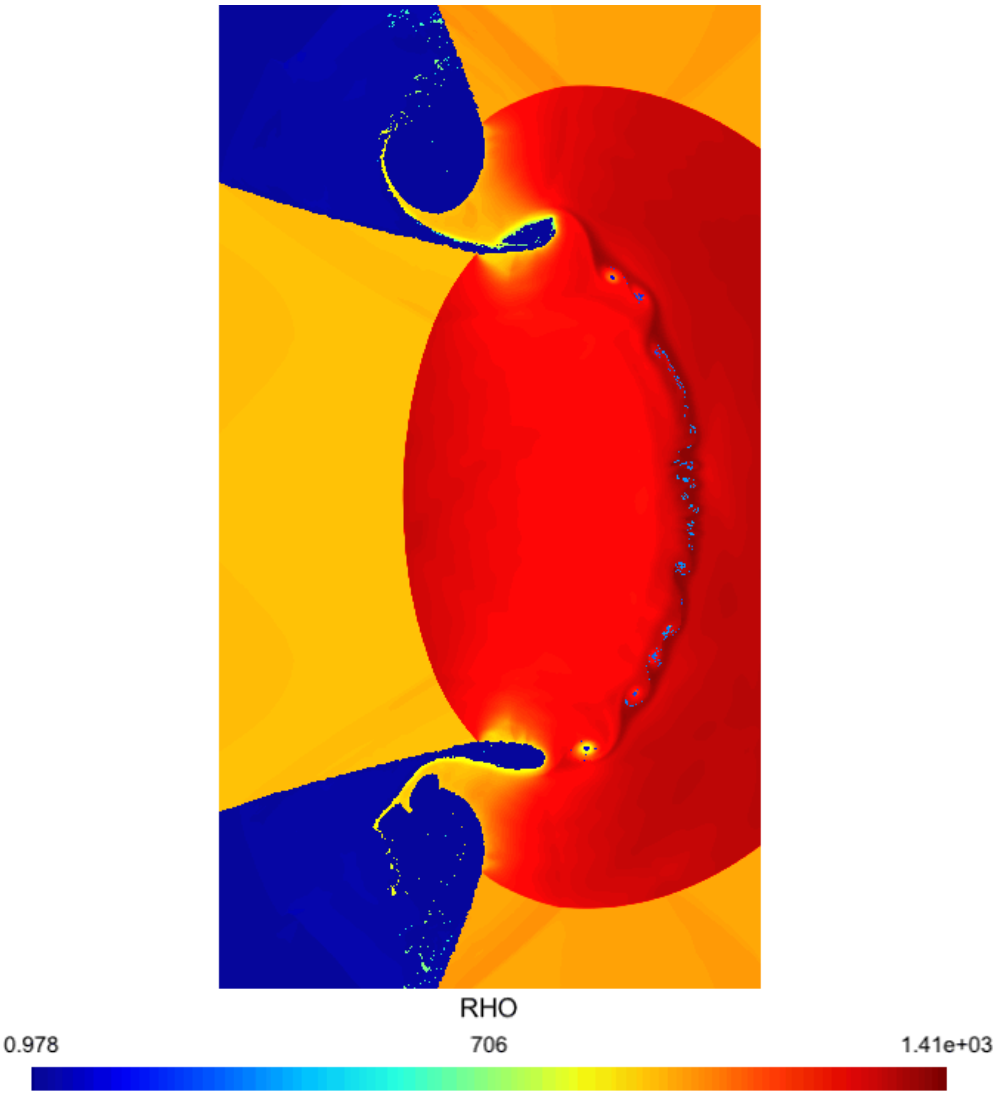
Numerical results

$t_{\max} = 0.45$ ms
Grid: 40,000×20,000 (4 billions unknowns for each time step)
GPU time: 30 h (10×NVIDIA K20)

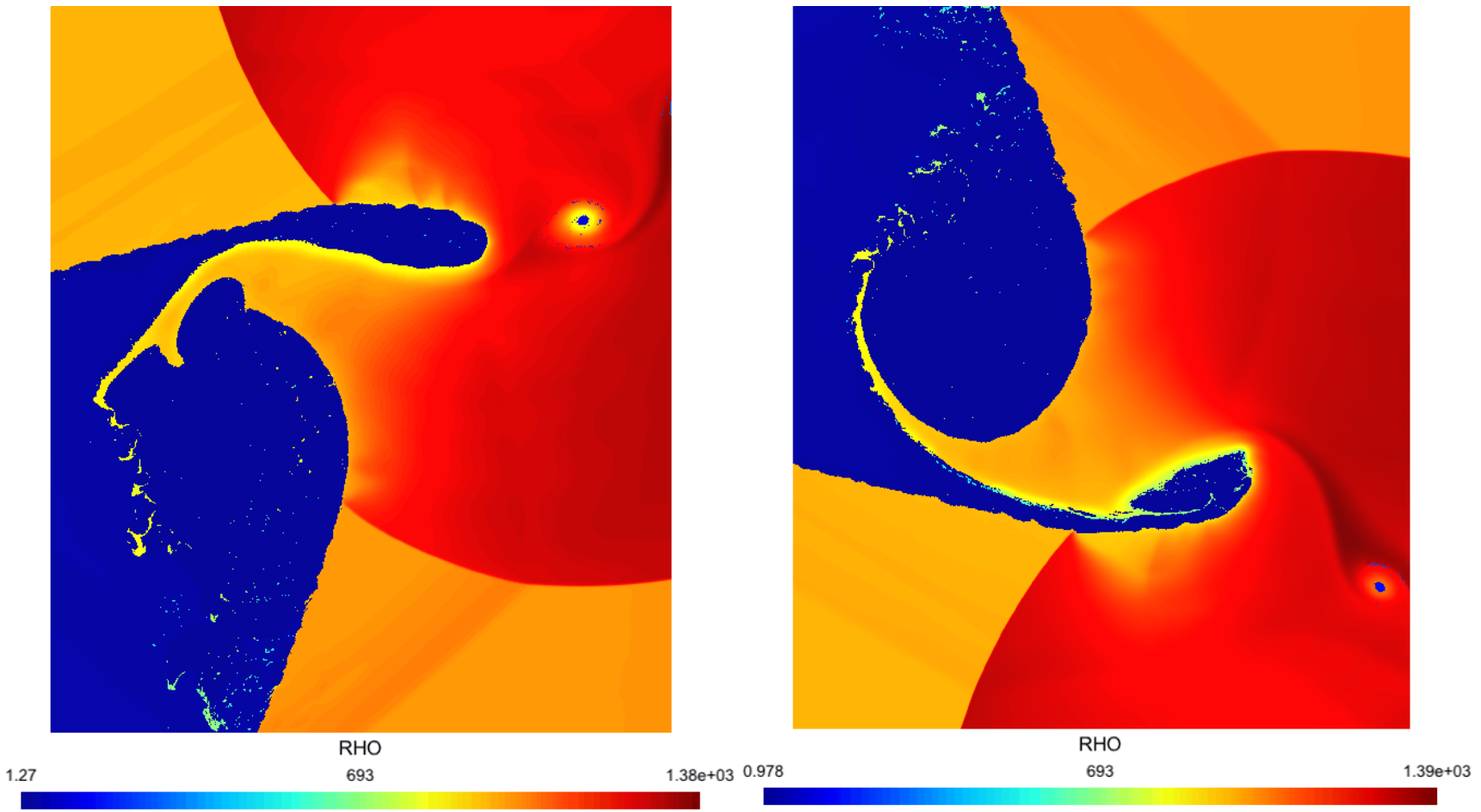
Density



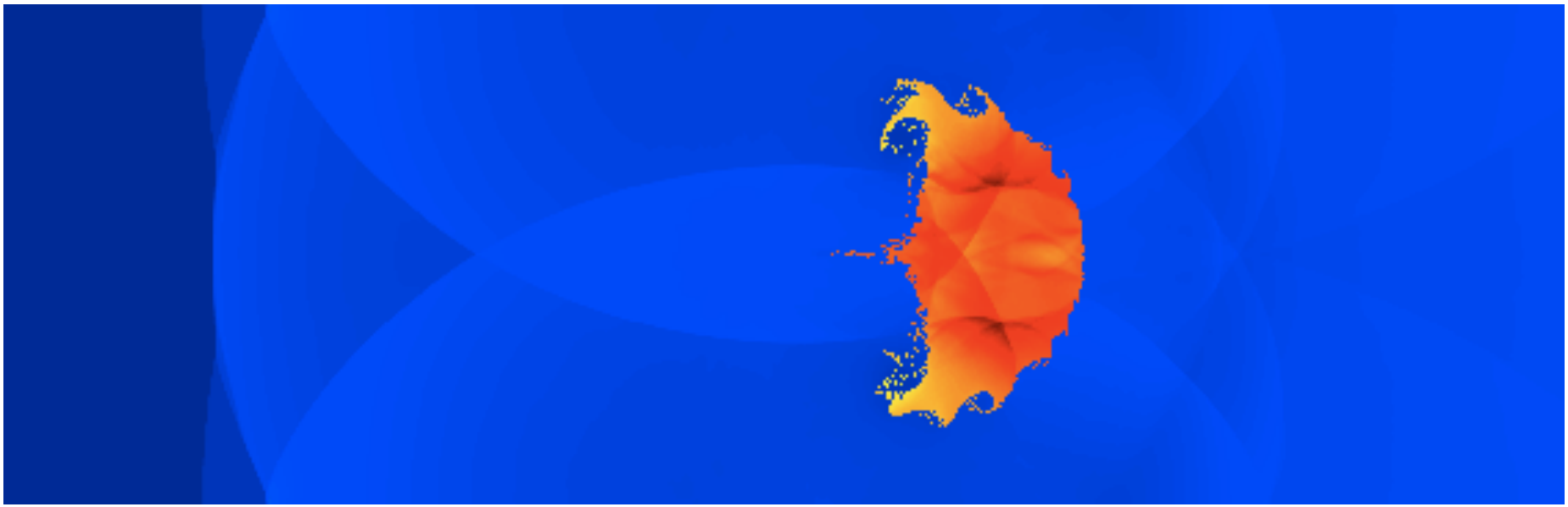
Zoom 1



Zoom 2



Shock-droplet interaction (III)



Conclusion

- convex hyperbolic domain of the two-fluid model with physical pressure law. Still numerical inaccuracy in the mixture region.
- Random scheme for solving two-fluid compressible flows with non-convex hyperbolicity domain
- the scheme enjoys interesting stability and conservation properties.
- It is well adapted to multicore computations.

Abgrall, Rémi; Karni, Smadar Ghost-fluids for the poor: a single fluid algorithm for multifluids. 1–10, Internat. Ser. Numer. Math., 140, 141, Birkhäuser, Basel, 2001.

Bachmann M., Helluy P., Mathis H., Müller S., Jung J. Random sampling remap for compressible two-phase flows [hal-00546919 - version 1] (15/12/2010)

Barberon, Thomas; Helluy, Philippe; Rouy, Sandra Practical computation of axisymmetrical multifluid flows. Int. J. Finite Vol. 1 (2004), no. 1, 34 pp.

C. Chalons, F. Coquel. Computing material fronts with Lagrange-Projection approach. HYP2010 Proc. http://hal.archives-ouvertes.fr/hal-00548938/fr/.

Colella, Phillip Glimm's method for gas dynamics. SIAM J. Sci. Statist. Comput. 3 (1982), no. 1, 76–110.


J.-P. Croisille. Contribution à l'étude théorique et à l'approximation par éléments finis du système hyperbolique de la dynamique des gaz multidimensionnelle et multiespèces. PhD thesis, Université Paris VI, 1990.


B. Després, F. Lagoutière. Numerical resolution of a two-component compressible fluid model with interfaces. Prog. Comp. Fluid Dyn. 7 (6) (2007) 295-310.


A. Harten, P. D. Lax, C. D. Levermore, W. J. Morokoff. Convex entropies and hyperbolicity for general Euler equations. SIAM J. Numer. Anal. 35 (1998), no. 6, 2117–2127


P. Helluy, H. Mathis. Pressure laws and fast Legendre transform. Math. Models Methods Appl. Sci. 21 (2011), no. 4, 745–775.

 J. Jung. Schémas numériques adaptés aux accélérateurs multicœurs pour les écoulements bifluides. PhD thesis. Université de Strasbourg. 2013.

 S. Kock, F. Lagoutière. An anti-diffuse numerical scheme for the simulation of interfaces between compressible fluids by means of a five-equation model. Journal of Comp. Phy. 229 (2010) 2773-2809.

 Müller, Siegfried; Helluy, Philippe; Ballmann, Josef Numerical simulation of a single bubble by compressible two-phase fluids. Internat. J. Numer. Methods Fluids 62 (2010), no. 6, 591–631.

 Saurel, R.; Abgrall, R. A simple method for compressible multifluid flows. SIAM J. Sci. Comput. 21 (1999), no. 3, 1115–1145