A multi-scale fast semi-Lagrangian method for rarefied gas dynamics

V. Rispoli[1],
G. Dimarco[2], R. Loubère[1],

[1]Institut de Mathématique de Toulouse (IMT), France

[2]Dipartimento di matematica ed informatica, Ferrara, Italy.

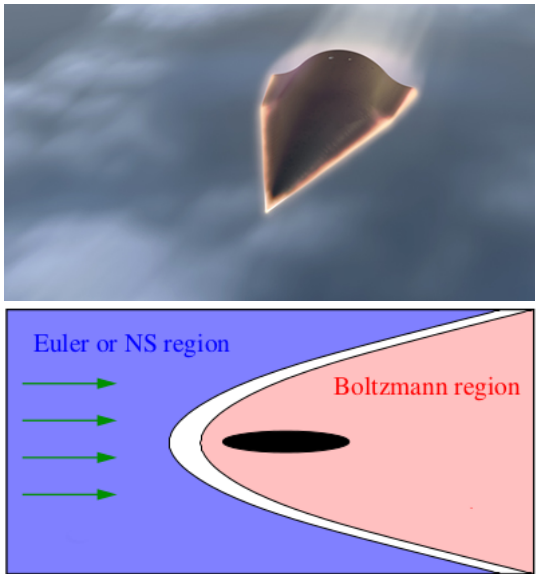
www.math.univ-toulouse.fr/∼vrispoli/
vrispoli@math.univ-toulouse.fr

SHARK-FV 2014 Conference - Ofir, Portugal
30.04.2014

Motivations

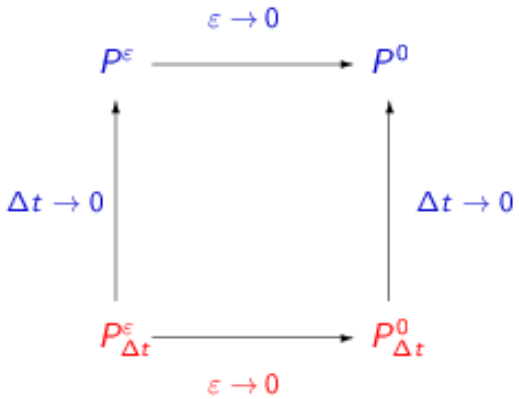
Non equilibrium and multi-scale

- Many applications involve non equilibrium gas flows (hypersonic objects, plasmas)
- Breakdowns of fluid models (Euler or NS) $\Rightarrow$ connection between equilibrium and non equilibrium regions
- Combine macroscopic/fluid numerical schemes with microscopic/kinetic ones



Possible solutions

- These problems involve mutli-scale solutions in time and/or space
- Construct numerical methods which address the multi-scale nature of solutions (AP).
- Exploit physical properties of the system via Domain Decomposition techniques



Kinetic - Fluid models

Boltzmann-BGK description of rarefied gaz dynamics

$$\partial_t f + \boldsymbol{V} \cdot \nabla_{\boldsymbol{X}} f = \frac{1}{\tau}(M_f - f), \qquad \boldsymbol{X} \in \Omega \subset \mathbb{R}^3, \boldsymbol{V} \in \mathbb{R}^3$$

$f = f(\boldsymbol{X}, \boldsymbol{V}, t)$ density of particles, $\tau > 0$ is the relaxation time. BGK-type collisions

$$M_f = M_f[\rho, \boldsymbol{U}, T](\boldsymbol{V}) = \frac{\rho}{(2\pi\theta)^{3/2}} \exp\left(\frac{-\|\boldsymbol{U} - \boldsymbol{V}\|^2}{2\theta}\right)$$

where $\rho \in \mathbb{R}$, $\rho > 0$ and $\boldsymbol{U} = (u, v, w)^t \in \mathbb{R}^3$ are the density and mean velocity, θ defined as $\theta = RT$ with T the temperature, R gas constant.

Macroscopic moments

Moments ρ, $\boldsymbol{U}$ and T are related to f in 3D by

$$\rho = \int_{\mathbb{R}^3} f \, d\boldsymbol{V}, \qquad \boldsymbol{U} = \frac{1}{\rho} \int_{\mathbb{R}^3} \boldsymbol{V} f \, d\boldsymbol{V}, \qquad \theta = \frac{1}{3\rho} \int_{\mathbb{R}^3} \|\boldsymbol{V} - \boldsymbol{U}\|^2 \, f \, d\boldsymbol{V}$$

with total energy $E = \frac{1}{2} \int_{\mathbb{R}^3} \|\boldsymbol{V}\|^2 f \, d\boldsymbol{V} = \frac{1}{2}\rho\|\boldsymbol{U}\|^2 + \frac{3}{2}\rho\theta$

Coupling of the models

Boltzmann-BGK description when $\tau \to 0$

If number of collisions tends to ∞ then $\tau \to 0$ therefore $f \to M_f$ and from Boltzmann-BGK one retrieves Euler compressible gas dynamics

$$\begin{aligned} \frac{\partial \rho}{\partial t} + \nabla_{\boldsymbol{x}} \cdot (\rho \boldsymbol{U}) &= 0 \\ \frac{\partial \rho \boldsymbol{U}}{\partial t} + \nabla_{\boldsymbol{x}} \cdot (\rho \boldsymbol{U} \otimes \boldsymbol{U} + p I) &= 0 \\ \frac{\partial E}{\partial t} + \nabla_{\boldsymbol{x}} \cdot ((E + p) \boldsymbol{U}) &= 0 \end{aligned}$$

Pressure $p = \rho\theta$ is given by a perfect gas equation of state with gas constant $\gamma = 2/3 + 1 = 5/3$. Set $F = (\rho, \rho \boldsymbol{U}, E)^t$.

Coupling strategy

Kinetic/microscopic model is Boltzmann-BGK – Fluid/macroscopic model is Euler system.

Fast Kinetic Scheme (FKS)

General first order scheme: Equations

Prelims

Semi-Lagrangian scheme for Discrete Velocity Model (DVM) approximation of the kinetic equation. Kinetic equation + velocity grid $\Longrightarrow$ linear hyperbolic system with source terms. However particle or lattice Boltzmann interpretations are also possible.

DVM

Let $\mathcal{K}$ be a set of N multi-indices of $\mathbb{N}^3$ with bounds. Then the Cartesian grid $\mathcal{V}$ of $\mathbb{R}^3$

$$\mathcal{V} = \{\boldsymbol{V}_k = k\Delta v + \boldsymbol{W},\ k \in \mathcal{K}\}$$

Δv the grid step in velocity space. Discrete collision invariants: $m_k = \left(1, \boldsymbol{V}_k, \frac{1}{2}\|\boldsymbol{V}_k\|^2\right)^t$.
Continuous distribution f is replaced by

$$f_{\mathcal{K}}(\boldsymbol{X}, t) = (f_k(\boldsymbol{X}, t))_k\,, \qquad f_k(\boldsymbol{X}, t) \approx f(\boldsymbol{X}, \boldsymbol{V}_k, t)$$

Fluid quantities are retrieved back from f_k using

$$F(\boldsymbol{X}, t) = \sum_{k\in\mathcal{K}} m_k\, f_k(\boldsymbol{X}, t)\, \Delta v$$

Fast Kinetic Scheme (FKS)

General first order scheme: DVM

Discrete velocity BGK model

Set of N evolution equations in $\mathcal{V}$ where $\mathcal{E}_k[F]$ is a suitable approximation of M_f.

$$\partial_t f_k + \boldsymbol{V}_k \cdot \nabla_{\boldsymbol{X}} f_k = \frac{1}{\tau}(\mathcal{E}_k[F] - f_k),\ k = 1, .., N$$

Space/Time discretization

Cartesian uniform grid $\mathcal{X}$ of $\mathbb{R}^{d_x}$: $\mathcal{X} = \{\boldsymbol{X}_j = j\Delta x + \boldsymbol{Y},\ j \in \mathcal{J}\}$, $\boldsymbol{Y}$ is a vector of $\mathbb{R}^3$ and Δx is the grid step in the physical space.
Time discretization: $t^{n+1} = t^n + \Delta t$ with Δt the time step that is defined by a CFL condition.

Time splitting procedure

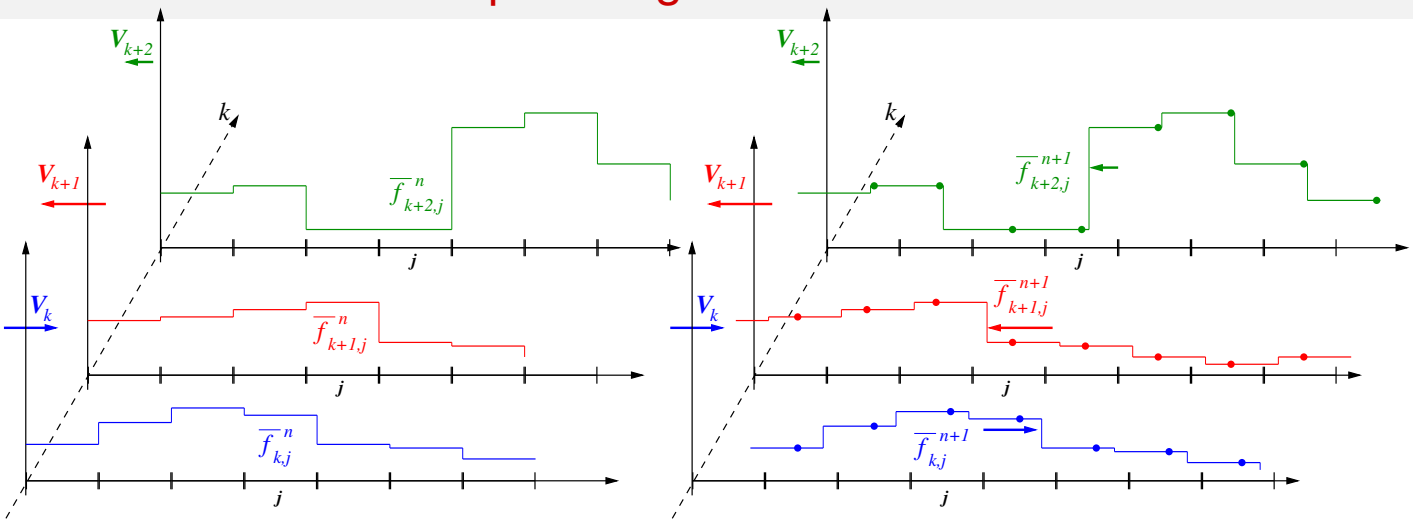
The fully discretized system is solved by a time splitting. Transport stage solves the LHS, Relaxation stage solves the RHS (using solution from transport stage)

$$\textit{Transport stage} \longrightarrow \partial_t f_k + \boldsymbol{V}_k \cdot \nabla_{\boldsymbol{X}} f_k = 0$$

$$\textit{Relaxation stage} \longrightarrow \partial_t f_k = \frac{1}{\tau}(\mathcal{E}_k[F] - f_k)$$

Fast Kinetic Scheme (FKS)

General first order scheme: Transport stage



Let $f^0_{j,k}$ be the pointwise data at t^0 at any point $\boldsymbol{X}_j$: $f^0_{j,k} = f(\boldsymbol{X}_j, \boldsymbol{V}_k, t^0)$ and $\mathcal{E}^0_{j,k}[F]$ the equilibrium distribution approximation of $M^0_{j,k} = M_f(\boldsymbol{X}_j, \boldsymbol{V}_k, t^0)$

We denote $\overline{f}^0_k(\boldsymbol{X})$ a piecewise continuous function for all $\boldsymbol{X} \in \Omega$ associated with mesh $\mathcal{X}$ at the time t^0 and for velocity $\boldsymbol{V}_k$ in finite volume sense

$$\overline{f}^0_{k,j} = \frac{1}{|\Omega_j|} \int_{\Omega_j} f(\boldsymbol{X}, \boldsymbol{V}_k, t^0)\, d\boldsymbol{X}, \qquad \text{on} \quad \Omega_j = [\boldsymbol{X}_{j-1/2}; \boldsymbol{X}_{j+1/2}]$$

Exact transport during Δt

$$\overline{f}^*_k = \overline{f}^n_k(\boldsymbol{X} - \boldsymbol{V}_k \Delta t), \qquad \forall \boldsymbol{X} \in \Omega$$

Fast Kinetic Scheme (FKS)

General first order scheme: Relaxation stage

Relaxation step solution locally resolved on the grid

$$\partial_t f_{j,k} = \frac{1}{\tau}(\mathcal{E}_{j,k}[F] - f_{j,k})$$

with initial data coming from the transport step given by $f^*_{j,k} = \bar{f}^*_k(\boldsymbol{X}_j)$, for all k, j.
Macroscopic quantities needed to compute the Maxwellian

$$F^n_j = F^*_j = \sum_{k \in \mathcal{K}} m_k \, f^*_{j,k} \, \Delta v$$

Moments before (F^n_j) and after (F^*_j) are unchanged: preservation of macroscopic quantities.

Then

$$f^{n+1}_{j,k} = \exp(-\Delta t/\tau) f^*_{j,k} + (1 - \exp(-\Delta t/\tau))\mathcal{E}_k[F^*_j]$$

Fast Kinetic Scheme (FKS)

Conservation of macroscopic quantities

Constrained optimization formulation ($d_x = 3$)

- let $\hat{f} = \left(\hat{f}_1, \hat{f}_2, \ldots, \hat{f}_N\right)^t$ be the pointwise distribution vector and $f = (f_1, f_2, \ldots, f_N)^t$ be the unknown which fulfill the conservation of moments
- $C_{(d_x+2)\times N} = \left((\Delta v)^3, \boldsymbol{V}_k(\Delta v)^3, \|\boldsymbol{V}_k\|^2/2(\Delta v)^3\right)^t$ a constant in time matrix
- $F_{(d_x+2)\times 1} = (\rho, \rho\boldsymbol{U}, E)^t$ be the vector of the conserved quantities.

Conservation can be imposed solving[a]:

$$\text{Given } \hat{f} \in \mathbb{R}^N,\ C \in \mathbb{R}^{(d_x+2)\times N},\ \text{and } F \in \mathbb{R}^{(d_x+2)\times 1},$$
$$\text{find } f \in \mathbb{R}^N \text{ that minimizes } \quad \|\hat{f} - f\|_2^2 \text{ under constraints } Cf = F.$$

Using a Lagrange multiplier $\lambda \in \mathbb{R}^{d_x+2}$, the objective function to be optimized is $L(f, \lambda) = \sum_{k=1}^{N} |\hat{f}_k - f_k|^2 + \lambda^T(Cf - F)$. Exactly solved by

$$f = \hat{f} + C^T(CC^T)^{-1}(F - C\hat{f}).$$

Also done for the equilibrium distribution $\mathcal{E}[F]$ starting from $M_f[F]$.

[a] Gamba et al JCP, 228 (2009)

Fast Kinetic Scheme (FKS)

Properties

Properties

- Globally conservative, unconditionally positive (if constrained optimization is)
- When rarefied $\rightarrow$ dense regimes $\Longrightarrow$ projection over the equilibrium becomes important. Accuracy diminishes in fluid regime because the projection is first order accurate.
- Δt under CFL (but stability $\forall \Delta t$). However splitting error is of the order of Δt.

Towards an ultra efficient kinetic scheme. Part I: basics on the BGK equation, G. Dimarco and R. Loubère, Journal of Computational Physics, vol. 255, pp. 680–698 (2013)

Extension 1: second order in time

Time step is solved using a Strang splitting strategy. Δt follows a CFL condition

$$\Delta t \max_k \left(\frac{\|\boldsymbol{V}_k\|}{\Delta x} \right) < 1$$

High Order Fast Kinetic Scheme (HOFKS)

High order in space extension

Recall

$$f_{j,k}^{n+1} = \exp(-\Delta t/\tau) f_{j,k}^{*} + (1 - \exp(-\Delta t/\tau))\mathcal{E}_k[F_j^{*}]$$

High-Order Fast Kinetic Scheme (HOFKS): second order in space

Idea: solve the equilibrium part of the distribution function with a macroscopic scheme instead of a kinetic scheme.
Moments at t^* from the transport stage are now computed by a High Order shock capturing scheme (MUSCL here).

- In the limit $\tau \to 0$ HOFKS corresponds to the HO shock capturing scheme
- Nominally second order in the fluid limit
- higher accuracy and efficiency in 3D (smaller # of cell for same accuracy)

Reference

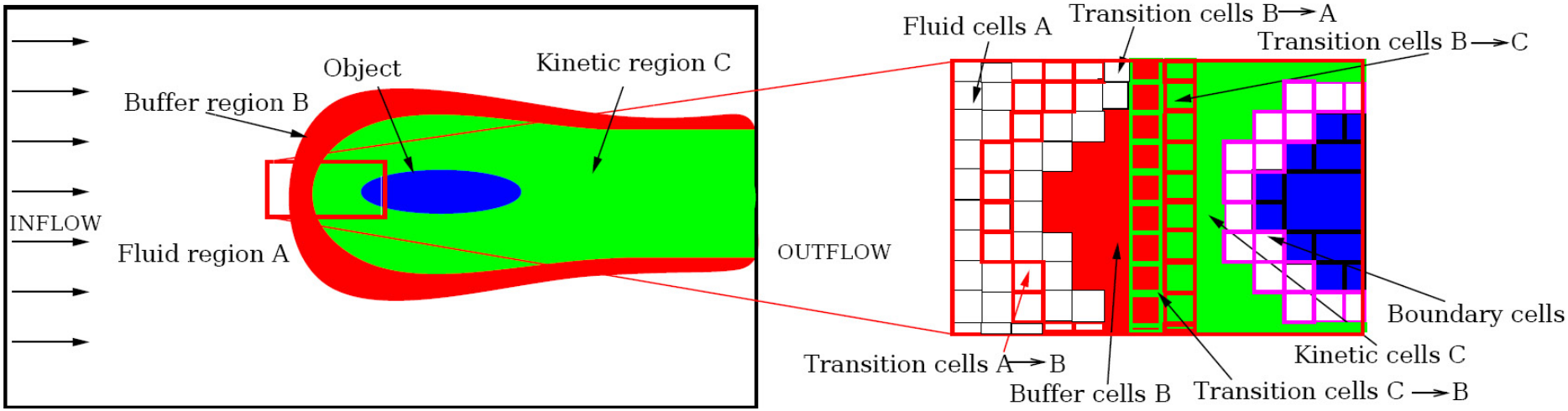
Towards an ultra efficient kinetic scheme Part II: The High-order case, G. Dimarco and R. Loubère, Journal of Computational Physics, Volume 255, pp 699-719 (2013)

HOFKS with Domain Decomposition

Automatic Domain Decomposition

Motivation

- Not the entire domain may need the expensive kinetic (microscopic) description
- Idea: reduce as much as possible the "kinetic region", improving efficiency at same accuracy



Reference

A multiscale fast semi-Lagrangian method for rarefied gas dynamics, G. Dimarco, R. Loubère and V. Rispoli, submitted to JCP (2014)

HOFKS with Domain Decomposition

Automatic Domain Decomposition

Automatic Domain Decomposition ingredients

- ***A***: fluid zone, ***B***: buffer zone, ***C***: kinetic zone
- all regions evolve in time
- carefully treat transition regions
- Inside the buffer zone ***B***, use a kinetic model with a sparser grid (fewer particles in PIC) Think of it as an intermediate region!

Transition cells

Using the HOFKS we can use

$$f_{j,k}^{n+1} = \exp(-\Delta t/\tau) f_{j,k}^{*} + (1 - \exp(-\Delta t/\tau))\, \nabla_j F^n$$

thus making the scheme very efficient also at the interfaces.

HOFKS with Domain Decomposition

Breakdown criterium [Kolobov et al, JCP 231 (2012)]

Define in Ω, depending on macroscopic velocity $\boldsymbol{U} = (u, v, w)^t$ and pressure p

$$S(\boldsymbol{X}) = \tau \sqrt{\left(\frac{\nabla p}{p}\right)^2 + \frac{1}{\|\boldsymbol{U}\|^2}\left[\left(\frac{\partial u}{\partial x}\right)^2 + \left(\frac{\partial v}{\partial y}\right)^2 + \left(\frac{\partial w}{\partial z}\right)^2\right]}$$

Update system's description: fix a threshold $\beta > 0$.
Then, for every cell in space Ω_j

if $S(\boldsymbol{X}_j) < \beta$ then model A is assigned to cell Ω_j
if $S(\boldsymbol{X}_j) \geq \beta$ then model C is assigned

We chose $\beta = 10^{-3}$ for 2D simulations and $\beta = 10^{-2}$ for 3D ones

HOFKS with Domain Decomposition

“Reduced kinetic” buffer

Algorithm

- Set a contour buffer around the kinetic region (fixed size!)
  This could be done everywhere indeed
- Solve the “sparser” kinetic model
- Compare solutions and correct if necessary

Plus and something to do

+ computations can be reused
? How to choose the small grid size
? What error do we accept to tolerate
- detect when to apply this strategy

HOFKS with Domain Decomposition

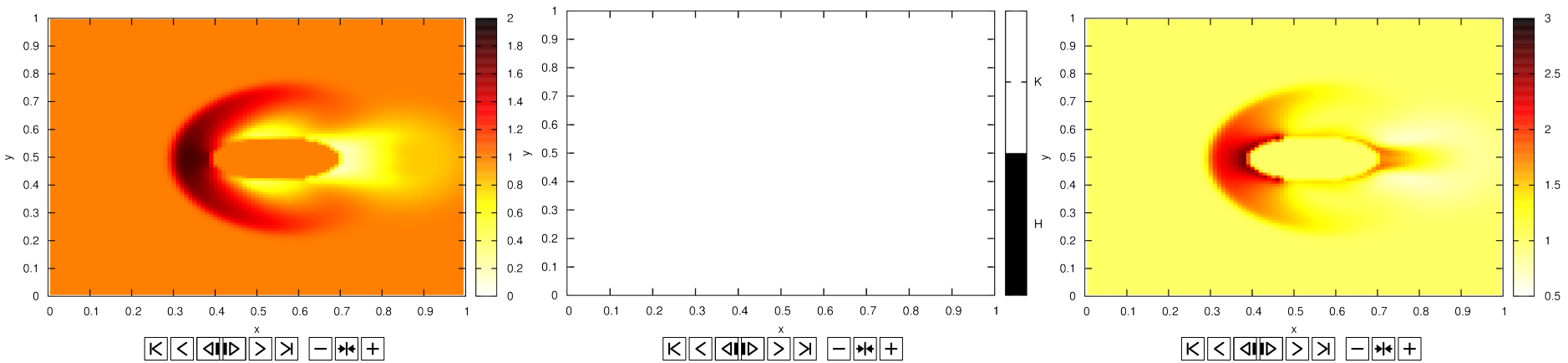
Numerical results

Threshold parameter study, I

2D wing- (ellipse-) like test case

- $[0;1]^2$ (100^2 cells), velocity space $[-5,5]^2$ (16^2 points)
- $\rho^0 = 1$, $\boldsymbol{U}^0 = (2,0)$, $T^0 = 1$ and $\tau = 10^{-3}$
- Varying $\beta = K, 10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, H$

Mass, Domain Decomposition and Temperature evolution



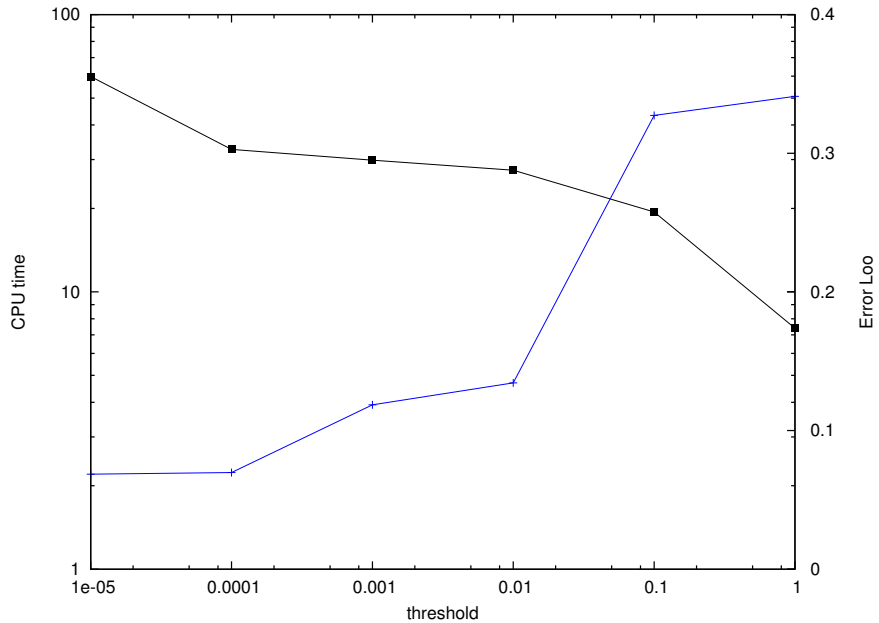
HOFKS with Domain Decomposition

Numerical results

Threshold parameter study, II

Test how β threshold influences the error (with respect to kinetic results) and the CPU time. Run kinetic, fluid models (extreme cases) and DD with different values of β

#	β	CPU (s)	Error L^1	Error L^∞
1	K	83.31	0.00	0.0000
2	10^{-5}	59.77	65.82	0.0685
3	10^{-4}	32.66	83.74	0.0697
4	10^{-3}	29.86	197.48	0.1185
5	10^{-2}	27.46	212.59	0.1344
6	10^{-1}	19.42	479.91	0.3273
7	H	7.40	575.65	0.3411



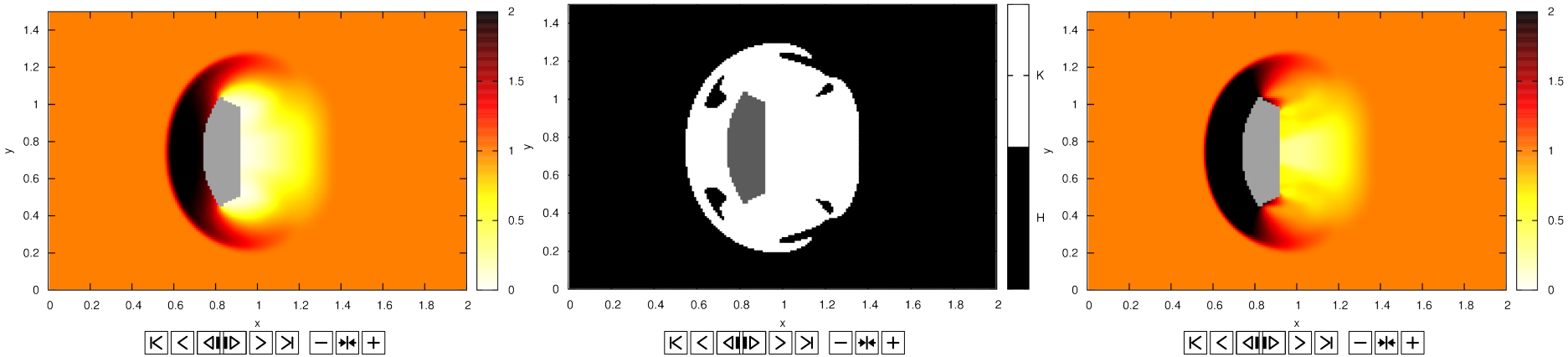
HOFKS with Domain Decomposition

Numerical results: 2D re-entry capsule

Initialization: re-entry capsule-like object

- $\Omega = [0; 2] \times [0; 1.5]$ (200 $\times$ 150 cells), velocity space $[-5, 5]^2$ (16^2 points)
- $\rho^0 = 1$, $\boldsymbol{U}^0 = (2, 0)$, $T^0 = 1$ and $\tau = 3 \cdot 10^{-4}$ Inflow on west boundary, outflow elsewhere

Mass, Domain Decomposition and Temperature evolution



Observable

- Detached shock wave to occur upfront the capsule
- Complex wave pattern behind the capsule

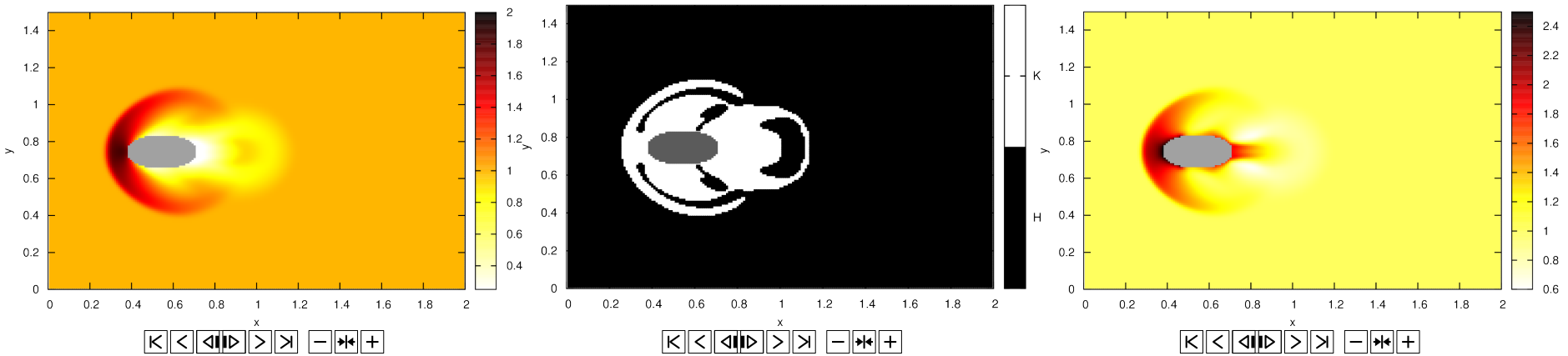
HOFKS with Domain Decomposition

Numerical results: 2D wing with varying profile

Initialization: wing-like object 1

- $\Omega = [0; 2] \times [0; 1.5]$ (200 $\times$ 150 cells), velocity space $[-5, 5]^2$ (16^2 points)
- $\rho^0 = 1$, $\boldsymbol{U}^0 = (2, 0)$, $T^0 = 1$ and $\tau = 3 \cdot 10^{-3}$. Inflow on west boundary, outflow elsewhere

Mass, Domain Decomposition and Temperature evolution



Observable

- Detached shock wave to occur upfront the wing
- Complex wave pattern behind. Asymmetric flow.

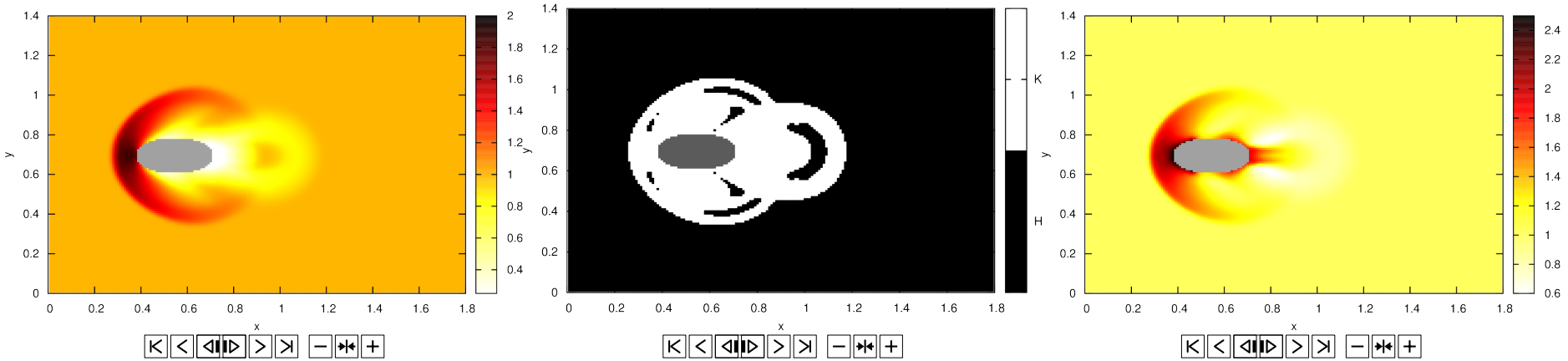
HOFKS with Domain Decomposition

Numerical results: 2D wing with varying profile + vortex

Initialization: wing-like object 2

- $\Omega = [0; 2] \times [0; 1.5]$ (200 $\times$ 150 cells), velocity space $[-5, 5]^2$ (16^2 points)
- $\rho^0 = 1$, $\boldsymbol{U}^0 = (2, 0)$, $T^0 = 1$ and $\tau = 10^{-5}$. Inflow on west boundary, outflow elsewhere

Mass, Domain Decomposition and Temperature evolution



Observable

- Detached shock wave to occur upfront the wing
- Complex wave pattern behind. Asymmetric flow and vortexes appear

HOFKS with Domain Decomposition

Numerical results: 3D Sod test

Initialization: 3D Sod test

- $\Omega = [0; 1]^3$ (100^3 cells), velocity space is $[-10, 10]^3$ (13^3 points), $\tau = 2.5 \times 10^{-4}$
- $\rho_L = 1$, $\boldsymbol{U}_L = \mathbf{0}$, $T_C = 5$
- $\rho_R = 0.125$, $\boldsymbol{U}_R = \mathbf{0}$, $T_R = 4$
- Reflecting or outflow boundary conditions

Times table

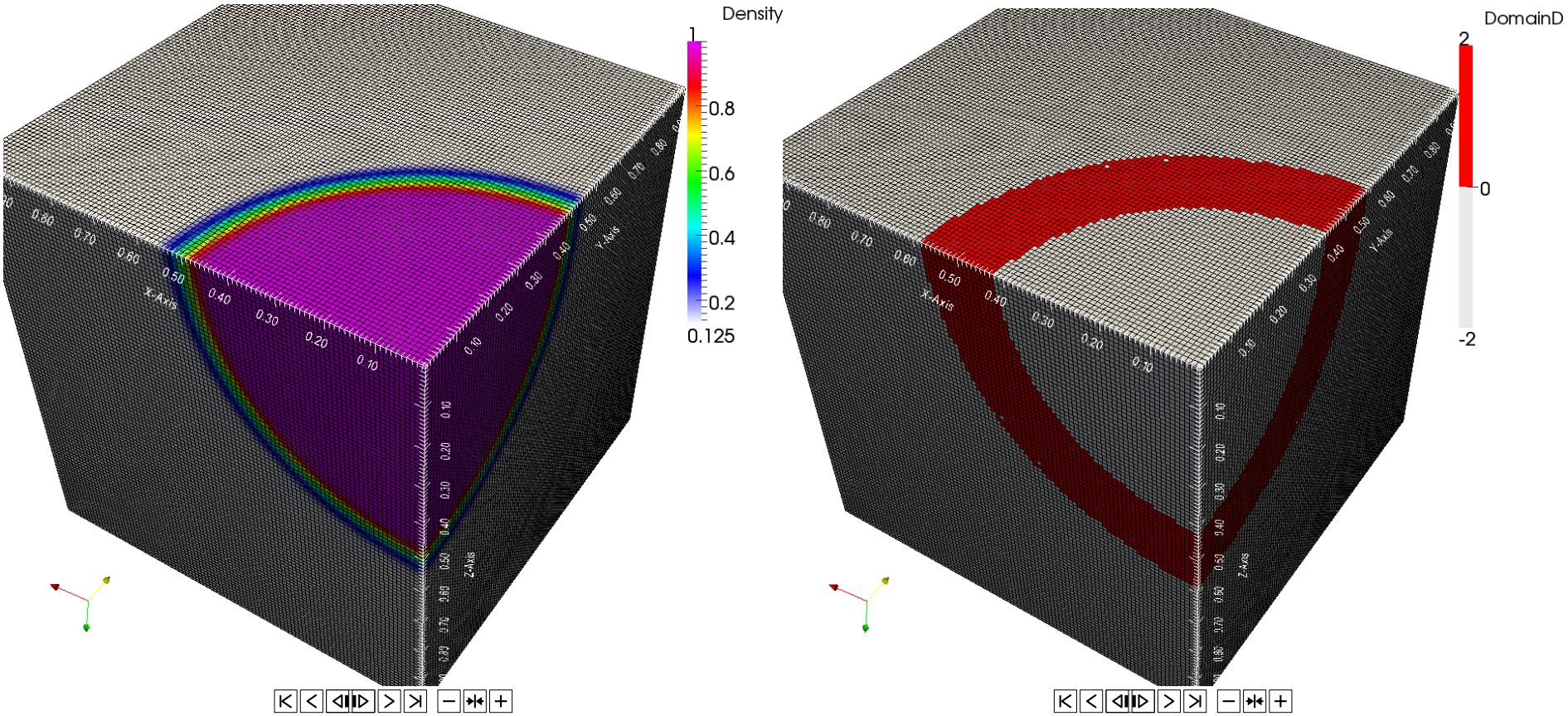
N_v	$\mathcal{V}$	**Cell #** ($N_x \times N_v$)	**Cycle**	**Time**	**Time/cycle**	**Time/cell**	**Ratio**
13^3	$[-10, 10]^3$	**KINETIC** $100^3 \times 13^3$	250	29.15h	419.7s	418×10^{-6}	—
		DOMAIN DEC. $100^3 \times 13^3$	250	3.87h	55.80s	56×10^{-6}	7.5
		HYDROD. 100^3	191	0.25h	4.72s	5×10^{-6}	89

Gain in terms of memory is also very important

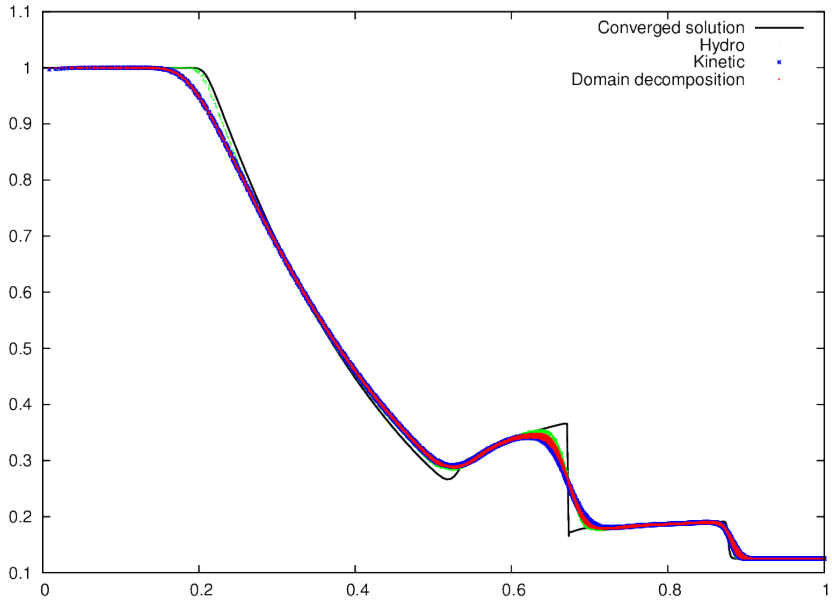
HOFKS with Domain Decomposition

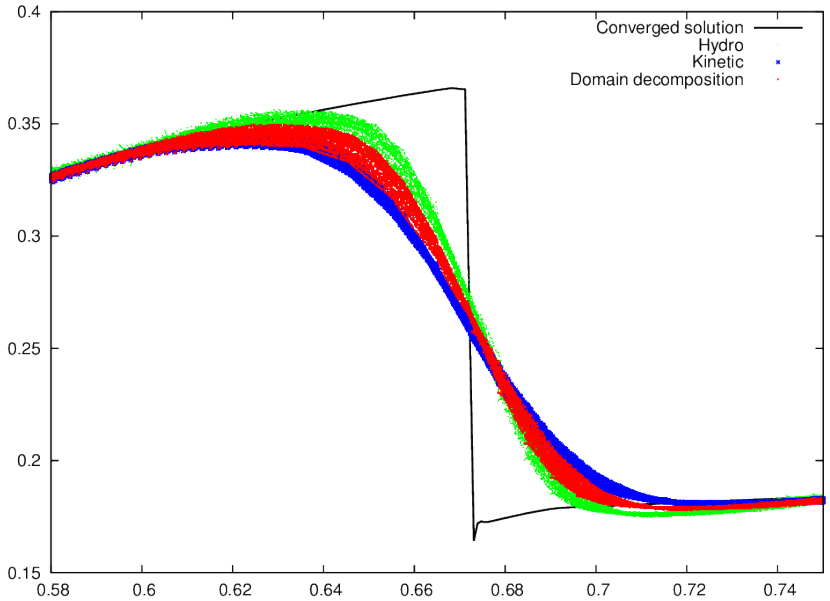
Numerical results: 3D explosion problem

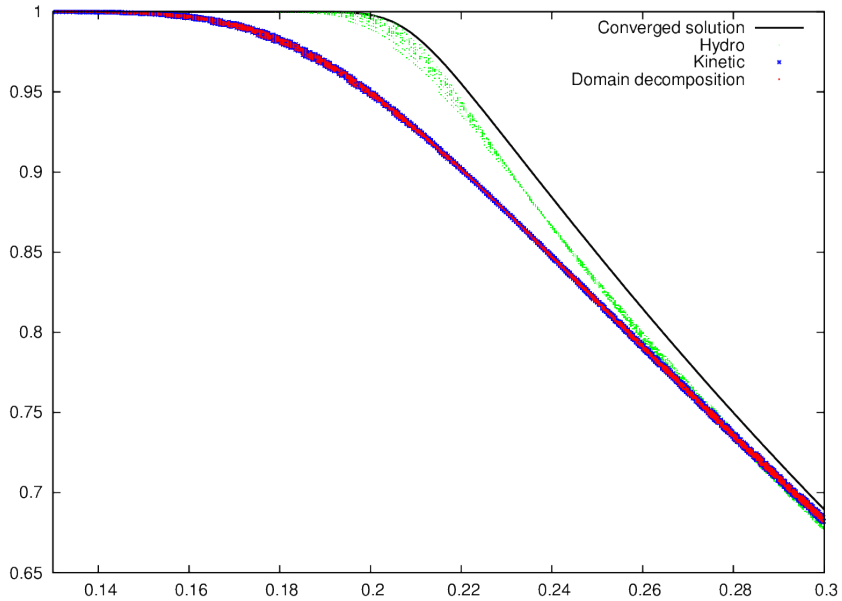
Mass density and Domain Decomposition evolution

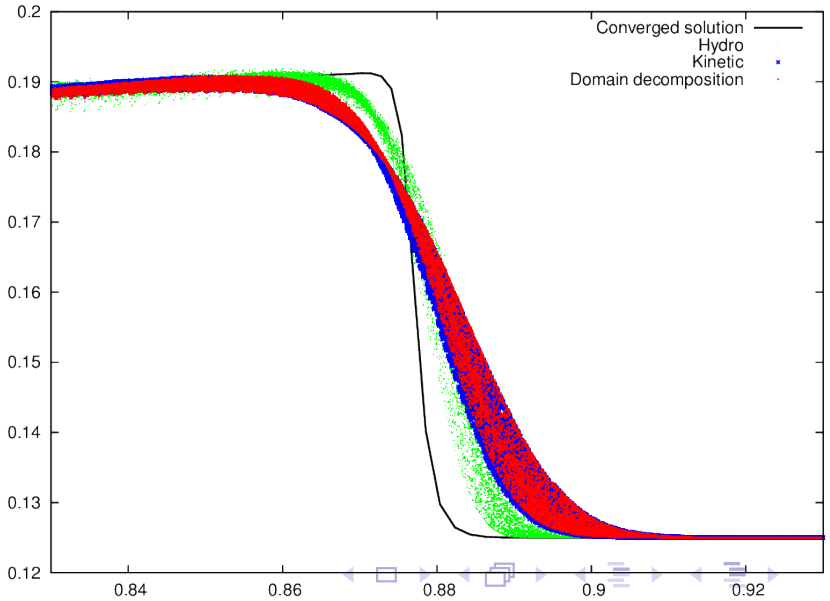


HOFKS with Domain Decomposition









HOFKS with HPC

Extensions 3

High Performance computing: parallelization with OpenMP and CUDA (GPU)

- No matter how well designed a 3D$\times$3D kinetic scheme is, it is not possible to rely only on sequential machines.
- Parallelization using OpenMP and CUDA (on GPUs) is to be implemented to show how //-friendly the scheme is.

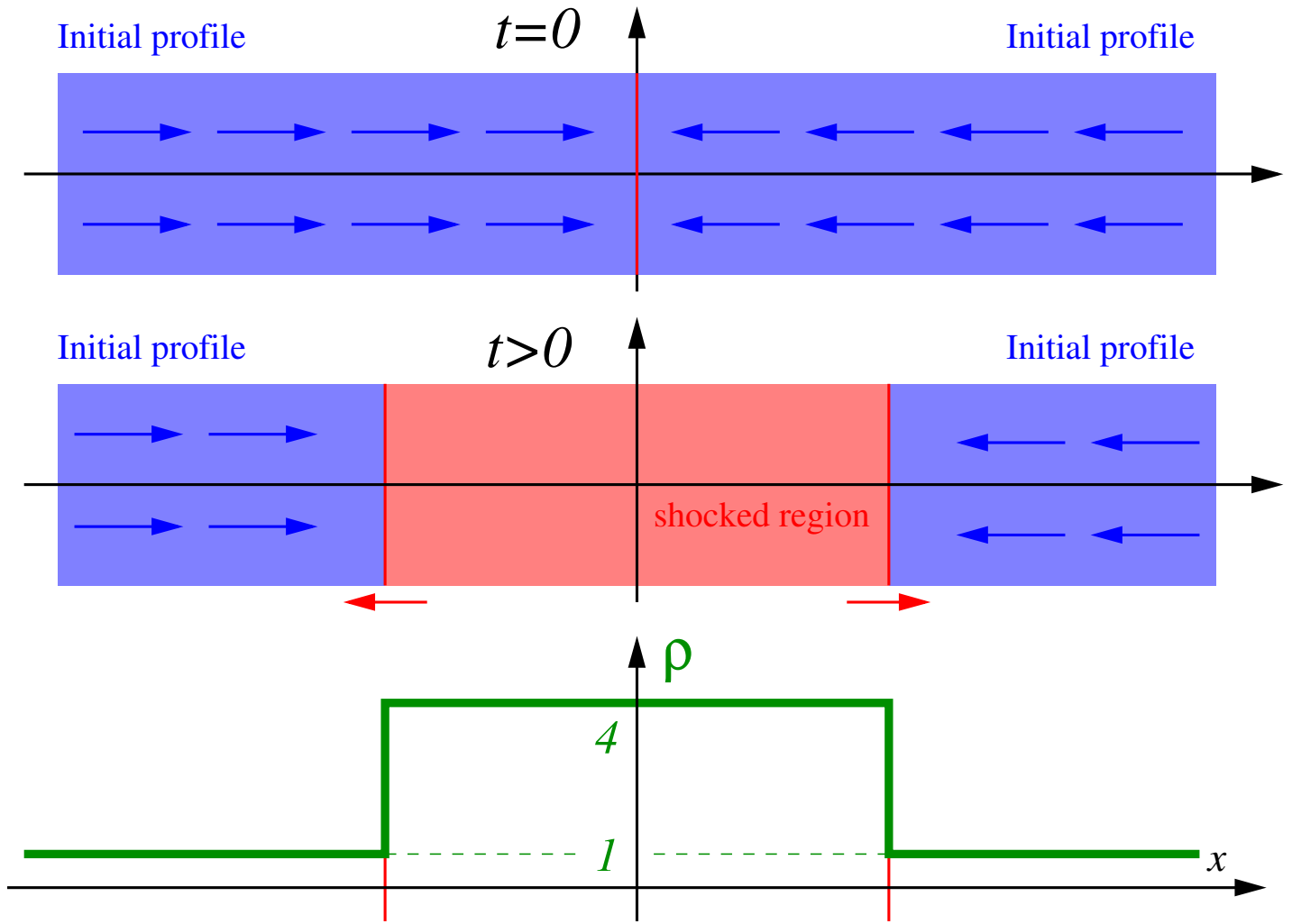
Towards an ultra efficient kinetic scheme. Part III: High-Performance-Computing, G. Dimarco, R. Loubère and J. Narski, submitted to JCP (2014)

HOFKS codes

Comparison of 3 versions of the code: Sequential, OpenMP parallel and CUDA parallel

Adaptive Velocity Mesh Refinement

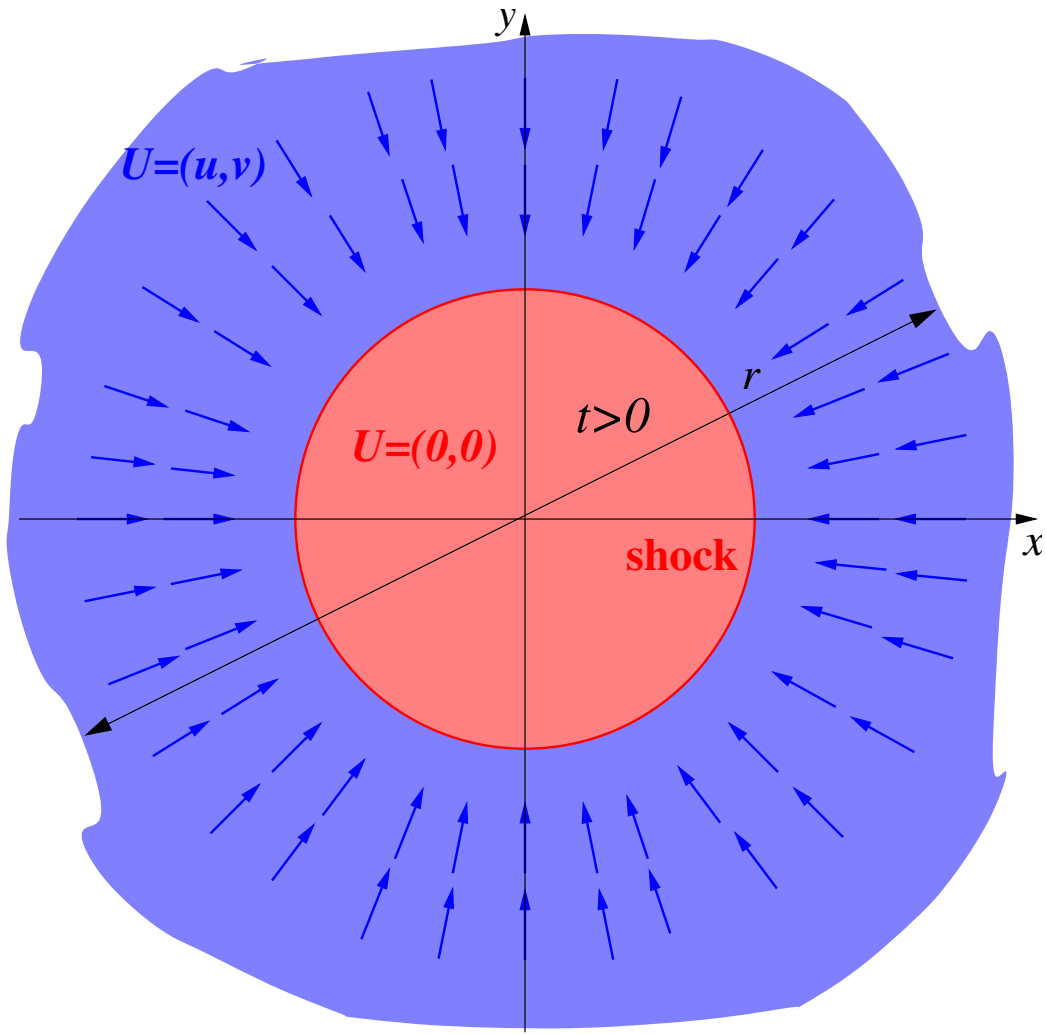
Work really in progress... (the Noh test in 1D)



Adaptive Velocity Mesh Refinement

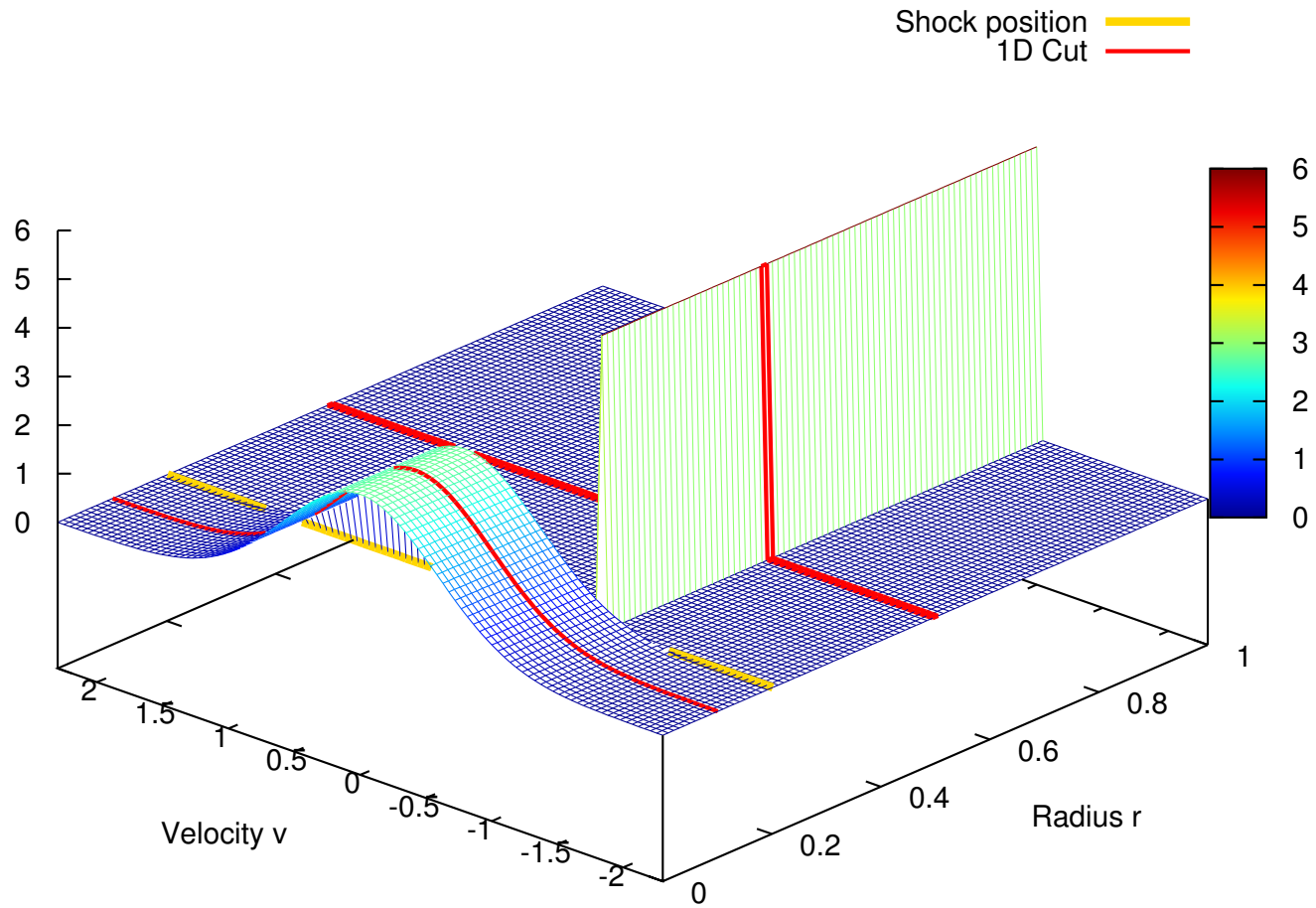
Work really in progress...

- Noh test in 2D



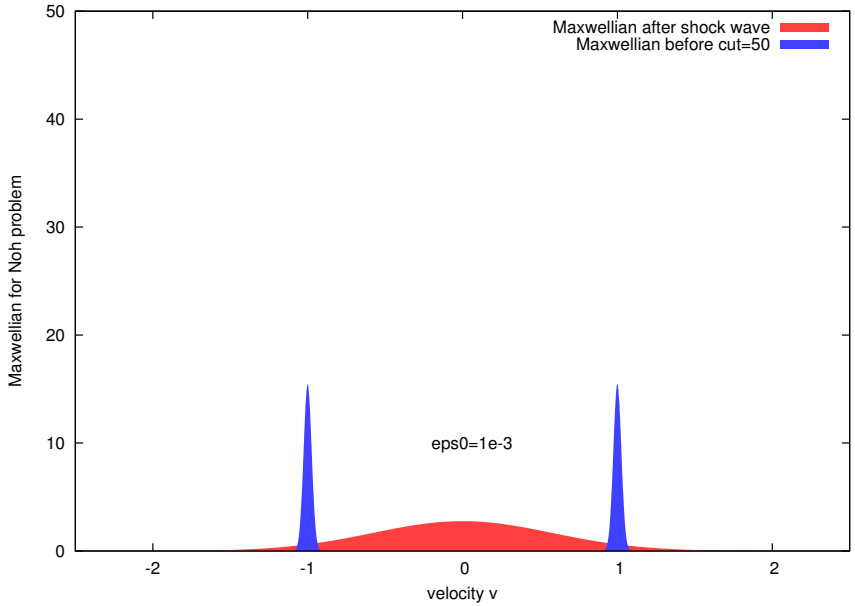
Adaptive Velocity Mesh Refinement

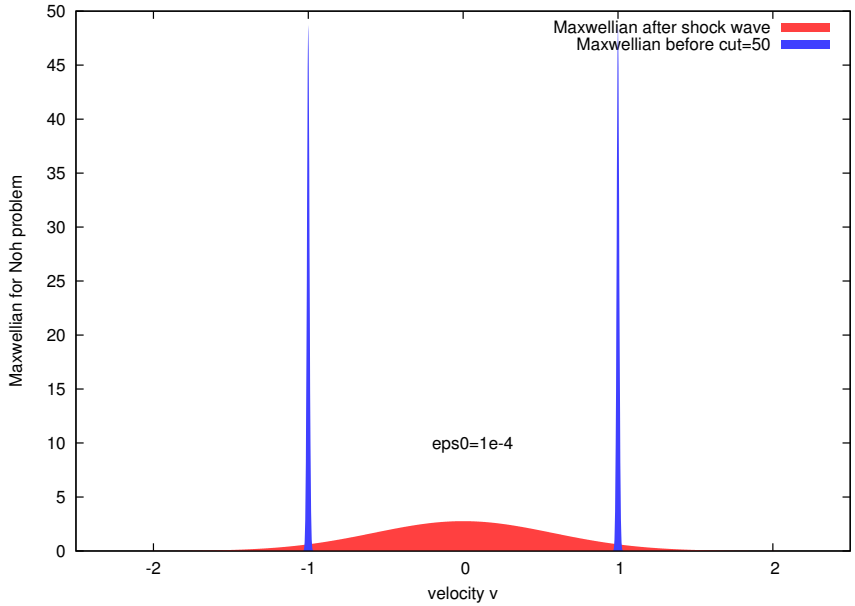
The solution in the $x - v$ phase space

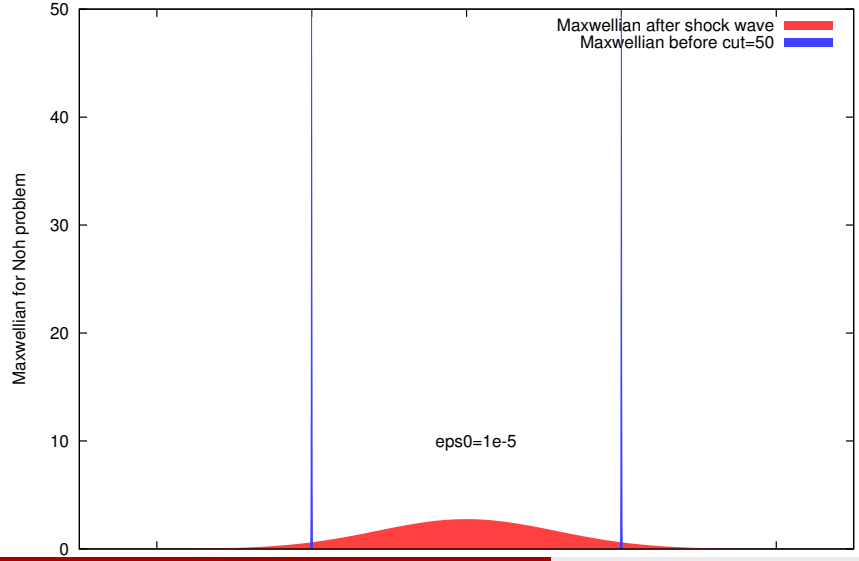


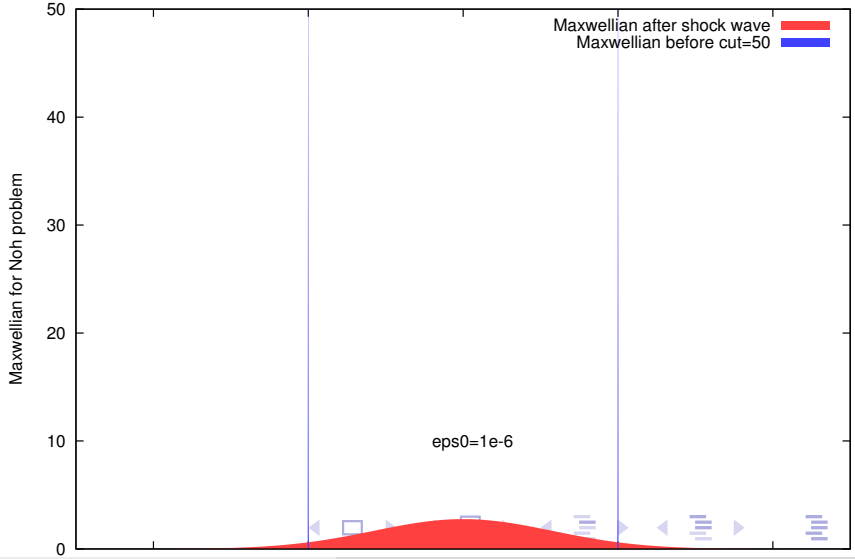
Adaptive Velocity Mesh Refinement

Solution Maxwellians for different temperatures





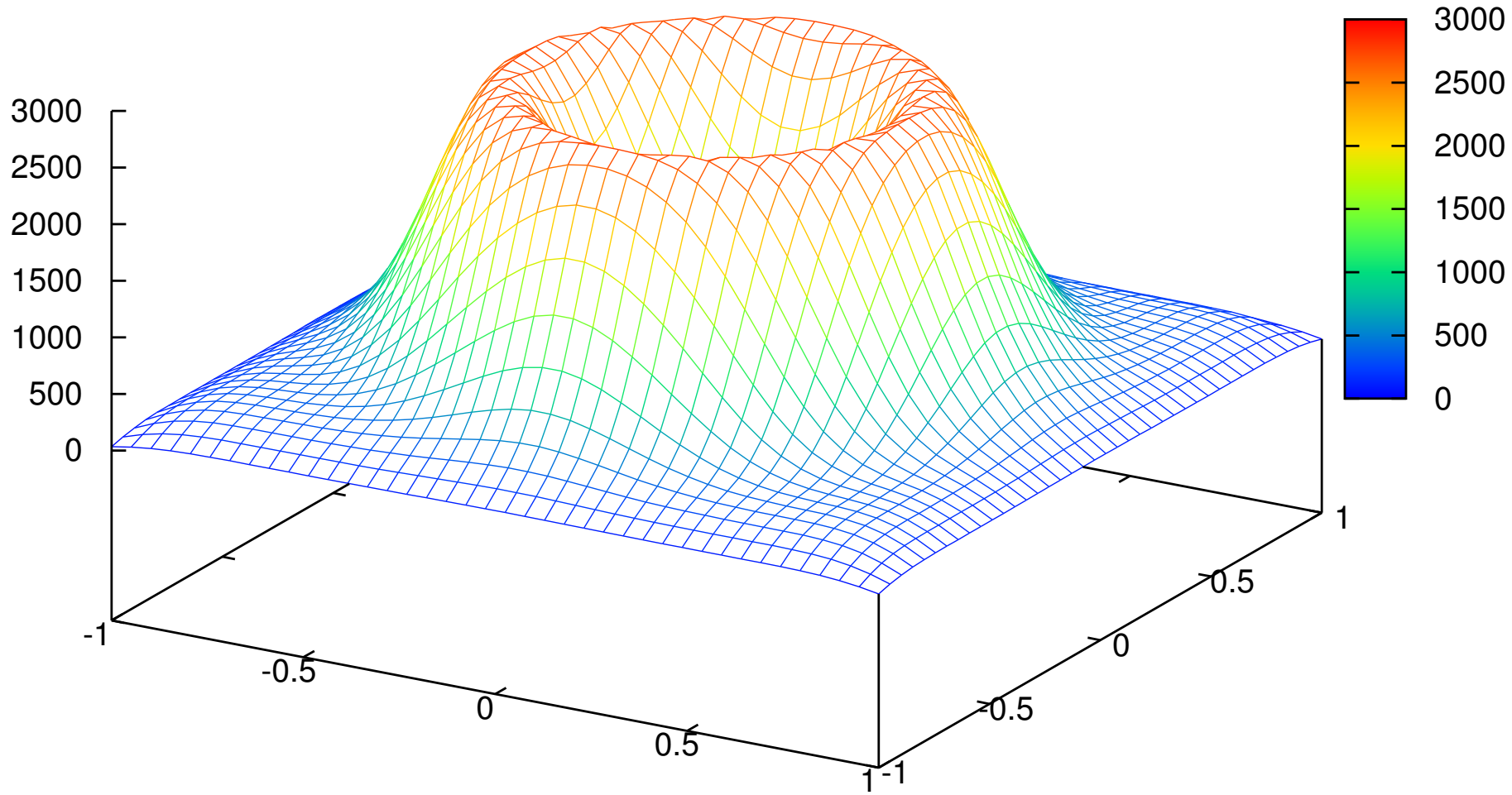




Adaptive Velocity Mesh Refinement

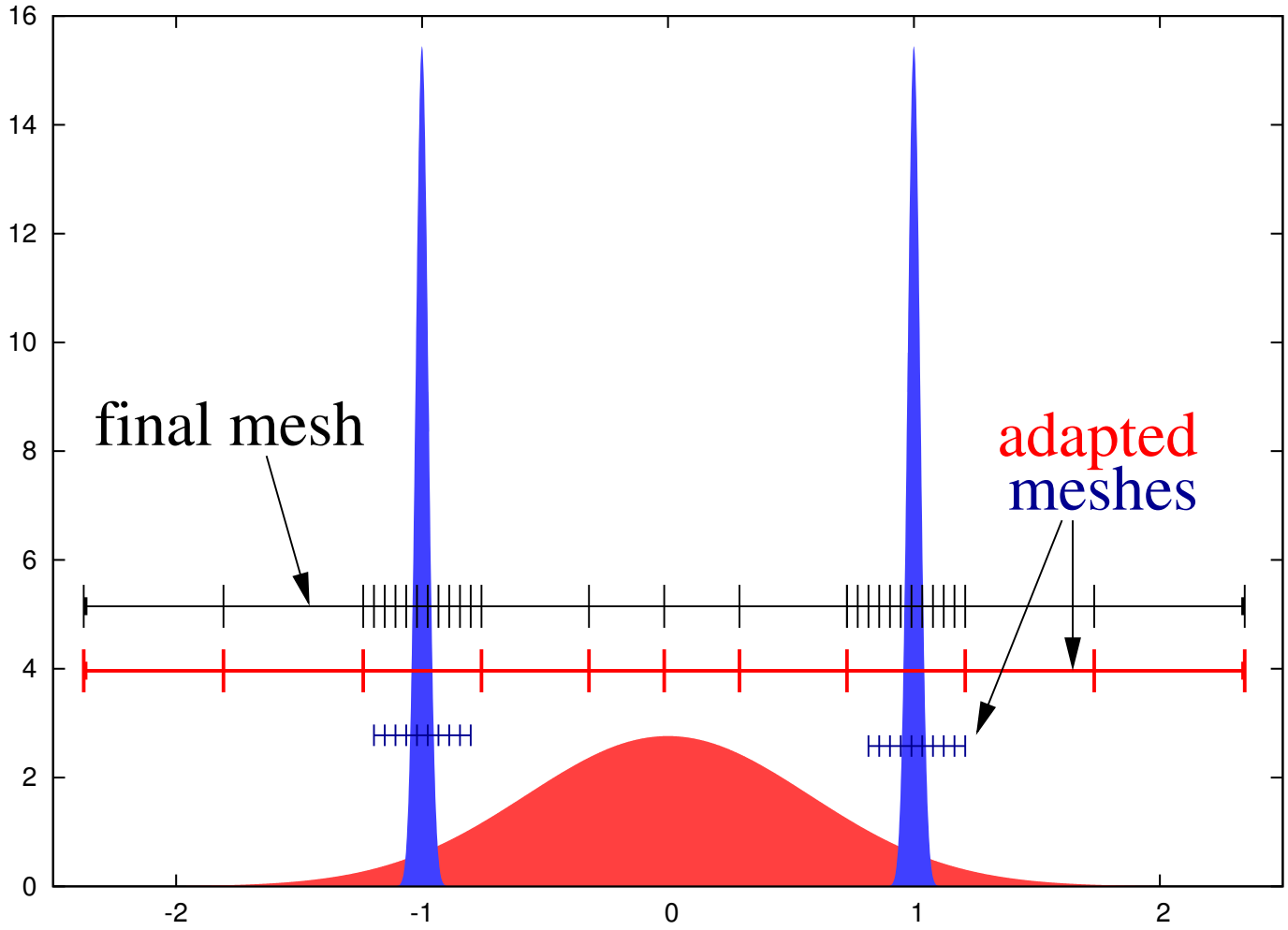
What happens in 2D

Initial condition for the Noh test in 2D:
velocity distribution (Maxwellian type) at some point



Adaptive Velocity Mesh Refinement

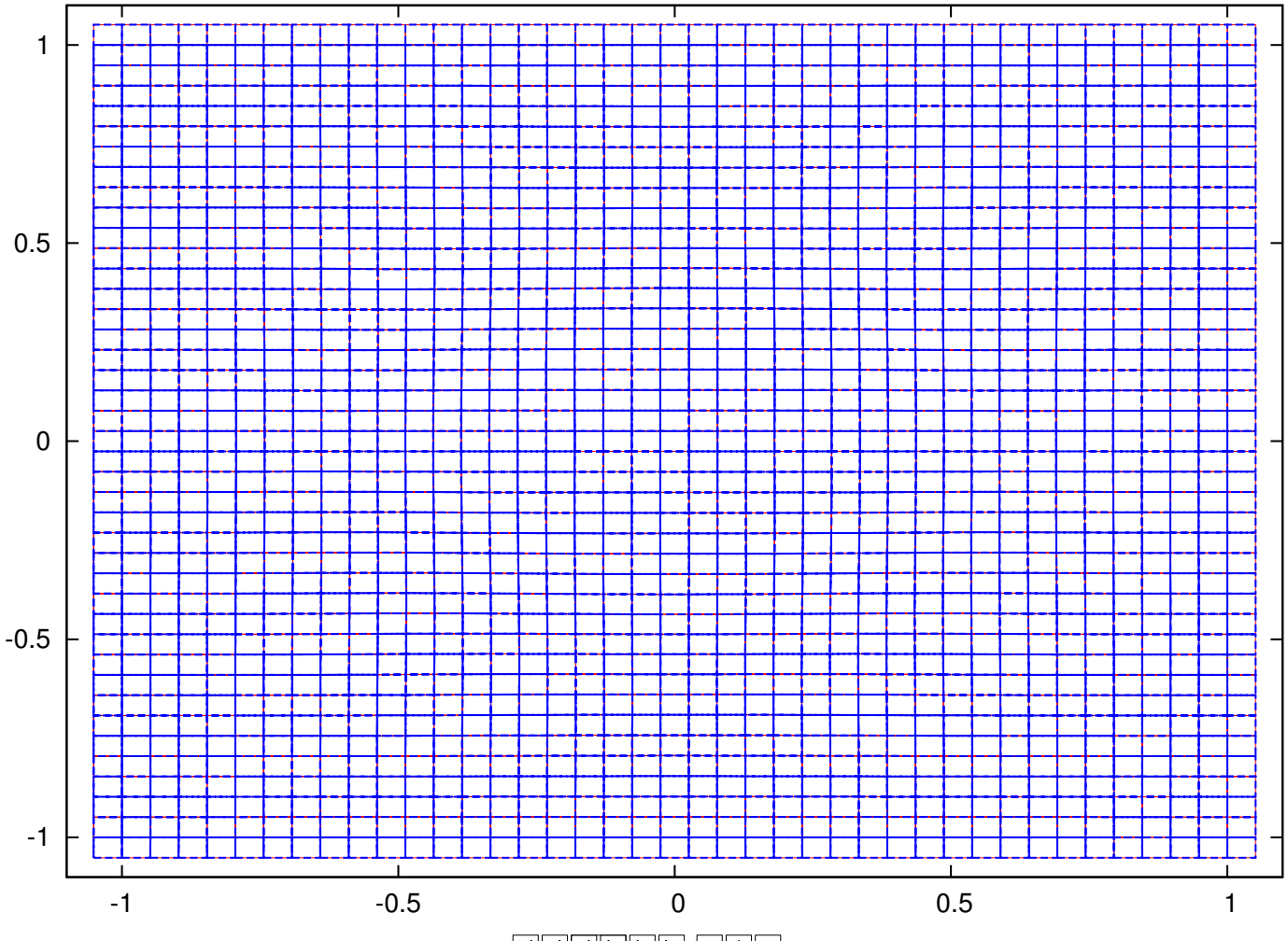
Problem: cover the full domain with the finer grid requires $N_v > 5000 \Rightarrow$ Adaptive grid



Adaptive Velocity Mesh Refinement

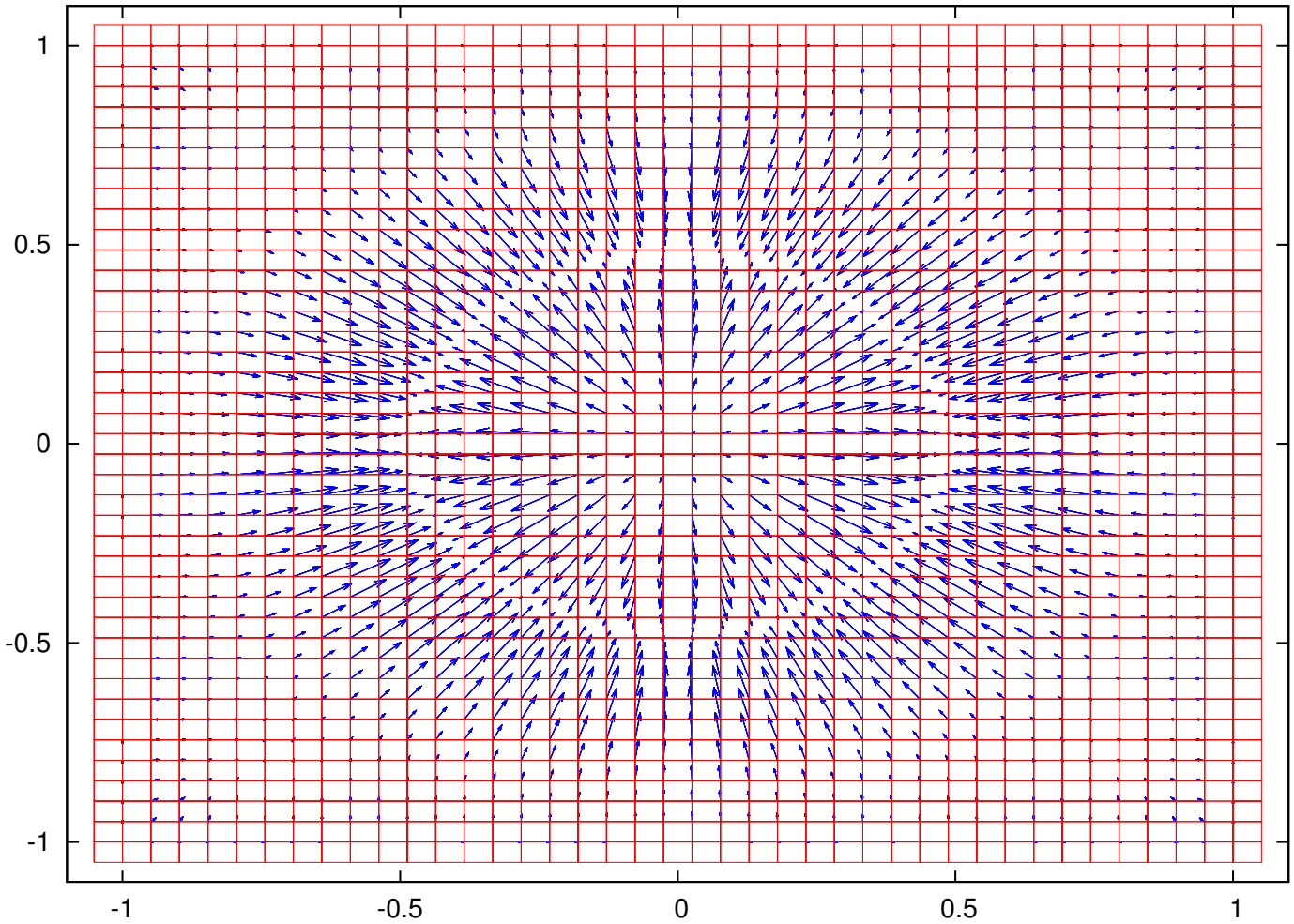
Moving the grid

Velocity cells concentrate in regions of stronger gradients



Adaptive Velocity Mesh Refinement

Cell velocities



Perspectives

What to do

- going Higher Order
- solve a wider class of models (beyond BGK): Vlasov, Boltzmann, etc.
- simulate more realistic problems $\leftrightarrow$ optimization in velocity space
- more computational optimization

Problems

- admissible techniques must be efficient and accurate
- mandatory parallel implementation

Solutions (in theory!)

- dynamic setting of the velocity numerical domain (similar to AMR)
- collisions more complex than BGK
- High Order might help
- ideas??