Task parallel implementation of a DG solver for implicit kinetic relaxation schemes applied to fluid systems

J. Badwaik, David Coulette, E. Franck, P. Helluy, C. Hillairet, H. Oberlin, M. Mehrenberger, L. Mendoza L. Navoret

IRMA Strasbourg & Inria TONUS, France

INRIA - Plasma workgroup
Nice - 2017-05-18

Outline

Section 1

Context and overall approach

Target problem

Typical problem

Unknowns "macroscopic" fields $U(t,\mathbf{x}) : \mathbb{R}^{d+1} \to R^{n_c}$ macroscopic fields.
System of (mostly) hyperbolic conservation laws

$$\partial_t U + \partial_\alpha F^\alpha(U) = \left\{ \partial_\alpha \left[\mathcal{D}^{\alpha\beta}(U) \partial_\beta U \right] + \mathcal{S} \right\}$$

F^α : fluxes (nonlinear) - $\mathcal{D}$: parabolic terms (diffusion),$\mathcal{S}$ source term
Systems of interest : Euler/Navier Stokes - MHD

Numerical challenges

- explicit schemes
  - CFL conditions : time scale constrained by space grid
  - forces to resolve possibly unwanted fast times scales
- implicit schemes
  - large nonlinear system
  - costly matrix assembly/storage/inversion.

Xin-Jin flux relaxation method

Flux relaxation scheme

- each flux component is replaced by an additional new unknown W^α.
- W^α linear transport (acoustic like system)
- relaxation of V^α towards actual flux.

$$\left\{ \begin{array}{ll} \partial_t U + \partial_\alpha W^\alpha & = 0 \\ \partial_t W^\alpha + A^{\alpha\beta}\partial_\beta U & = \frac{R}{\varepsilon}(F^\alpha(U) - W^\alpha) \end{array} \right. \quad (1)$$

- operator splitting : linear transport step + local relaxation step.

A divide and conquer strategy

- larger system $n_c \times (d+1)$ but
  - non-locality is solved linearly (wave system)
  - non-linearity is local
- Caveats :
  - the splitting error generates diffusion/dispersion
  - all variables are still coupled in the wave system.

Diagonalizing transport - the 1D case

Starting from the relaxation system

$$\partial_t \begin{pmatrix} U \\ V \end{pmatrix} + \underbrace{\begin{pmatrix} 0 & 1 \\ a^2 & 0 \end{pmatrix}}_{T} \partial_x \begin{pmatrix} U \\ V \end{pmatrix} = \begin{pmatrix} 0 \\ \frac{R}{\varepsilon}(F(U) - V) \end{pmatrix}$$

Transport matrix T has eigenvalues $\lambda_\pm = \pm|a|$ with eigenvectors $f_\pm \propto [1, \pm a]^t$.

$$\begin{pmatrix} U \\ V \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ a & -a \end{pmatrix} \begin{pmatrix} f_+ \\ f_- \end{pmatrix} \quad \begin{pmatrix} f_+ \\ f_- \end{pmatrix} = \frac{1}{2a} \begin{pmatrix} a & 1 \\ a & -1 \end{pmatrix} \begin{pmatrix} U \\ V \end{pmatrix}$$

so that the system on $f = [f_+, f_-]$ reads

$$\partial_t \begin{pmatrix} f_+ \\ f_- \end{pmatrix} + \begin{pmatrix} a & 0 \\ 0 & -a \end{pmatrix} \partial_x \begin{pmatrix} f_+ \\ f_- \end{pmatrix} = \begin{pmatrix} \frac{R}{\varepsilon} & 0 \\ 0 & \frac{R}{\varepsilon} \end{pmatrix} \begin{pmatrix} f_+^{eq}(U) - f_+ \\ f_-^{eq}(U) - f_- \end{pmatrix},$$

with the equilibrium functions $f_\pm^{eq}(U) = \frac{U}{2} \pm \frac{F(U)}{2a}$.

- the transport operator is diagonal
- simple advection with constant coefficient (no acoustics)
- in a op. splitting scheme f_+ and f_- might be transported independently.

Generic discrete kinetic relaxation scheme

Kinetic relaxation system [Nat98] [Bou99]

$$\partial_t f + \partial_\alpha V^\alpha f = \frac{\Omega}{\varepsilon}(f^{eq}(Pf) - f)$$

- a vector f of n_v kinetic fields $f_i(t, \mathbf{x}), \mathbf{x} \in \mathbb{R}^d$
- n_v constant velocities $\mathbf{v}_i \in \mathbb{R}^d$ and d diagonal matrices $V^\alpha, \alpha = x, y, \ldots$ with $V^\alpha_{ij} = \delta_{ij} v^\alpha_i$
- a $n_c \times n_v$ ($n_c < n_v$) matrix P mapping f to the macroscopic fields vector $U = Pf$, with $U \in \mathbb{R}^{n_c}$.
- an equilibrium vector operator $f^{eq} : \mathbb{R}^{n_c} \to \mathbb{R}^{n_v}$ such that $Pf^{eq}(Pf) = Pf, \forall f$.
- a $n_v \times n_v$ relaxation matrix Ω
- U conserved during relaxation i.e $P\Omega g = 0, \forall g \in \mathrm{KerP}$.

- Applying P and $\varepsilon \to 0$: $\partial_t(Pf^{eq}(Pf)) + \partial_\alpha PV^\alpha f^{eq}(Pf) = 0$
- consistent with $\partial_t U + \partial_\alpha F^\alpha(U) = 0$ provided

$$\mathcal{C}\left\{\begin{array}{rl} Pf^{eq}(U) &= U \\ PV^\alpha f^{eq}(U) &= F^\alpha(U), \alpha = x, y \ldots \end{array}\right.$$

1D simple example - Euler with standard $D1Q3$ scheme

Model notation : $DdQq$ from Lattice Boltzmann Method scheme with $q = n_v$
Macroscopic system : still Euler isothermal

$$\left\{ \begin{array}{rl} \partial_t \rho + \partial_x q & = 0 \\ \partial_t q + \partial_x\left(\frac{q^2}{\rho} + \rho c^2\right) & = 0 \end{array} \right.$$

- $n_c = 2$ macroscopic fields $U = (\rho, q = \rho u)$.
- $n_v = 3$ the velocity set is $(-\lambda, 0, \lambda)$.
- $V^x = \mathrm{diag}[\lambda, 0, -\lambda]$
- $P = \begin{pmatrix} 1 & 1 & 1 \\ -\lambda & 0 & \lambda \end{pmatrix}$ $PV^x = \begin{pmatrix} -\lambda & 0 & \lambda \\ \lambda^2 & 0 & \lambda^2 \end{pmatrix}$
- note that $\mathrm{rank}(\mathcal{C}) = 3$.
- equilibrium function
  $f^{eq} = \left[\frac{\rho}{2\lambda^2}\left(u^2 + c^2 - \lambda u\right), \frac{\rho}{\lambda^2}\left(\lambda^2 - u^2 - c^2\right), \frac{\rho}{2\lambda^2}\left(u^2 + c^2 + \lambda u\right)\right]^t$

1D simple example- Euler with "vectorial" simplicial scheme $2 \times D1Q2$

Macroscopic system

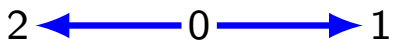
$$\begin{cases} \partial_t \rho + \partial_x q &= 0 \\ \partial_t q + \partial_x (\frac{q^2}{\rho} + \rho c^2) &= 0 \end{cases}$$

- $n_c = 2$ macroscopic fields $U = (\rho, q = \rho u)$.
- $n_v = 4$, for each scalar field the velocity set is $(-\lambda, \lambda)$.
- $P = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{pmatrix}$
- $V^x = \mathrm{diag}[\lambda, -\lambda, \lambda, -\lambda]$
- equilibrium function (generic form) : $f^{eq}_{k,i} = \frac{U_k}{2} + \frac{1}{2\lambda^2} F_k(U) \cdot v_{k,i}$.
- equilibrium function $f^{eq} = \left[\frac{\rho}{2} + \frac{q}{2\lambda}, \frac{\rho}{2} - \frac{q}{2\lambda}, \frac{q}{2} + \frac{q^2/\rho + \rho c^2}{2\lambda}, \frac{q}{2} - \frac{q^2/\rho + \rho c^2}{2\lambda} \right]^t$
- adding a stationary $(v = 0)$ nodes for each scalar quantity is straightforward. With arbitrary $f^{eq}_{\rho,0}(U), f^{eq}_{\rho,0}(U)$ we have

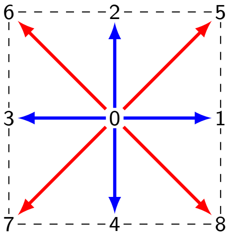
$$f^{eq} = \left[\frac{\rho - f^{eq}_{\rho,0}}{2} + \frac{q}{2\lambda}, \frac{\rho - f^{eq}_{\rho,0}}{2} - \frac{q}{2\lambda}, \frac{q - f^{eq}_{q,0}}{2} + \frac{q^2/\rho + \rho c^2}{2\lambda}, \frac{q - f^{eq}_{q,0}}{2} - \frac{q^2/\rho + \rho c^2}{2\lambda} \right]^t$$

Standard Lattice Boltzmann models

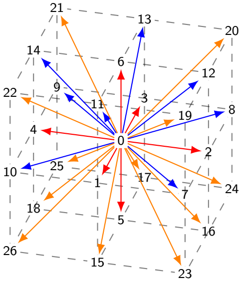
Notation : DdQq with $d = 1,2,3$ space dimension q number of velocities.



D1Q3



D2Q9

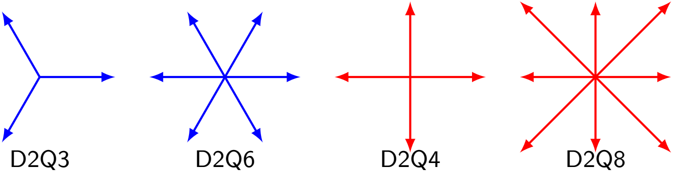


D3Q27

- P built from low-order polynomials : macro quantities are moments.
- splitting error generates diffusive terms : can mimic physical diffusion
- Can be applied to any hyperbolic system of conservations laws (fluid mechanics [CD98], Maxwell [Gra14], MHD [Del02], etc.)
- transport is exact : regular mesh generated from the velocity set (translation group)
- transport is easy but $\Delta t / \Delta x$ linked : integer CFL-like condition

"Vectorial" kinetic model

- Rationale : for some hyperbolic system of interest (eg MHD), macroscopic variables cannot be considered as moments of various order of the same discrete kinetic set (ie. $\mathcal{C}$ is structurally unsolvable).
- Basic idea : one (small) set of q velocities $\mathbf{v}_{k,i}$ per scalar macroscopic field U_k.
- the velocity sets need not be the same for each U_k.
- P has a block diagonal structure : $1 \leq k \leq n_c$: $U_k = \sum_{i=1}^{q_k} f_{k,i}$. (all scalar fields are 0^{th} order moments.
- examples:
  - minimal set for solvability : a d simplex, $q = d+1$ velocities. (D1Q2,D2Q3,D3Q4)
  - velocity pairs along cartesian axes : $q = 2d$ (D1Q2; D2Q4,D3Q6)
- more complex sets can be obtained by adding velocities : "arbitrary" , zero velocity node, scaled, rotated sets...
- strike a balance between number of velocities and artifacts (anisotropy...)



Chapman-Enskog expansion

Micro/Macro splitting : we write $f = f^{eq}(U) + g$, with $U = Pf \in \mathbb{R}^{n_c}$ and $g \in \mathrm{KerP}$.

$$\begin{cases} \partial_t U + \partial_\alpha P V^\alpha f^{eq}(U) = -\partial_\alpha P V^\alpha g \\ \partial_t g + \partial_\alpha V^\alpha g = -\Omega g - [\partial_U f^{eq} \partial_t U + V^\alpha \partial_U f^{eq} \partial_\alpha U] \end{cases}$$

Expansion up to first order of $U = U_0 + \varepsilon U_1 + \mathcal{O}(\varepsilon^2)$ and $g = \varepsilon g_1 + \mathcal{O}(\varepsilon^2)$ yields

$$\partial_t U + \partial_\alpha F^\alpha(U) = \varepsilon \partial_\alpha \left\{ \underbrace{P V^\alpha \Omega^{-1} \left[V^\beta \partial_U f^{eq} - \partial_U f^{eq} \partial_U F^\beta \right]}_{D^{\alpha\beta}} \partial_\beta U \right\} + \mathcal{O}(\varepsilon^2)$$

In the BGK case ($\Omega = I$) the diffusion tensor reads

$$D^{\alpha\beta} = P V^\alpha V^\beta \partial_U f^{eq} - \partial_U F^\alpha \partial_U F^\beta$$

- dissipation criterion $X^t D^{\alpha\beta} X \geq 0, X \in \mathbb{R}^{n_c}$ constrains the velocity scale λ of V^α.
- the diffusion term of a discrete first order time implicit scheme has the same structure.
- at finite but small ε , $D^{\alpha\beta}$ can be used to mimic (small) physical diffusion.

Comparison with Xin-Jin scheme

For $d > 1$ is the kinetic system still equivalent to the Xin-Jin relaxation scheme ?

- Compare Xin-Jin system on the variables/flux pairs U, W^α to the kinetic system on $(Pf, PV^\alpha f)$ assuming only $Pf^{eq}(U) = U$ and $PV^\alpha f^{eq}(U) = F^U$.
- equation on the conserved variables

$$\partial_t U + \partial_\alpha W^\alpha = 0 \quad vs \quad \partial_t Pf + \partial_\alpha PV^\alpha f$$

- equation on the discrete flux variables

$$\partial_t W^\alpha + B^{\alpha\beta}\partial_\beta U = \frac{R}{\varepsilon}(F^\alpha(U) - W^\alpha) \quad vs \quad \partial_t(PV^\alpha f) + \partial_\beta PV^\alpha V^\beta f = \frac{PV^\alpha \Omega}{\varepsilon}(f^{eq} - f)$$

- diffusion term from Chapman-Enskog expansion $\varepsilon\partial_\alpha\left[D^{\alpha\beta}\partial_\beta U\right]$

$$D^{\alpha\beta}(U) = \Omega^{-1}\left[a^2 B^{\alpha\beta} - \partial_{\mathbf{U}} F^\alpha \partial_{\mathbf{U}} F^\beta\right] \quad vs \quad D^{\alpha\beta}(U) = PV^\alpha\Omega^{-1}[V^\beta\partial_{\mathbf{U}} f^{eq} - \partial_{\mathbf{U}} f^{eq}\partial_{\mathbf{U}} F^\beta]$$

Remarks

- overall structure is similar
- all the transport structure is fixed by the choice of velocity basis.
- dissipation criterion on velocity scale λ.

Back to our example

$D1Q3$ model

$$D = \begin{bmatrix} 0 & 0 \\ 2u(u^2 - c^2) & \lambda^2 - c^2 - u^2 \end{bmatrix}$$

Dissipation condition

$$\lambda^2 \geq c^2 + u^2$$

$2D1Q2$ vectorial model

$$D = \begin{bmatrix} -c^2 + 2\lambda^2 + u^2 & -2u \\ 2u\left(-c^2 + u^2\right) & -c^2 + 2\lambda^2 - 3u^2 \end{bmatrix}$$

Dissipation condition

$$\lambda^2 \geq (c \pm u)^2$$

Trade-off on the choice of λ

- sufficiently large to ensure dissipation (avoid anti-diffusion)
- sufficiently small to avoid over-diffusion (and dispersion...)

Section 2

Space discretization - transport scheme

Transport schemes

Whishlist

- complex geometry, curved meshes
- flexibility $h-p$ refinement
- cfl-free
- efficiency

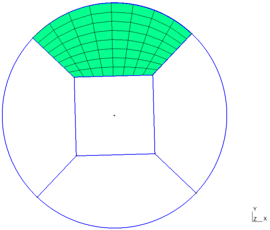
Candidates

- finite volumes
- discontinuous galerkin
- semi-lagrangian
- stochastic schemes (Glimm)

DG - Implicit upwind transport scheme 1

We consider a coarse mesh made of hexahedral curved macrocells

- Each macrocell is itself split into smaller subcells of size h.
- In each subcell L we consider polynomial basis functions ψ_k^L of degree p.
- Expansion on the polynomial basis: discontinuous approximation of f.



$$f(x, v, p\Delta t) \simeq f_L^p(x, v) = \sum_k f_{L,k}^p(v)\psi_k^L(x), \quad x \in L.$$

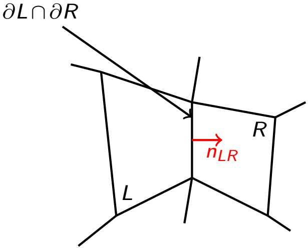
DG - Implicit upwind transport scheme 2

Implicit DG approximation scheme

$\forall L, \forall k$

$$\int_L \frac{f_L^p - f_L^{p-1}}{\Delta t} \psi_k^L - \int_L v \cdot \nabla \psi_k^L f_L^p + \int_{\partial L} \left(v \cdot n^+ f_L^p + v \cdot n^- f_R^p \right) \psi_k^L = 0.$$

- time step index: p
- R denotes the neighbor cells along ∂L.
- $v \cdot n^+ = \max(v \cdot n, 0)$,
  $v \cdot n^- = \min(v \cdot n, 0)$.
- n_{LR} is the unit normal vector on ∂L oriented from L to R.



Features

- implicit scheme , unconditionally stable , (h, p) refinement
- requires a priori the resolution of a large linear system for each v.

Getting high order in time : symmetric splitting

Bibliography: [MQ02]

Splitting schemes made of **symmetric** building blocks

All steps implemented as θ weighted schemes: setting $\theta = 1/2$ (Crank-Nicolson) $\rightarrow$ symmetric

- Transport (T)
- Macroscopic source (S)
- BGK Relaxation (R).

1^{st} order	$T(\Delta t)$	$S(\Delta t)$	$R(\Delta t)$		
2^{nd} order	$T(\Delta t/2)$	$R(\Delta t/2)$	$S(\Delta t)$	$R(\Delta t/2)$	$T(\Delta t/2)$
2^{nd} order collapsed	$R(\Delta t/2)$	$S(\Delta t)$	$R(\Delta t/2)$	$T(\Delta t)$	

Beyond second order : composition methods

Using the 2^{nd} order scheme as a basic building block, we can build higher-order time schemes using composition methods [KL97] [Suz90]
Example : Suzuki 4^{th} order fractal scheme with 5 fractional substeps $\gamma_i \Delta t$ with $\gamma_0 = \gamma_1 = \gamma_3 = \gamma_4 = (4 - 4^{1/3})^{-1}$ and $\gamma_2 = -4^{1/3}(4 - 4^{1/3})^{-1}$.

Section 3

Task-Based parallel solver

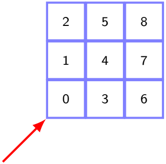
The benefits of upwinding

upwind flux $\rightarrow$ data dependencies follow the (constant) velocity

- transport operator can be cast into Block Triangular Form (BTF) by appropriate data renumbering.
- inversion : BTF + inversion of diagonal blocks.
- data blocks at the subcell scale : too small for efficient parallelism.

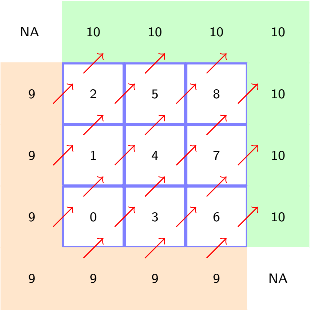
Coarse grain block structure at macrocell level

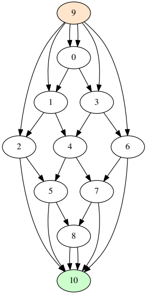
- L is *upwind* with respect to R if $v \cdot n_{LR} > 0$ on $\partial L \cap \partial R$.
- In a cell L, the solution depends only on the values of f in the upwind macrocells.



Dependency graph

For a given velocity v we can build a dependency graph. Vertices are associated to macrocells and edges to macrocells interfaces or boundaries. We consider two fictitious additional vertices: the “upwind” vertex and the “downwind” vertex.



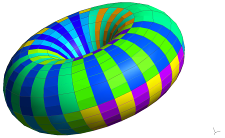


- The dependency graph yields a coarse block triangular ordering
- the local system in each macrocell is solved "on the fly" using the KLU library.
- no need to assemble, store, and factorize the global system.

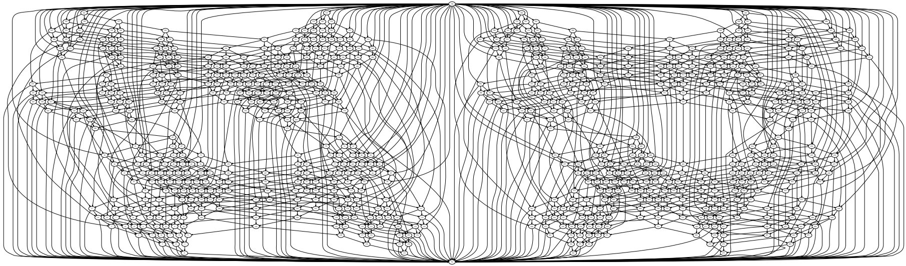
Transport solver parallelism

- ideal across velocities (uncoupled)
- across macrocells : can be high but load imbalance
- realistic mesh : complex to manage...

Toroidal mesh - 720 macrocells



Toroidal mesh - transport graph for $(1, 0, 0)$ velocity.



We need smart task scheduling

Kirsch : Task-Based parallel DG-LBM solver

Here comes StarPU(http://starpu.gforge.inria.fr)

- StarPU is a task-based scheduling library developed at Inria Bordeaux
- Task description : codelets, inputs (R), outputs (W or RW).
- The user submits tasks in a correct sequential order.
- StarPU schedules the tasks in parallel if possible.
- MPI extension easy : dispatch data and declare owner process : communications handled transparently.

SCHNAPS + StaRPU + Kinetic Schemes = KIRSCH

- starting point SCHNAPS : general DG explicit solver.
- StarPU + Optimization for Kinetic Relaxation schemes
- KIRSCH : KInetic Representation for SCHnaps

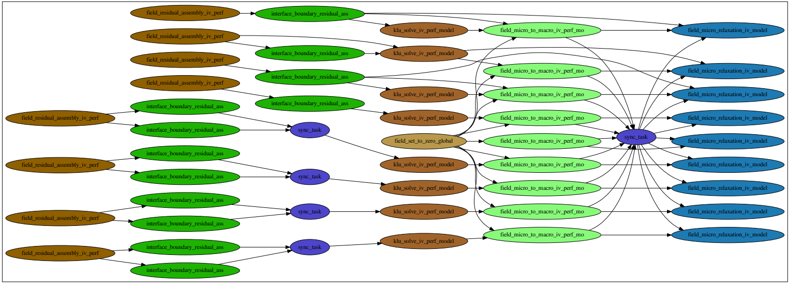
KIRSCH design

- base ops : transport(T), relaxation(R), sources (S) weighted implicit integrators.
- each op encapsulated in a StarPU codelet.
- physics in the equilibrium functions : flexibility.
- high order time schemes are built from second order blocks.

What StarPU does for us

Task graph for D2Q9 model

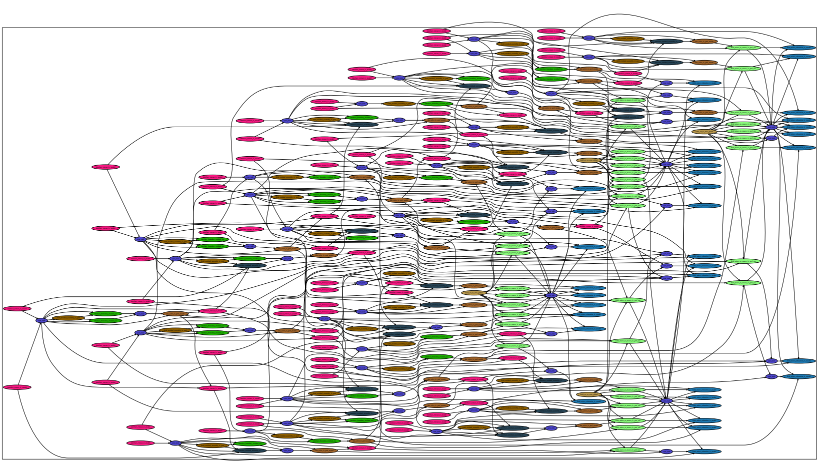
- a single 2D macrocell
- a single time step of the first order scheme (T + R)



Pretty simple...

What StarPU does for us

- 4 macrocells in a 2D square
- a single time step of the first order scheme (T + R)



Slightly more complicated....

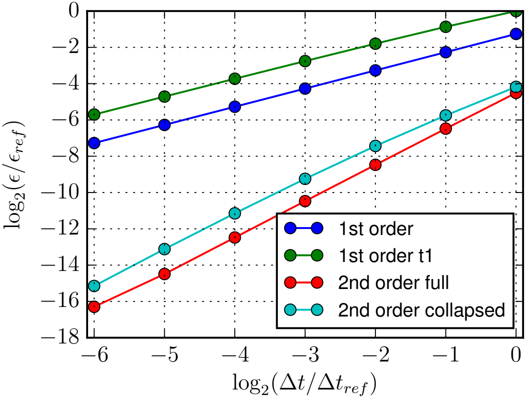
Section 4

Some numerical results

Euler gravity stationary

- D2Q9 model ,Euler stationary state in a constant gravity field $\mathbf{g} = g\mathbf{e}_y$.
- Analytical solution $\rho = \rho_0 e^{-gy/T}$

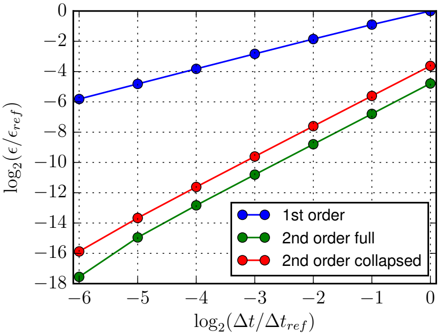
Δt Convergence of $L2$ error on macroscopic data wrt analytical solution.



First order t1 : first order splitting with fully implicit $\theta = 1$ blocks.
First order : first order splitting with $\theta = 1/2$ second order symmetric blocks.

Euler Gaussian pulse

- D2Q9 model on a square : 8x8 elements, 10x10 subcells, 3^{rd} order
- Initial condition : narrow gaussian density bump $\rho = 1 + 0.1\exp(-40*(x^2+y^2))$.
- Convergence evaluated from highly time-resolved solution.



Drifting MHD stationary vortex : $7 \times D2Q4$

Basic MHD (+divergence cleaning)

$$\begin{cases} \partial_t \rho + \nabla \cdot \rho \mathbf{u} &= 0 \\ \partial_t \rho \mathbf{u} + \nabla \cdot \left[\rho \mathbf{u}\mathbf{u} + (p + \frac{B^2}{2}) I - \mathbf{B}\mathbf{B} \right] &= 0 \\ \partial_t Q + \nabla \cdot \left[(Q + p + \frac{B^2}{2}) \mathbf{u} - (\mathbf{B} \cdot \mathbf{u}\mathbf{B}) \right] &= 0 \\ \partial_t \mathbf{B} + \nabla \cdot (\mathbf{B}\mathbf{u} - \mathbf{u}\mathbf{B} + \psi I) &= 0 \\ \partial_t \psi + \nabla \cdot (c_h^2 \mathbf{B}) &= 0 \end{cases}$$

Closure

$$p = (\gamma - 1)\rho e = (\gamma - 1)\left[Q - \rho \frac{u^2}{2} + \frac{B^2}{2} \right]$$

Simple 2D stationary solution (azimuthal symmetry) + constant drift velocity

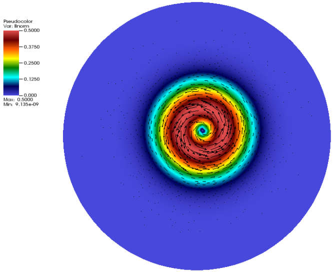
$$\begin{cases} \rho &= \rho_0 \\ \mathbf{u} &= u_0 \left[\mathbf{u}_{drift} + h(r) \mathbf{e}_\vartheta \right] \\ \mathbf{B} &= b_0 h(r) \mathbf{e}_\vartheta \\ p(r) &= p_0 + \frac{b_0^2}{2}(1 - h^2(r)) \\ b_0^2 &= \rho u_0^2 \end{cases}$$

expressed in cylindrical coordinates in the drifting frame ($\mathbf{r_0} = \mathbf{u}_{drift} t$).

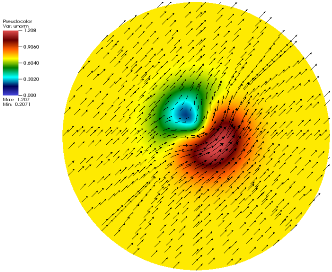
2D MHD drifting vortex

Parameters : $\rho = 1.0, p_0 = 1,\ u_0 = b_0 = 0.5,\ \mathbf{u}_{drift} = [1,1]^t,\ h(r) = exp[(1-r^2)/2]$

Magnetic field



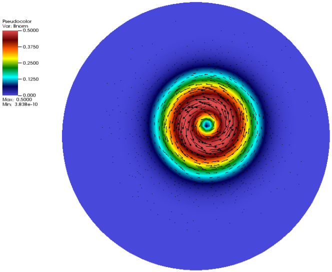
Velocity



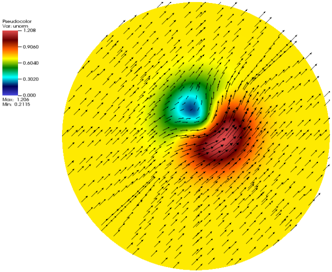
2D MHD drifting vortex

Parameters : $\rho = 1.0, p_0 = 1,\ u_0 = b_0 = 0.5,\ \mathbf{u}_{drift} = [1,1]^t,\ h(r) = exp[(1-r^2)/2]$

Magnetic field



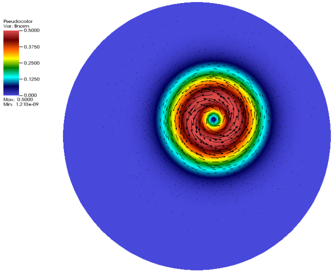
Velocity



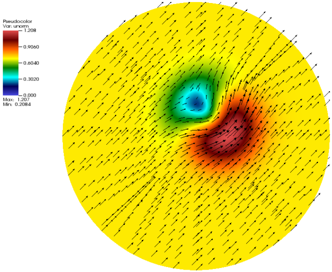
2D MHD drifting vortex

Parameters : $\rho = 1.0, p_0 = 1,\ u_0 = b_0 = 0.5,\ \mathbf{u}_{drift} = [1,1]^t,\ h(r) = exp[(1-r^2)/2]$

Magnetic field



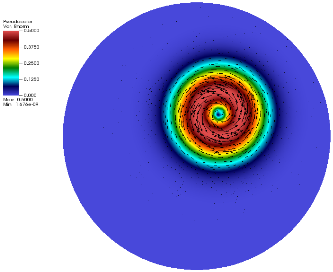
Velocity



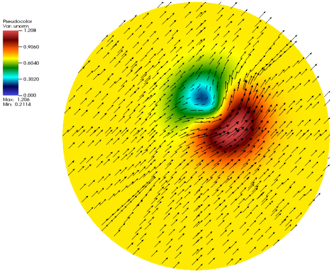
2D MHD drifting vortex

Parameters : $\rho = 1.0, p_0 = 1,\ u_0 = b_0 = 0.5,\ \mathbf{u}_{drift} = [1,1]^t,\ h(r) = exp[(1-r^2)/2]$

Magnetic field



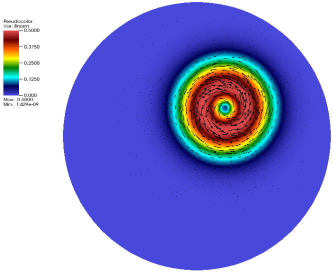
Velocity



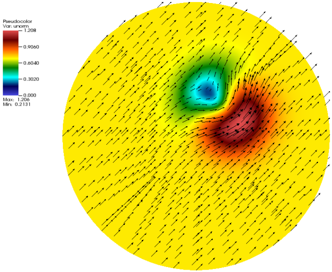
2D MHD drifting vortex

Parameters : $\rho = 1.0, p_0 = 1,\ u_0 = b_0 = 0.5,\ \mathbf{u}_{drift} = [1,1]^t,\ h(r) = exp[(1-r^2)/2]$

Magnetic field



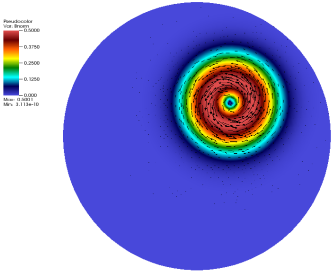
Velocity



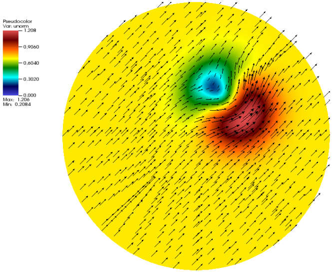
2D MHD drifting vortex

Parameters : $\rho = 1.0, p_0 = 1,\ u_0 = b_0 = 0.5,\ \mathbf{u}_{drift} = [1,1]^t,\ h(r) = exp[(1-r^2)/2]$

Magnetic field



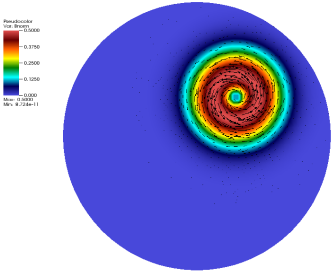
Velocity



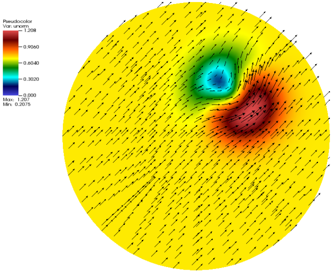
2D MHD drifting vortex

Parameters : $\rho = 1.0, p_0 = 1,\ u_0 = b_0 = 0.5,\ \mathbf{u}_{drift} = [1,1]^t,\ h(r) = exp[(1-r^2)/2]$

Magnetic field



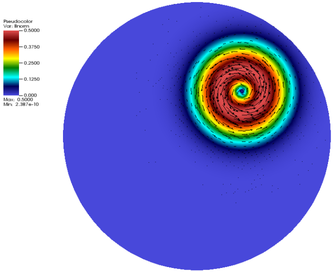
Velocity



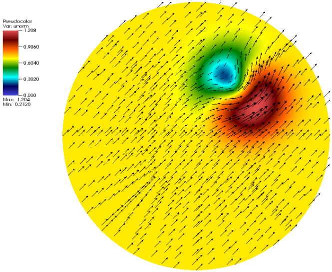
2D MHD drifting vortex

Parameters : $\rho = 1.0, p_0 = 1,\ u_0 = b_0 = 0.5,\ \mathbf{u}_{drift} = [1,1]^t,\ h(r) = exp[(1 - r^2)/2]$

Magnetic field



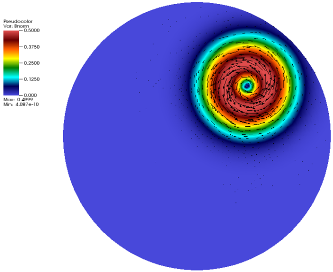
Velocity



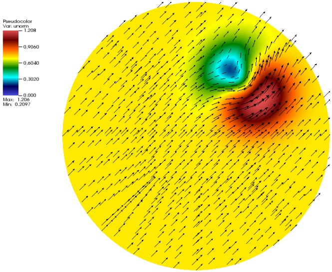
2D MHD drifting vortex

Parameters : $\rho = 1.0, p_0 = 1,\ u_0 = b_0 = 0.5,\ \mathbf{u}_{drift} = [1,1]^t,\ h(r) = exp[(1-r^2)/2]$

Magnetic field



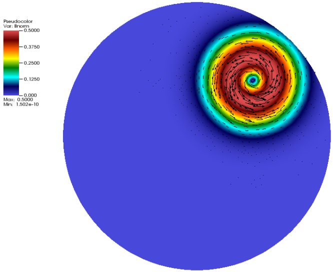
Velocity



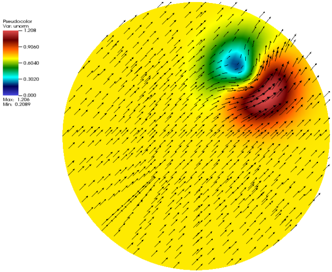
2D MHD drifting vortex

Parameters : $\rho = 1.0, p_0 = 1,\ u_0 = b_0 = 0.5,\ \mathbf{u}_{drift} = [1,1]^t,\ h(r) = exp[(1 - r^2)/2]$

Magnetic field



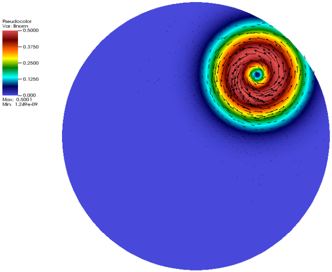
Velocity



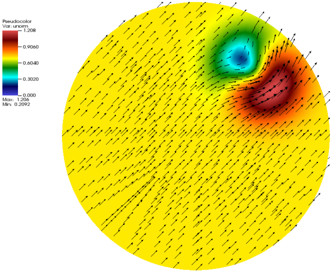
2D MHD drifting vortex

Parameters : $\rho = 1.0, p_0 = 1,\ u_0 = b_0 = 0.5,\ \mathbf{u}_{drift} = [1,1]^t,\ h(r) = exp[(1-r^2)/2]$

Magnetic field



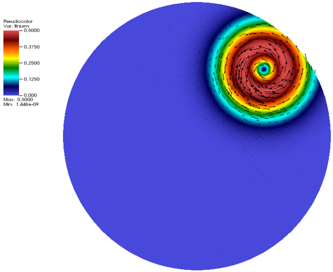
Velocity



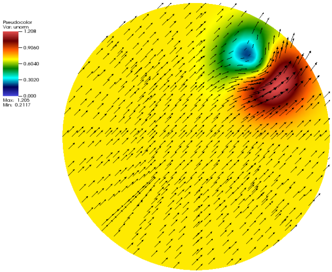
2D MHD drifting vortex

Parameters : $\rho = 1.0, p_0 = 1,\ u_0 = b_0 = 0.5,\ \mathbf{u}_{drift} = [1,1]^t,\ h(r) = exp[(1 - r^2)/2]$

Magnetic field



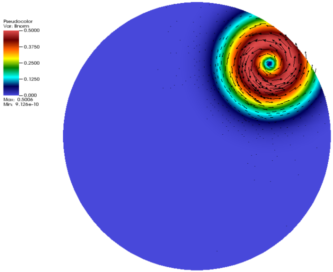
Velocity



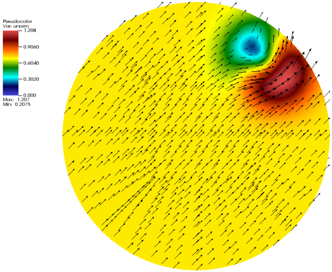
2D MHD drifting vortex

Parameters : $\rho = 1.0, p_0 = 1,\ u_0 = b_0 = 0.5,\ \mathbf{u}_{drift} = [1,1]^t,\ h(r) = exp[(1-r^2)/2]$

Magnetic field



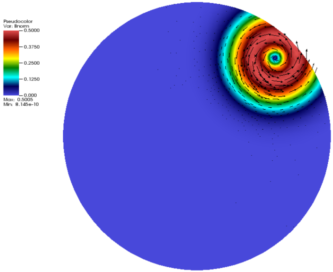
Velocity



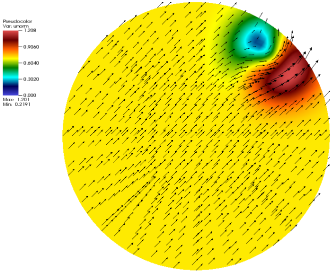
2D MHD drifting vortex

Parameters : $\rho = 1.0, p_0 = 1,\ u_0 = b_0 = 0.5,\ \mathbf{u}_{drift} = [1,1]^t,\ h(r) = exp[(1 - r^2)/2]$

Magnetic field



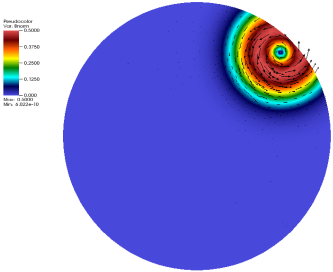
Velocity



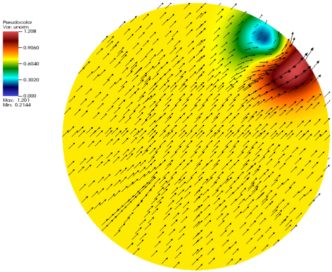
2D MHD drifting vortex

Parameters : $\rho = 1.0, p_0 = 1,\ u_0 = b_0 = 0.5,\ \mathbf{u}_{drift} = [1,1]^t,\ h(r) = exp[(1 - r^2)/2]$

Magnetic field



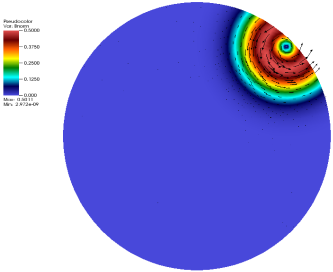
Velocity



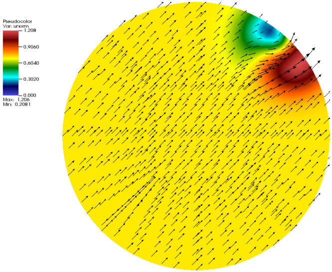
2D MHD drifting vortex

Parameters : $\rho = 1.0, p_0 = 1,\ u_0 = b_0 = 0.5,\ \mathbf{u}_{drift} = [1,1]^t,\ h(r) = exp[(1-r^2)/2]$

Magnetic field



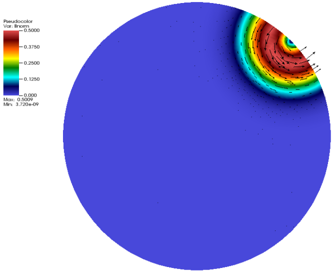
Velocity



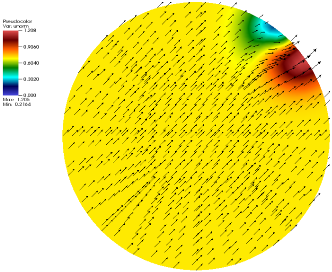
2D MHD drifting vortex

Parameters : $\rho = 1.0, p_0 = 1,\ u_0 = b_0 = 0.5,\ \mathbf{u}_{drift} = [1,1]^t,\ h(r) = exp[(1-r^2)/2]$

Magnetic field



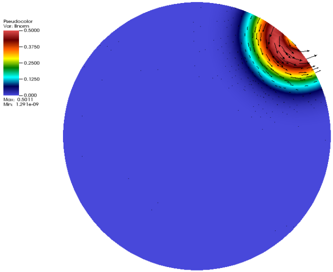
Velocity



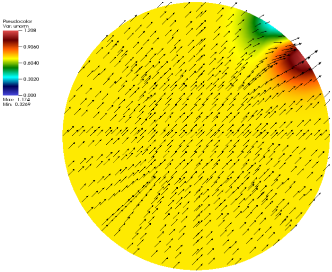
2D MHD drifting vortex

Parameters : $\rho = 1.0, p_0 = 1,\ u_0 = b_0 = 0.5,\ \mathbf{u}_{drift} = [1,1]^t,\ h(r) = exp[(1 - r^2)/2]$

Magnetic field

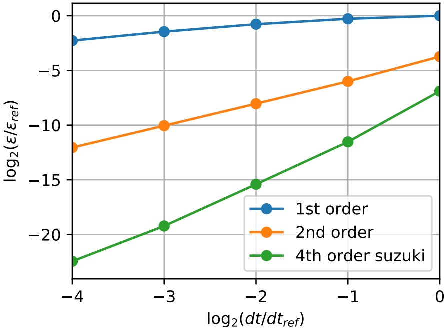


Velocity



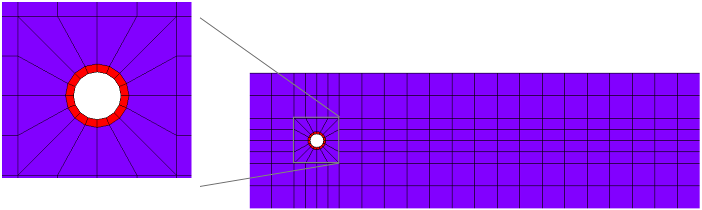
MHD drifiting vortex : time order check

- 2D disk mesh : 20 macro-elements
- refinement 8 , degree 5 : 2304 quadrature points per element
- six scalar fields (no div B cleaning)
- dt ranges from $dt_{ref} = 0.2$ to $dt = 0.006250$
- $hmin \approx 0.02$, discrete velocities norm $\|v_i\| = 4$;



$2D$ flow around a cylinder

- D2Q9 model for Isothermal Euler
- mesh is adapted to the geometry of the obstacle
- no-slip ($u = 0$) condition on the obstacle imposed using a penalization method in a small volume (red ring)
  - relaxation of each f_i towards $0.5(f_i + f_{\bar{i}})$ where $v_{\bar{i}} = -v_i$.
  - with CN scheme and $(\tau = 0) \rightarrow$ "bounce-back" operator : simply swap f_i values between opposite velocities.
- imposed state at boundaries with constant low Mach flow.



- imposed velocity field at inlet $u \approx 0.07c$.
- finite $\tau = 0.0001$.

Velocity field norm

- imposed velocity field at inlet $u \approx 0.07c$.
- finite $\tau = 0.0001$.

Velocity field norm

- imposed velocity field at inlet $u \approx 0.07c$.
- finite $\tau = 0.0001$.

Velocity field norm

- imposed velocity field at inlet $u \approx 0.07c$.
- finite $\tau = 0.0001$.

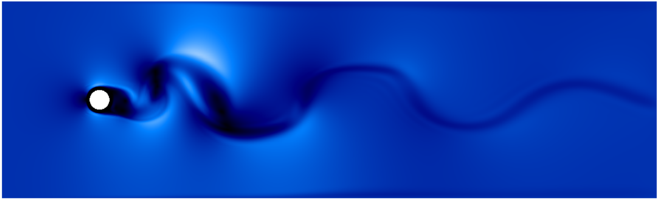
Velocity field norm

- imposed velocity field at inlet $u \approx 0.07c$.
- finite $\tau = 0.0001$.

Velocity field norm

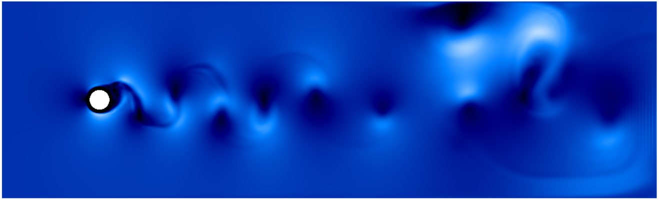
- imposed velocity field at inlet $u \approx 0.07c$.
- finite $\tau = 0.0001$.

Velocity field norm



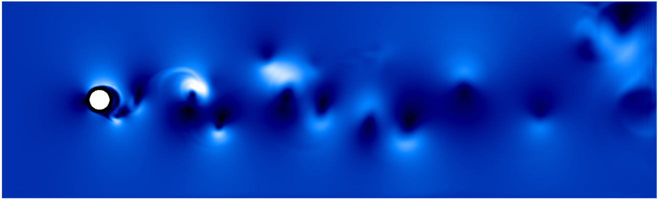
- imposed velocity field at inlet $u \approx 0.07c$.
- finite $\tau = 0.0001$.

Velocity field norm



- imposed velocity field at inlet $u \approx 0.07c$.
- finite $\tau = 0.0001$.

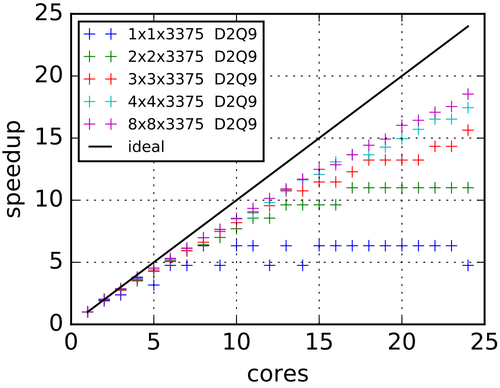
Velocity field norm



D2Q9 multithread performance

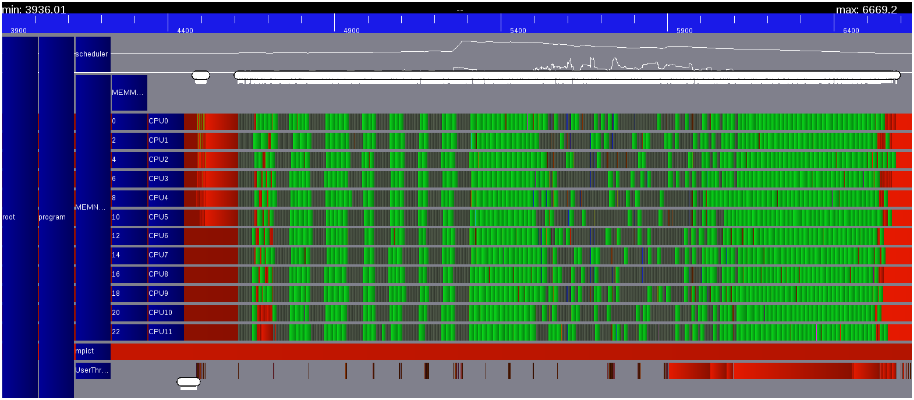
Full D2Q9 scheme on square grids. Constant dof number per macrocell. Number N of macrocells N from 1 to $64 = 8x8$.

- for 1 macrocell : saturation at $n_{core} = n_v$. This is expected.
- efficiency grows with N due to topological parallelism.



Gantt diagram

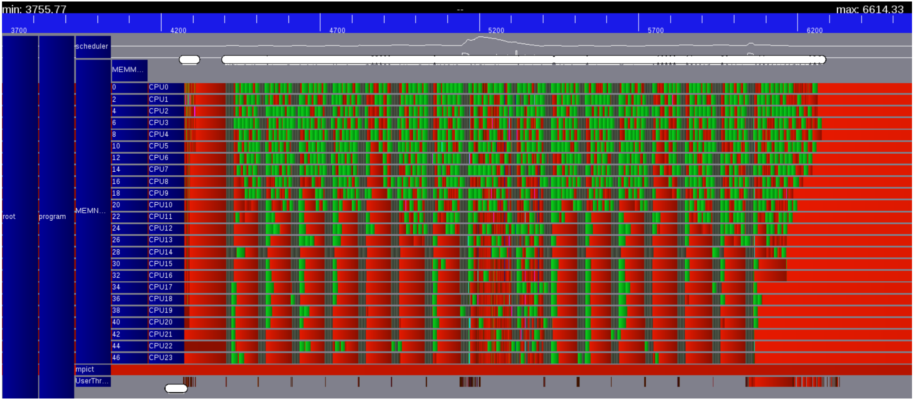
D2Q9 model - 12 threads



Low overhead - low sleep/idle time (in red)

Gantt diagram

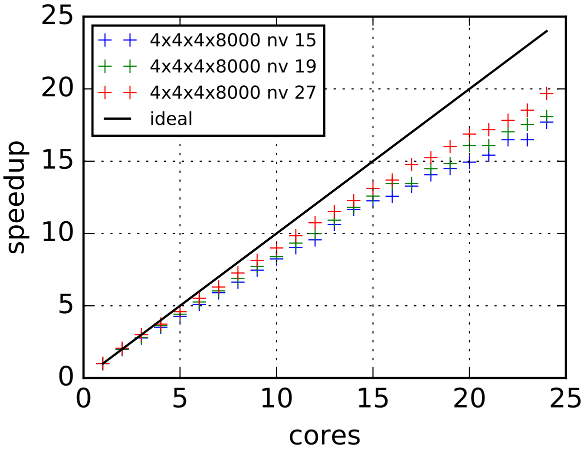
8x8 square - D2Q9 model - 24 threads



No enough work to do !

D3Q* multithread performance

D3Q15,D3Q19,D3Q27 models on a cube with 4x4x4 elements and 8000 dof per elements with eager scheduler.

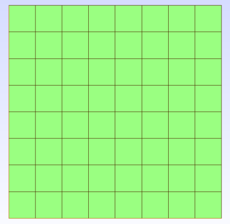


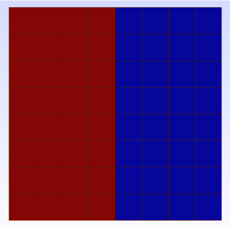
Preliminary MPI scalings : D2Q9

Irma Atlas : 4 nodes - 24 CPU per node
D2Q9 model square 8x8 elts x3200 dof

Wall time (s) for 100 iterations

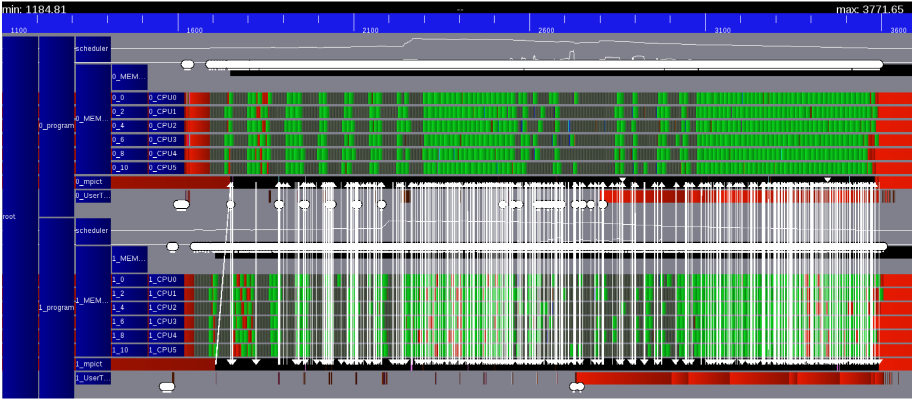
Nmpi/Nthreads	8	12	24
1	153	105	92
2	75	52	95





Gantt diagram

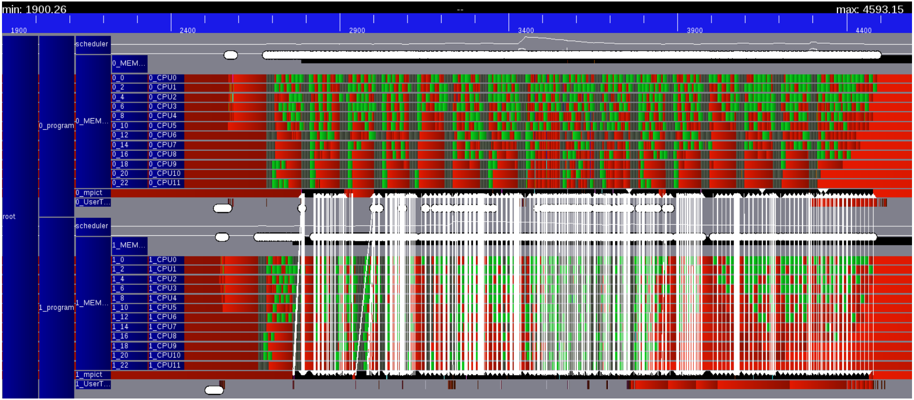
D2Q9 model - 2x6 threads



Low overhead/sleep time

Gantt diagram

8x8 square - D2Q9 model - 2x12 threads



Again : not enough work to do !

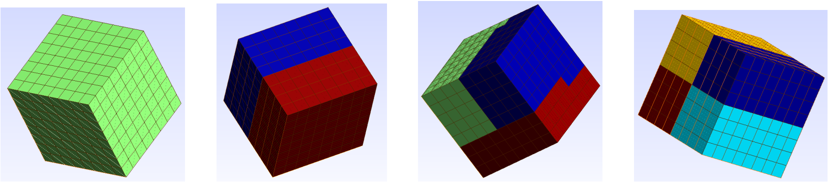
MPI scalings : D3Q15

D3Q15 - 4x4x4x8000 cubic macromesh - 50 iterations -Wall time (s)

Nthreads/Nmpi	1	2	3
10	303	151	103
15	207	105	71

D3Q15 8x8x8x4096 cubic macromesh 50 iterations - Wall time (s)

Nthreads/Nmpi	1	2	3	4
14	1290	752	409	312

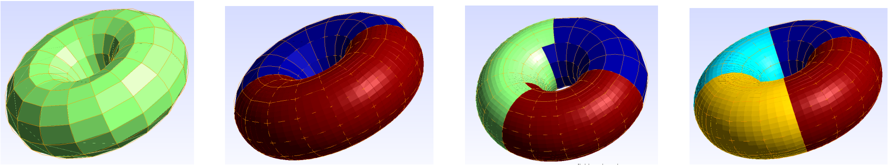


MPI Scaling : D3Q15 in a torus

Toroidal mesh : 720 macroelements x 3335 dof
2064 interfaces - 192 boundary faces

Wall time in sec for 100 interations.

Nthreads/Nmpi	1	2	3	4
14	6862	2772	1491	1014



Section 5

Conclusions

Advantages

- resolution of a set of simple independent problems (divide and conquer)
- major cost stems from transport solver DG OK, other solvers must be tried (SL , particle, ...)
- (very) low storage implicit solvers.
- high order in time.

Challenging points on the theory side / Ongoing Work on model generation

- stability/consistency/error control conditions are complex , particularly at high order
- adapt entropy based methods to vectorial models [Bou99] [Dub13].
- use 0 velocity nodes to perform better for stationary states.

Drawbacks (no free lunch)

- similar to relaxation model (diffusion , dispersion)
- BC implementation not trivial.

Design remark

- task based parallelization : flexibility in parallelization scheme
- the transport solver can be changed.

Bibliography I

[Bou99] F. Bouchut.
Construction of bgk models with a family of kinetic entropies for a given system of conservation laws.
Journal of Statistical Physics, 95(1):113–170, 1999.

[CD98] Shiyi Chen and Gary D Doolen.
Lattice boltzmann method for fluid flows.
Annual review of fluid mechanics, 30(1):329–364, 1998.

[Del02] Paul J Dellar.
Lattice kinetic schemes for magnetohydrodynamics.
Journal of Computational Physics, 179(1):95–126, 2002.

[Dub13] François Dubois.
Stable lattice boltzmann schemes with a dual entropy approach for monodimensional nonlinear waves.
Computers & Mathematics with Applications, 65(2):142–159, 2013.

[Gra14] Benjamin Graille.
Approximation of mono-dimensional hyperbolic systems: A lattice boltzmann scheme as a relaxation method.
Journal of Computational Physics, 266:74–88, 2014.

[KL97] William Kahan and Ren-Cang Li.
Composition constants for raising the orders of unconventional schemes for ordinary differential equations.
Mathematics of Computation of the American Mathematical Society, 66(219):1089–1099, 1997.

[MQ02] Robert I McLachlan and G Reinout W Quispel.
Splitting methods.
Acta Numerica, 11:341–434, 2002.

[Nat98] Roberto Natalini.
A discrete kinetic approximation of entropy solutions to multidimensional scalar conservation laws.
journal of differential equations, 148:292–317, 1998.

[Suz90] Masuo Suzuki.
Fractal decomposition of exponential operators with applications to many-body theories and monte carlo simulations.
Physics Letters A, 146(6):319–323, 1990.