Coupled fluid and kinetic plasma codes on GPUs

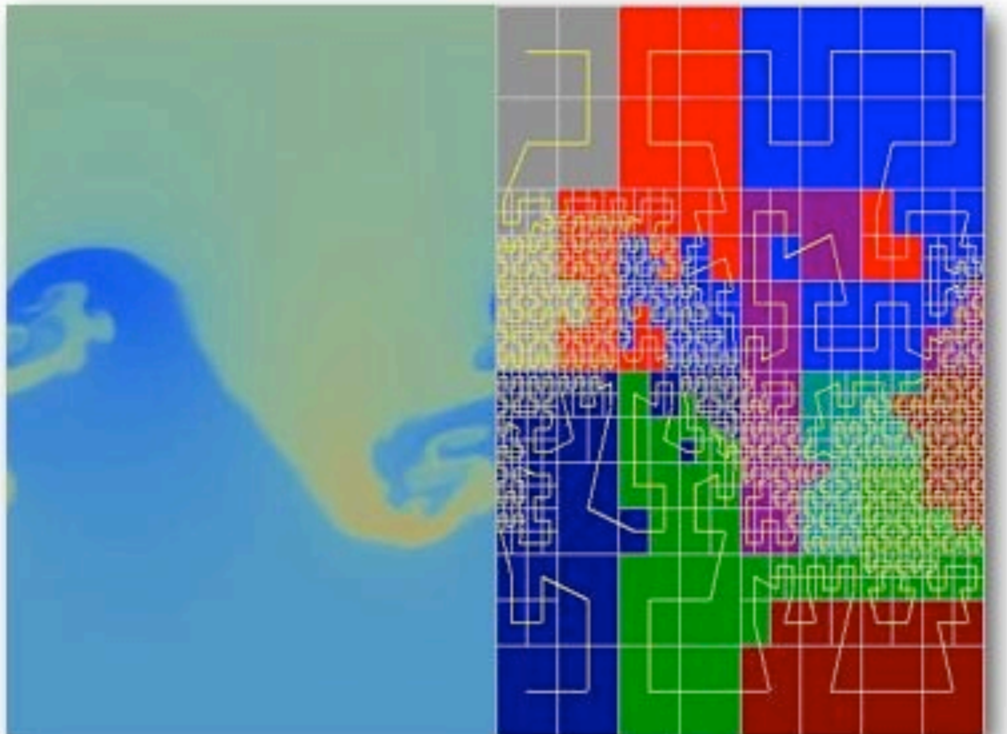
Martin Rieke, Thomas Trost, Rainer Grauer

Institute for Theoretical Physics I, Ruhr-University Bochum

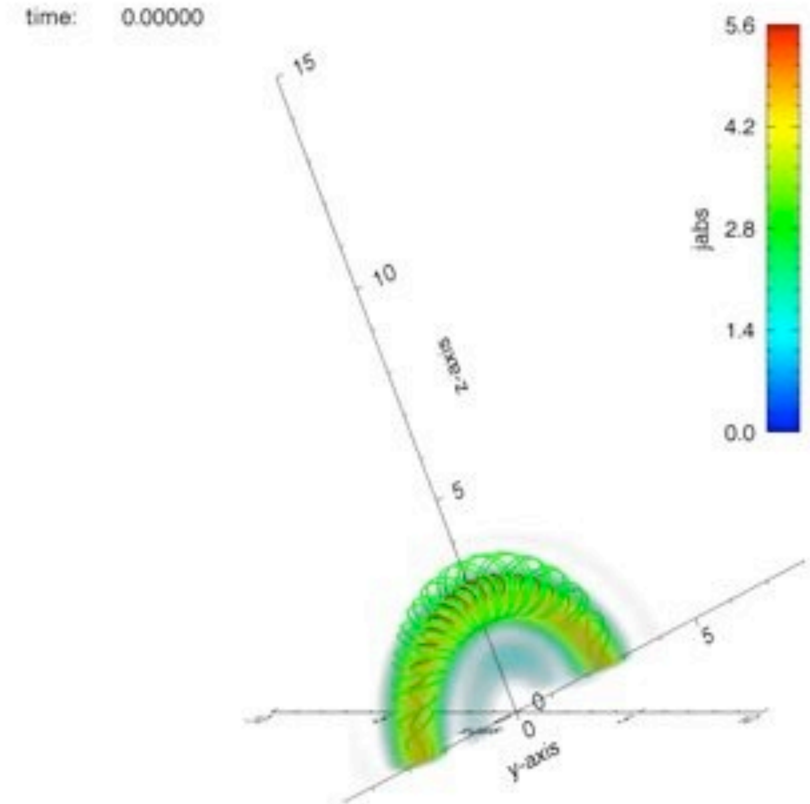
Les Houches 2013

What are we doing ?

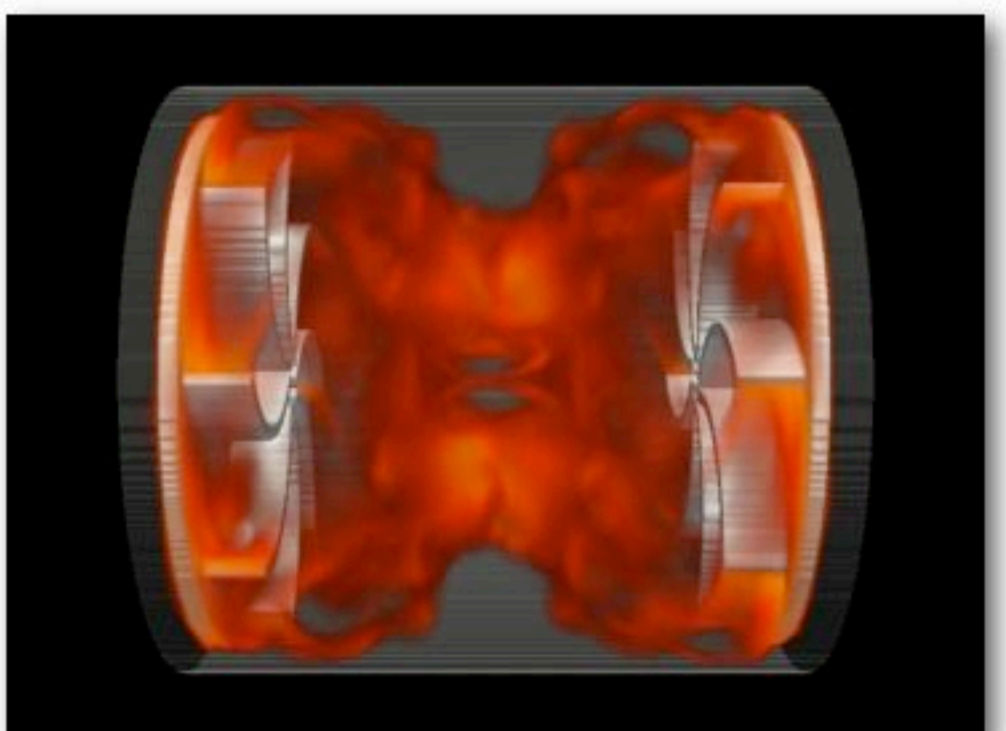
Numerical Methods: *racoon*



FlareLab: Soltwisch next week



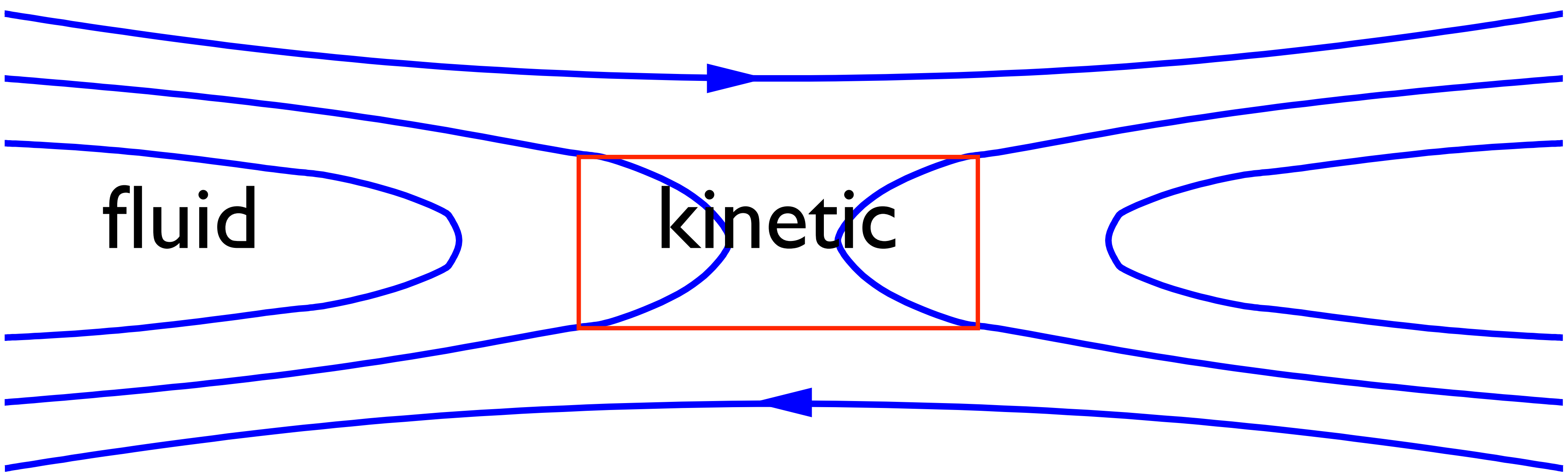
Dynamo simulations using Penalty



Turbulence: LaTu, cudaHYPE

Motivation

- fluid description
  MHD, Hall-MHD, 5- or 10 moment MHD
- kinetic description
  PIC, Vlasov
- Coupling fluid and kinetic simulations



Hyperbolic equations:

- weak solutions
- Riemann problem, Riemann solver
- CWENO
- div B = 0: divergence cleaning,   FCT
- 5- and 10-moment equations

PIC, Vlasov:

- PFC
- Boris push + back-substitution
- Darwin approximation
- Explicit Maxwell solver
- CUDA

Coupling:

- kinetic -> fluid
- fluid -> kinetic
- Examples

compressible MHD

in conservation form:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0$$

$$\frac{\partial \rho \mathbf{v}}{\partial t} + \nabla \cdot \left(\mathbf{v} \rho \mathbf{v} + \mathbf{I}(p + \frac{\mathbf{B}^2}{2}) - \mathbf{BB} \right) = 0$$

$$\frac{\partial e}{\partial t} + \nabla \cdot \left(\mathbf{v}(e + p + \frac{\mathbf{B}^2}{2}) - \mathbf{B}(\mathbf{v} \cdot \mathbf{B}) \right)$$

$$\frac{\partial B}{\partial t} + \nabla \cdot (\mathbf{vB} - \mathbf{Bv}) = 0$$

$$p = (\gamma - 1) \left(e - \frac{1}{2} \rho \mathbf{v}^2 - \frac{1}{2} \mathbf{B}^2 \right)$$

see also Rossmanith, Johnson; Shumlak, Hakim, ...

5 moments

$$\begin{aligned}
\partial_t \rho_s &= -\nabla \cdot \mathbf{u}_s \\
\partial_t \mathbf{u}_s &= -\nabla \cdot \Big(\rho_s^{-1} \mathbf{u}_s \otimes \mathbf{u}_s\Big) - \frac{1}{3}\nabla\Big(2\mathcal{E}_s - \rho_s^{-1}\mathbf{u}_s^2\Big) + \frac{q_s}{m_s}\Big(\rho_s \mathbf{E} + \mathbf{u}_s \times \mathbf{B}\Big) \\
\partial_t \mathcal{E}_s &= -\frac{1}{3}\nabla \cdot \Big(\rho_s^{-2}\big(5\rho_s\mathcal{E}_s - \mathbf{u}_s^2\big)\mathbf{u}_s\Big) + \frac{q_s}{m_s}\mathbf{u}_s \cdot \mathbf{E}
\end{aligned}$$

10 moments

$$\begin{aligned}
\partial_t \rho_s &= -\nabla \cdot (\mathbf{u}_s) \\
\partial_t \mathbf{u}_s &= -\nabla \cdot \mathsf{E}_s + \frac{q_s}{m_s}(\rho_s \mathbf{E} + \mathbf{u}_s \times \mathbf{B}) \\
\partial_t \mathsf{E}_s &= -\nabla \cdot \big[\mathbf{u} \vee (3\rho_s^{-1}\mathsf{E} - 2\rho_s^{-2}(\mathbf{u} \otimes \mathbf{u}))\big] + \frac{q_s}{m_s}\Big(2\mathbf{u}_s \vee \mathbf{E} + \mathsf{E}_s \times \mathbf{B} + (\mathsf{E}_s \times \mathbf{B})^{\mathrm{T}}\Big) + \mathsf{R}_{\mathrm{iso}}
\end{aligned}$$

with

$$\mathsf{R}_{\mathrm{iso}} = \frac{1}{\tau_s}\left(\frac{1}{3}(\mathrm{tr}\,\mathsf{P}_s)\mathbb{1} - \mathsf{P}\right) \quad \text{with} \quad \tau_s = \tau_0\sqrt{\frac{\det \mathsf{P}}{\rho_s^5}}$$

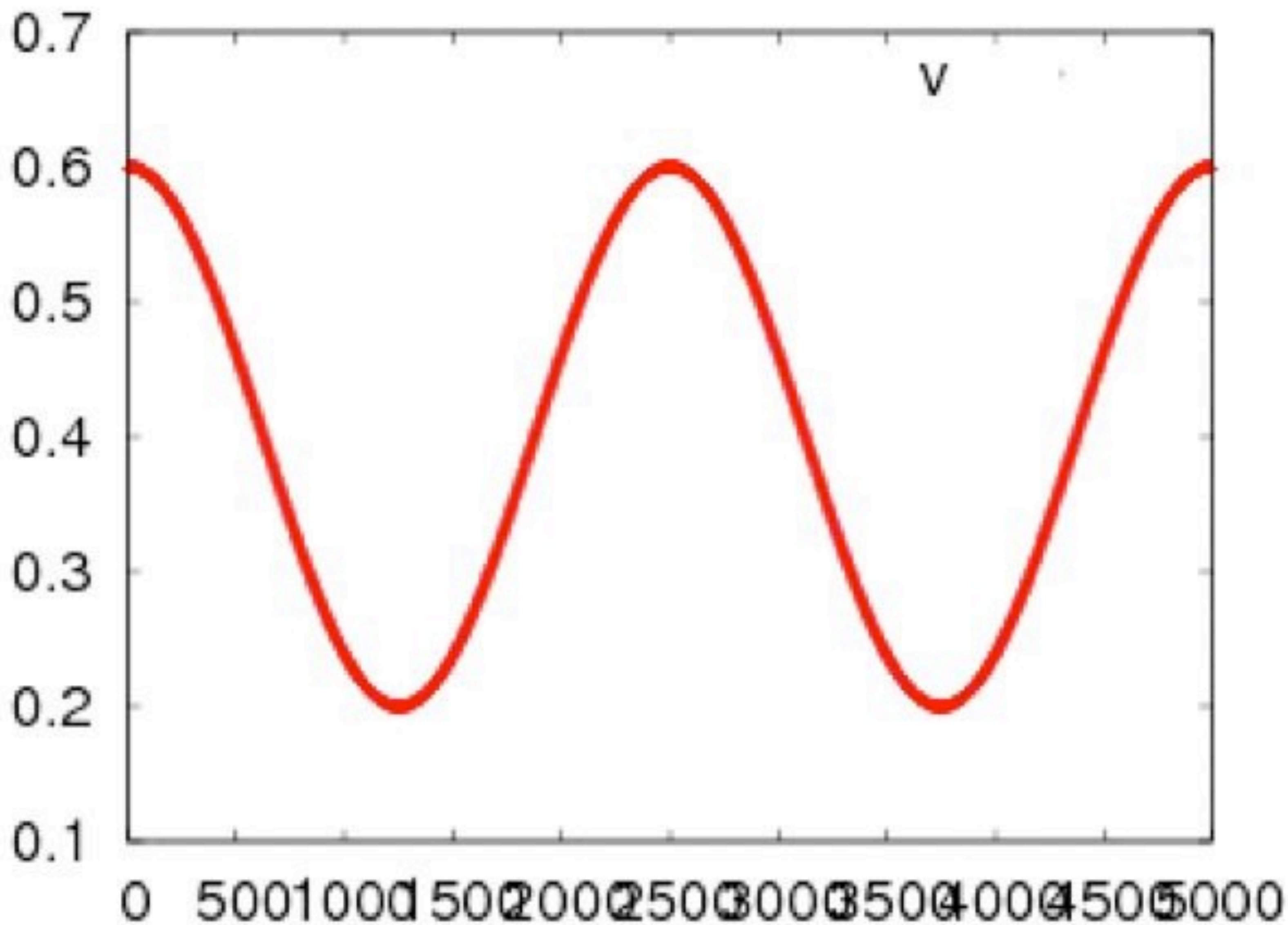
Faraday's and Ampère's law:

$$\begin{aligned}
\partial_t \mathbf{B} &= -\nabla \times \mathbf{E} \\
\partial_t \mathbf{E} &= c^2\left(\nabla \times \mathbf{B} - \mu_0 \sum_s \frac{q_s}{m_s}\mathbf{u_s}\right).
\end{aligned}$$

Model problem: Burgers equation

$$\partial_t v + \partial_x \frac{1}{2} v^2 = 0$$

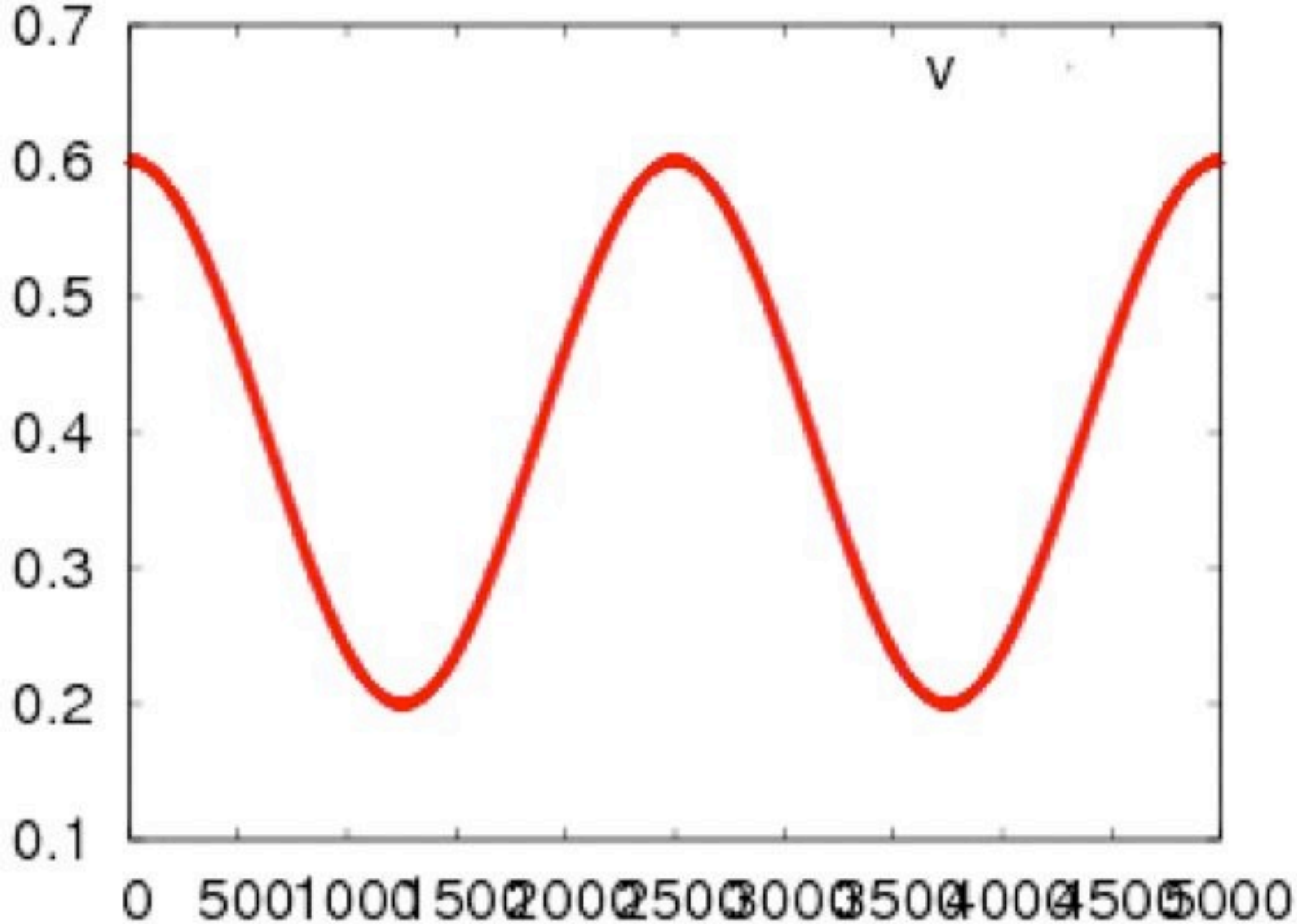
same type of differential equation



Simulation
with
finite differences

↓

need the concept
of *weak solutions*



shocks

$$\partial_t v + \partial_x f(v) = 0$$

Velocity $\tilde{v}$ of shock can be determined analytically
Rankine-Hugoniot condition:

$$(v_l - v_r)\tilde{v} = f(v_l) - f(v_r)$$

Rankine-Hugoniot condition for Burgers shock

$$\partial_t v + \partial_x \frac{1}{2} v^2 = 0 \qquad \Rightarrow \qquad \tilde{v} = \frac{1}{2}(v_l - v_r)$$

Now consider

$$\partial_t v + \partial_x \frac{1}{2} v^2 = 0 \qquad \Big| \cdot v^2$$

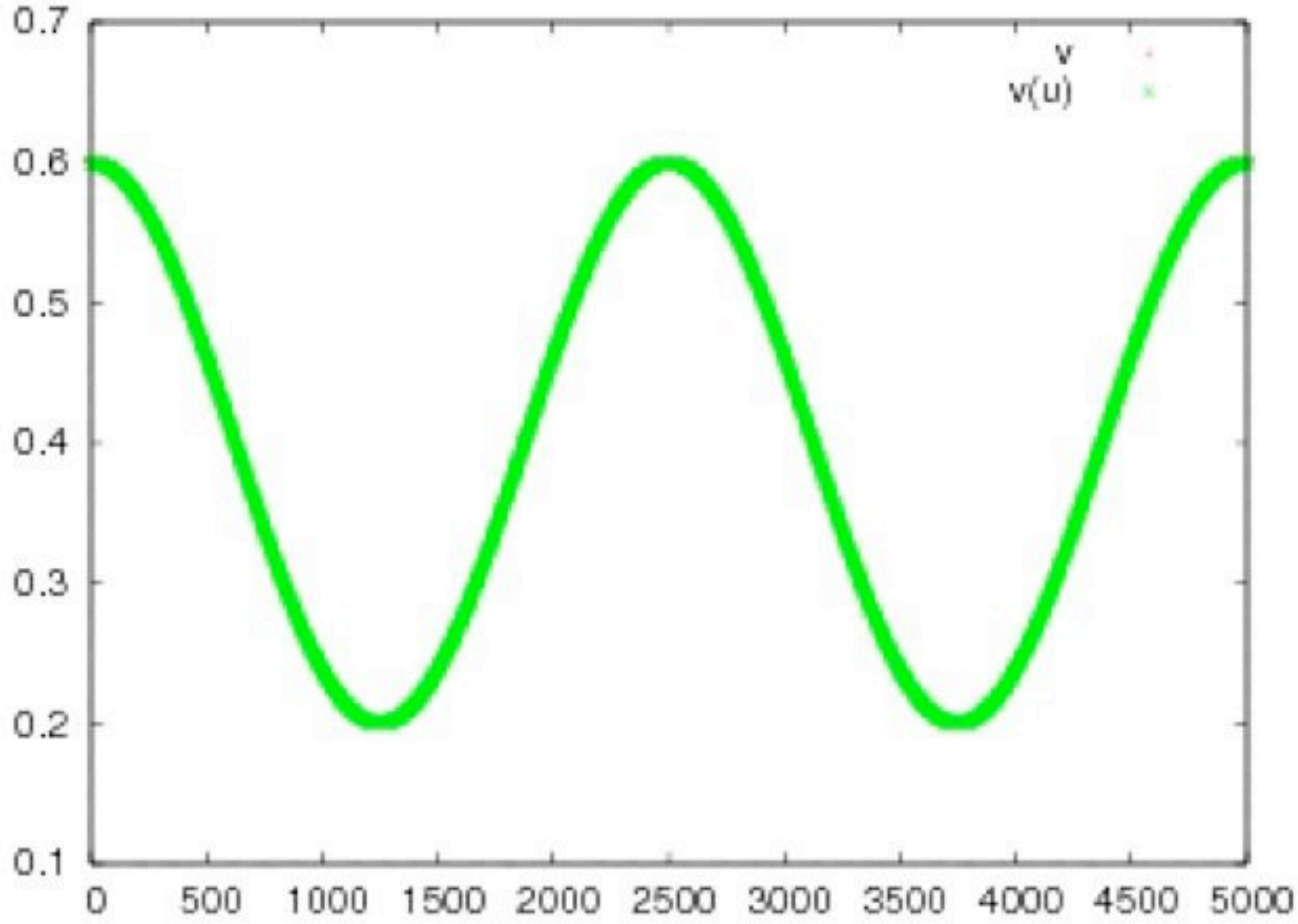
$$\partial_t \frac{1}{3} v^3 + \frac{1}{4} \partial_x v^4 = 0 \qquad \Big| u = v^3 \qquad \Rightarrow \qquad \tilde{v} = \frac{3}{4} \frac{v_l^4 - v_r^4}{v_l^3 - v_r^3}$$

$$\partial_t u + \partial_x \frac{3}{4} u^{4/3} = 0$$

Rankine-Hugoniot conditions for Burgers

$$\Rightarrow \tilde{v} = \frac{1}{2}(v_l - v_r)$$

$$\Rightarrow \tilde{v} = \frac{3}{4}\frac{v_l^4 - v_r^4}{v_l^3 - v_r^3}$$



Dissipation due to lack of smoothness

consider smooth solution:

$$\partial_t v + \frac{1}{2}\partial_x v^2 = 0 \qquad | \cdot v$$

$$\frac{1}{2}\partial_t v^2 + \frac{1}{3}\partial_x v^3 = 0 \qquad | \int \ , \quad E = \frac{1}{2}\int v^2 dx \ , \quad \text{periodic BC}$$

$$\partial_t E + \int \frac{1}{3}\partial_x v^3 dx = \partial_t E = 0 \qquad \text{energy conservation}$$

everything is fine !!!

consider shock at x_0

$$\partial_t E + \int_0^{x_0-\epsilon} \frac{1}{3}\partial_x v^3 dx + \int_{x_0+\epsilon}^{L} \frac{1}{3}\partial_x v^3 dx = 0$$

$$\partial_t E + \frac{1}{3}v^3(x_0-\epsilon) - \frac{1}{3}v^3(x_0+\epsilon) = 0$$

$$\partial_t E < 0$$

dissipation anomaly

incompressible Navier-Stokes:

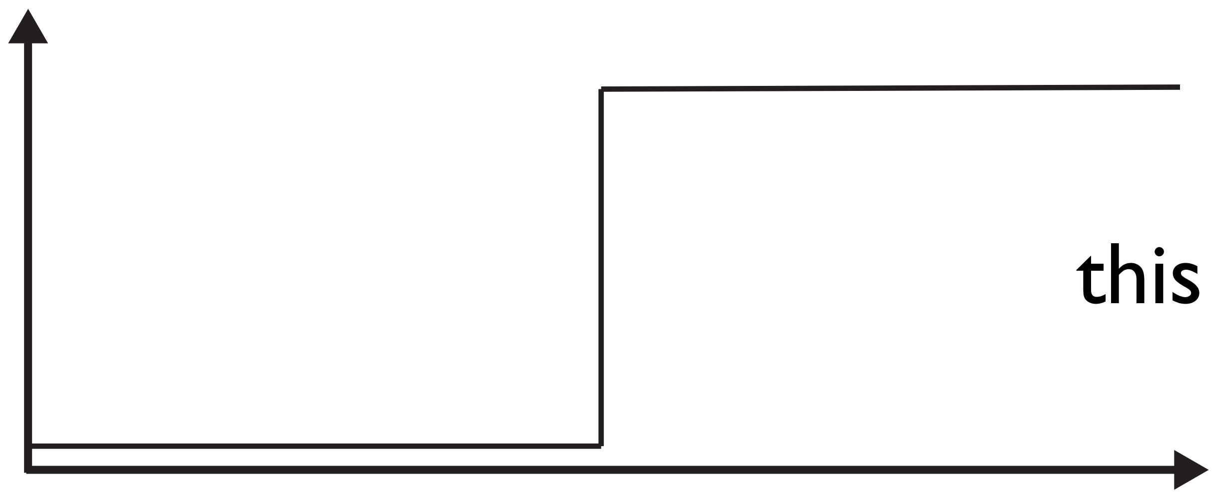
Onsager (1949): Lipschitz condition

$$|\mathbf{v}(\mathbf{r}+\mathbf{l}) - \mathbf{v}(\mathbf{r})| < \text{const } l^n \quad , \quad n > \frac{1}{3} \quad \Longrightarrow \text{energy conservation}$$

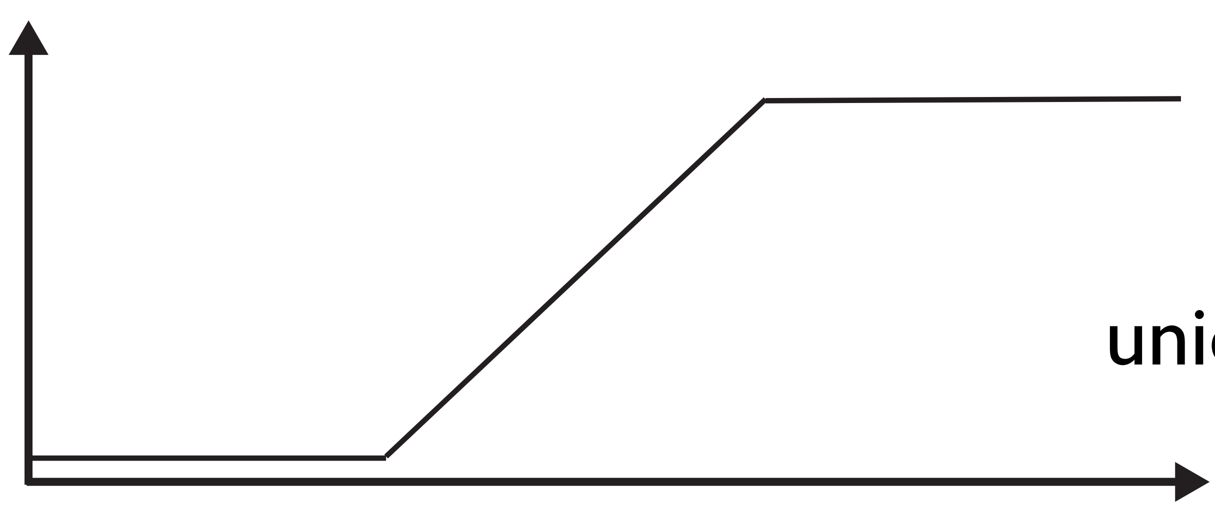
see review by Eyink and Sreenivasan (2006)

Entropy solution:

- weak solutions are not unique
- uniqueness enforced by entropy condition



this is a weak solution to Burgers



rarefaction wave:
unique weak solution to Burgers

compressible MHD

Riemann solvers

examples: Godunov, PPM, HLL(*), wave-propagation

- very good resolution of shocks
- very bad in smooth regions

ENO-schemes

- shock resolution not as good as from Riemann solvers,
- much better resolution of waves in smooth regions
- very easy!!!

We use now for more than 10 years CWENO-type schemes.

How do they work?

Semi-discrete central schemes, CWENO

Nessyahu and Tadmor (1990)
Kurganov and Levy (2000)

Why central schemes?

- no (aproximate) Riemann solver necessary
- dimension by dimension approach makes sence
- high order
- monotone, WENO, TVD depends on the reconstruction
- **easy** for complex problems

starting point

Lax-Friedrich

$$u_i^{n+1} = \frac{1}{2}(u_{i+1}^n + u_{i-1}^n) - \frac{dt}{2dx}(f(u_{i+1}^n) - f(u_{i-1}^n)$$

$\Longrightarrow$ dissipation $= \dfrac{(\Delta x)^2}{2\Delta t}$

useless, since

i) high dissipation — need high order

ii) dissipation depends on timestep — need semi-discrete scheme

Details

First, consider a 1D conservation law:

$$\partial_t u(x,t) + \partial_x f(u(x,t)) = 0$$

Fully discrete third order scheme

cell averages

$$\bar{u}_j^n \equiv \frac{1}{\Delta x} \int_{x_{j-1/2}}^{x_{j+1/2}} u(x, t^n) dx,$$

$\Longrightarrow$

$$\bar{u}_j^{n+1} = \bar{u}_j^n - \frac{1}{\Delta x} \int_{t^n}^{t^{n+1}} \left[f(u(x_{j+1/2}, \tau)) - f(u(x_{j-1/2}, \tau)) \right] d\tau$$

piecewise polynomial reconstruction from the cell averages

$$u(x, t^n) \approx \tilde{u}(x, t^n) = \sum_j P_j(x) \chi_{[x_{j-1/2}, x_{j+1/2}]}.$$

third order scheme: non-oscillatory parabolic reconstruction approximated function $\tilde{u}(x, t^n)$ discontinous at the cell boundaries $x_{j+1/2}$.

different limits $u^{n,+}_{j+1/2}$, $u^{n,-}_{j+1/2}$

$$u^{n,+}_{j+1/2} = P_{j+1}(x_{j+1/2}, t^n), \;\; u^{n,-}_{j+1/2} = P_j(x_{j+1/2}, t^n),$$

upper bound for the propagation speed of the discontinuities

$$a^n_{j+1/2} = \max_{u \in (u^{n,-}_{j+1/2}, u^{n,+}_{j+1/2})} \mathsf{abs}\left(\frac{\partial f}{\partial u}(u)\right),$$

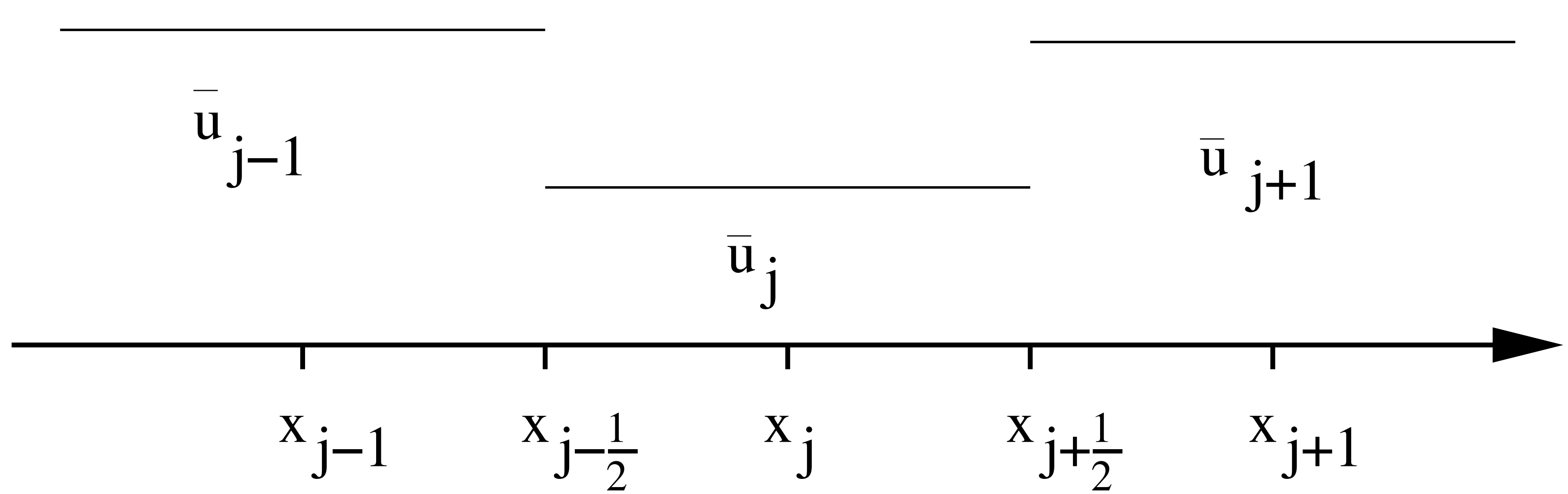
$\Longrightarrow$ non-smooth region limited to

$$x^n_{j+1/2,l} \equiv x_{j+1/2} - a^n_{j+1/2}\Delta t, \;\; x^n_{j+1/2,r} \equiv x_{j+1/2} + a^n_{j+1/2}\Delta t,$$

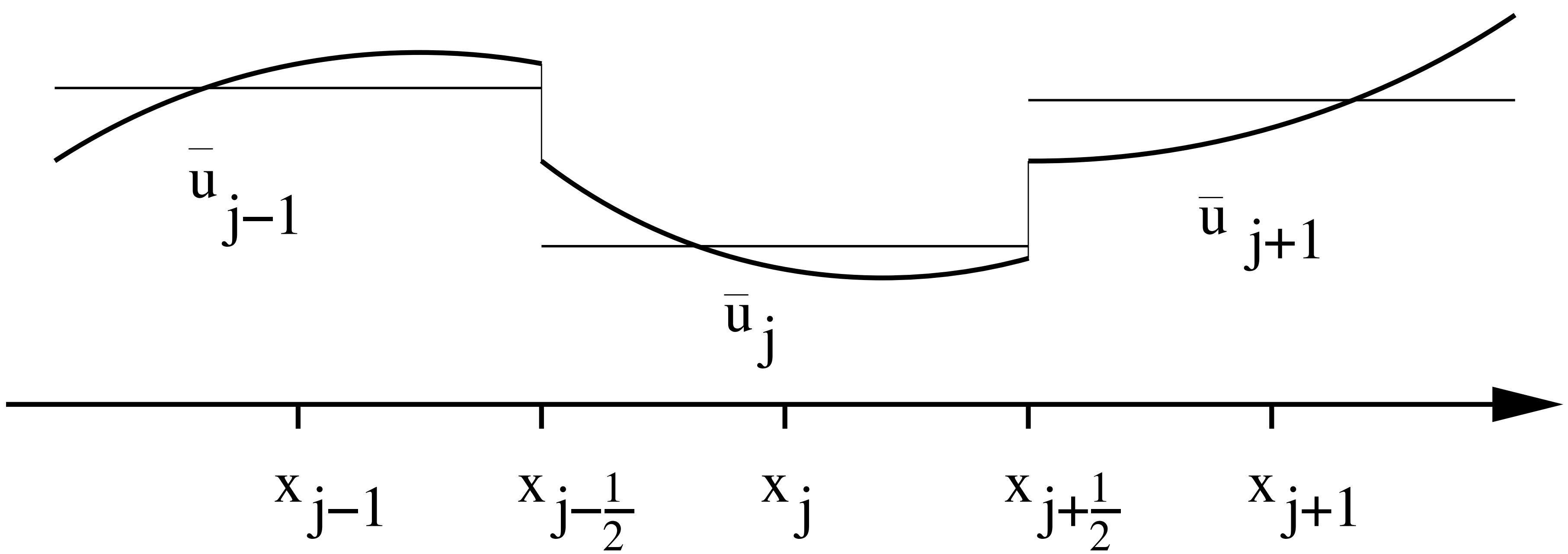
integrate smooth and non-smooth regions independently in time:
new cell averages $\bar{w}^{n+1}_j$ and $\bar{w}^{n+1}_{j+1/2}$ at time t^{n+1} on a non-uniformly spaced, twofold oversampled grid.

$\bar{u}^{n+1}_j$ follows from the $\bar{w}^{n+1}_j$ by polynomial reconstruction

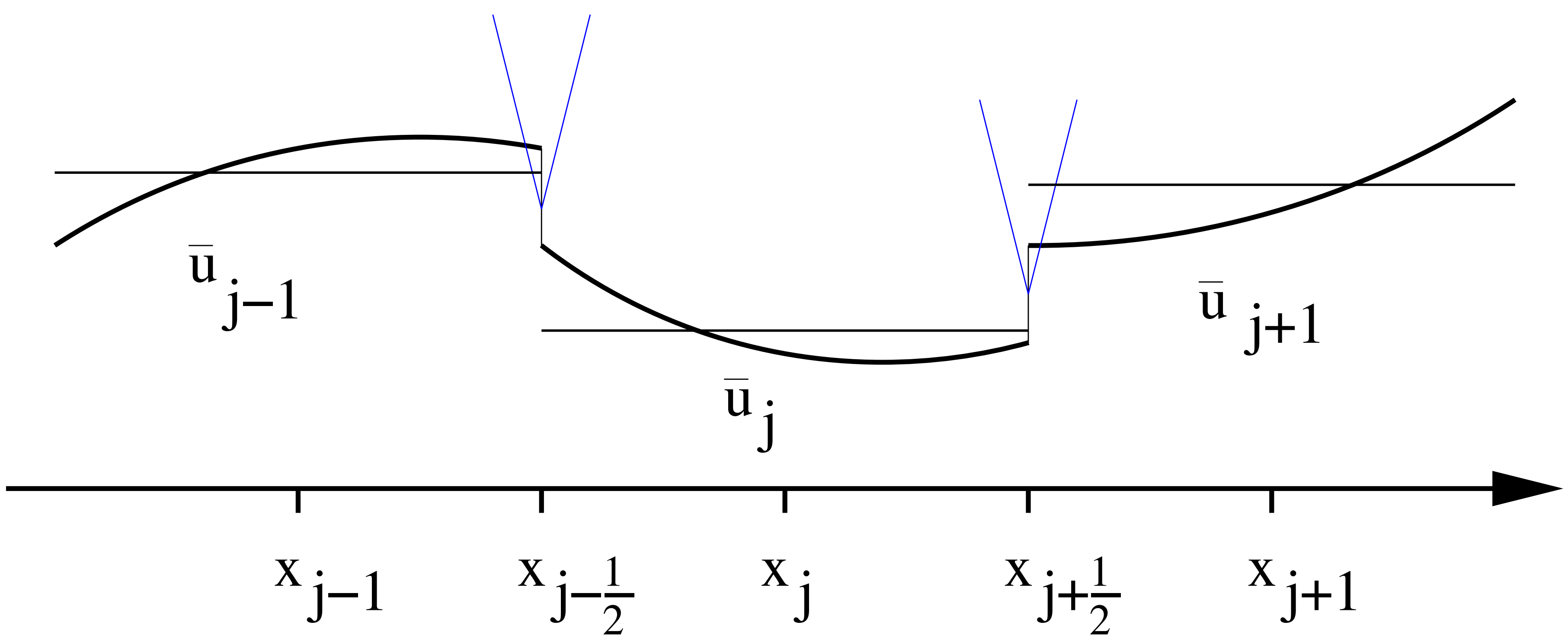
racoon: CWENO for compressible Euler and MHD



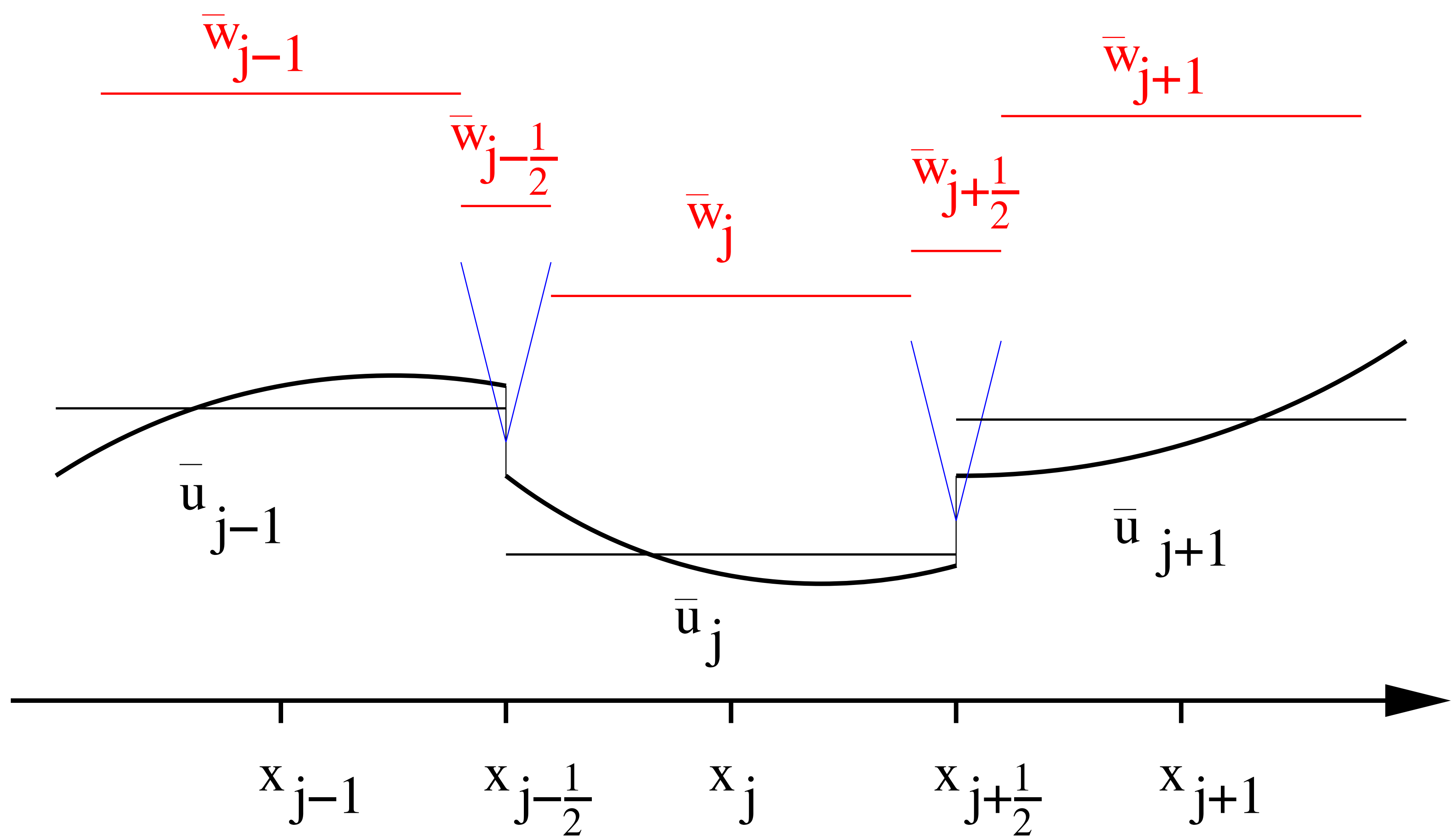
racoon: CWENO for compressible Euler and MHD



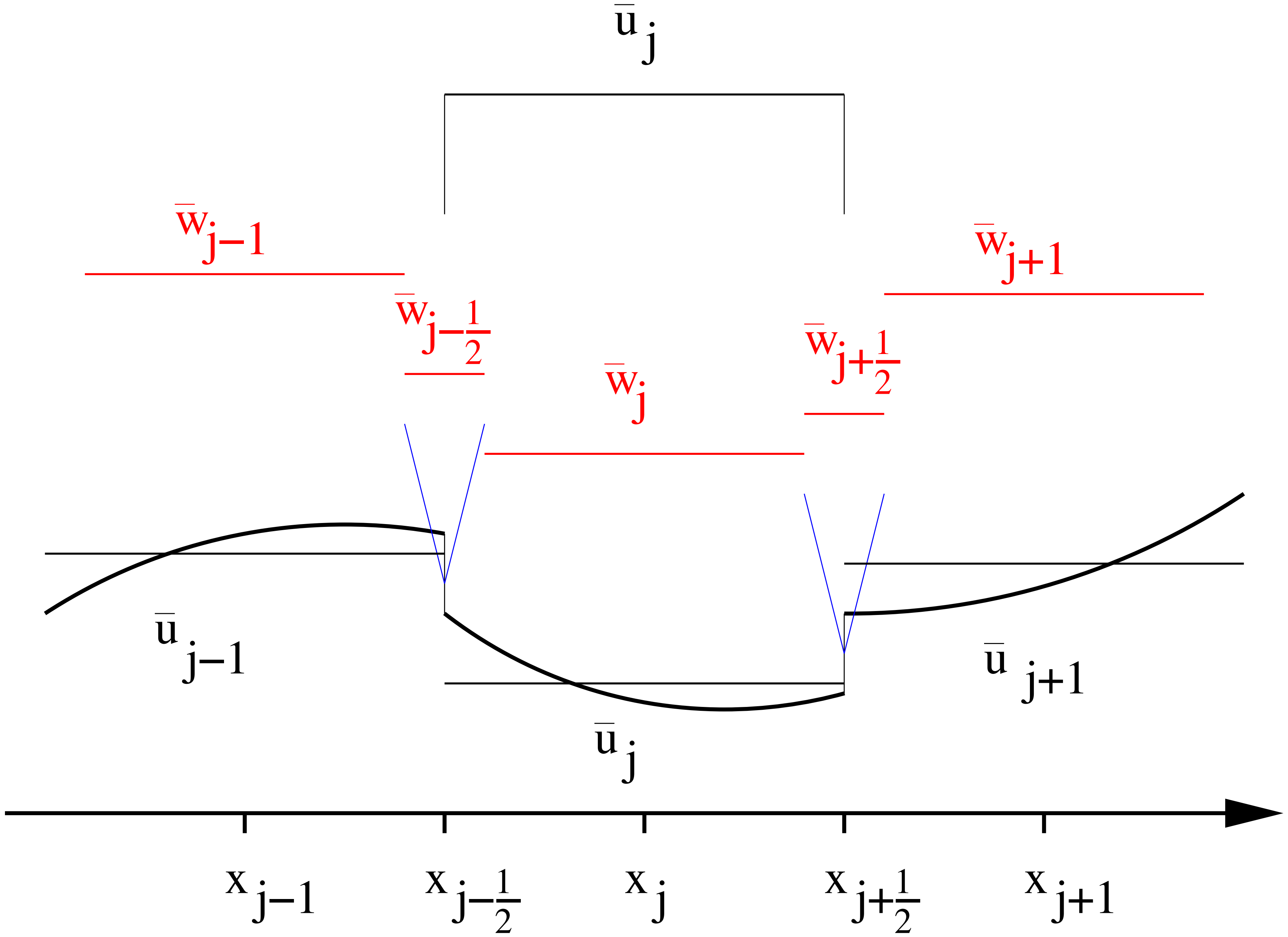
racoon: CWENO for compressible Euler and MHD



racoon: CWENO for compressible Euler and MHD



racoon: CWENO for compressible Euler and MHD



Which steps are actually performed?
Assume, we have the second order polynomial reconstruction:

$$P_j(x,t^n) = A_j + B_j(x-x_j) + \frac{1}{2}C_j(x-x_j)^2$$

A_j, B_j and C_j are determined using the given cell averages $\{u_j^n\}$. Details later.

Integrating over the non-smooth and smooth regions provides us with the non-uniform cell averages $\bar{w}_{j+1/2}^{n+1}$ and $\bar{w}_j^{n+1}$ at time t^{n+1}, respectively:

$$\begin{aligned}
\bar{w}_{j+1/2}^{n+1} &= \frac{A_j + A_{j+1}}{2} + \frac{\Delta x - a_{j+1/2}^n \Delta t}{4}(B_j - B_{j+1}) \\
&+ \left(\frac{\Delta x^2}{16} - \frac{a_{j+1/2}^n \Delta t \Delta x}{8} + \frac{(a_{j+1/2}^n \Delta t)^2}{12}\right)(C_j + C_{j+1}) \\
&- \frac{1}{2a_{j+1/2}^n \Delta t}\left\{\int_{t^n}^{t^{n+1}} \left[f(\tilde{u}(x_{j+1/2,r}^n,\tau)) - f(\tilde{u}(x_{j+1/2,l}^n,\tau))\right] d\tau\right\} \\
\bar{w}_j^{n+1} &= A_j + \frac{\Delta t}{2}(a_{j-1/2}^n - a_{j+1/2}^n)B_j \\
&+ \left[\frac{\Delta x^2}{24} - \frac{\Delta t \Delta x}{12}(a_{j-1/2}^n + a_{j+1/2}^n) + \frac{\Delta t^2}{6}\left((a_{j-1/2}^n)^2 - a_{j-1/2}^n a_{j+1/2}^n + (a_{j+1/2}^n)^2\right)\right] C_j \\
&- \frac{1}{\Delta x - \Delta t(a_{j-1/2}^n + a_{j+1/2}^n)}\left\{\int_{t^n}^{t^{n+1}} \left[f(\tilde{u}(x_{j+1/2,l}^n,\tau)) - f(\tilde{u}(x_{j-1/2,r}^n,\tau))\right] d\tau\right\}
\end{aligned}$$

Project the non-uniform, twofold oversampled cell averages $\{\bar{w}_j^{n+1}, \bar{w}_{j+1/2}^{n+1}\}$ back onto the original uniform grid $\{\bar{u}_j^{n+1}\}$.

Constant reconstruction in smooth region is sufficient

$$\begin{aligned}
\tilde{w}^{n+1}(x) &= \sum_j \tilde{w}_{j+1/2}^{n+1}(x)\chi_{[x_{j+1/2,l}^n, x_{j+1/2,r}^n]}(x) + \\
&\quad \sum_j \tilde{w}_j^{n+1}(x)\chi_{[x_{j-1/2,r}^n x_{j+1/2,l}^n]} \\
\tilde{w}_{j+1/2}^{n+1}(x) &= \tilde{A}_{j+1/2} + \tilde{B}_{j+1/2}(x - x_{j+1/2}) + \frac{1}{2}\tilde{C}_{j+1/2}(x - x_{j+1/2})^2, \\
\tilde{w}_j^{n+1}(x) &= \bar{w}_j^{n+1},
\end{aligned}$$

The new cell averages $\bar{u}_j^{n+1}$ can then be expressed as follows:

$$\begin{aligned}
\bar{u}_j^{n+1} &= \frac{1}{\Delta x}\left[\int_{x_{j-1/2}}^{x_{j-1/2,r}^n} \tilde{w}_{j-1/2}^{n+1}(x)dx + \int_{x_{j-1/2,r}^n}^{x_{j+1/2,l}^n} \tilde{w}_j^{n+1}(x)dx + \int_{x_{j+1/2,l}^n}^{x_{j+1/2}} \tilde{w}_{j+1/2}^{n+1}(x)dx\right] \\
&= \lambda a_{j-1/2}^n \tilde{A}_{j-1/2} + \left[1 - \lambda(a_{j-1/2}^n + a_{j+1/2}^n)\right]\bar{w}_j^{n+1} + \lambda a_{j+1/2}^n \tilde{A}_{j+1/2} \\
&\quad + \frac{\lambda \Delta t}{2}\left((a_{j-1/2}^n)^2 \tilde{B}_{j-1/2} - (a_{j+1/2}^n)^2 \tilde{B}_{j+1/2}\right) \\
&\quad + \frac{\lambda (\Delta t)^2}{6}\left((a_{j-1/2}^n)^3 \tilde{C}_{j-1/2} + (a_{j+1/2}^n)^3 \tilde{C}_{j+1/2}\right),
\end{aligned}$$

where $\lambda = \Delta t / \Delta x$.

Transition to the third order semi-discrete scheme

Now consider the limit of $\Delta t \longrightarrow 0$ to derive the semi-discrete scheme:

$$\frac{d}{dt}\bar{u}_j(t) = \lim_{\Delta t \longrightarrow 0} \frac{\bar{u}_j^{n+1} - \bar{u}_j^n}{\Delta t}.$$

Result:

$$\begin{aligned}
\frac{d\bar{u}_j}{dt} \quad = \quad & -\frac{1}{2\Delta x}\left[f(u^+_{j+1/2}(t)) + f(u^-_{j+1/2}(t)) - f(u^+_{j-1/2}(t)) + f(u^-_{j-1/2}(t))\right] \\
& + \frac{a_{j+1/2}(t)}{2\Delta x}\left[u^+_{j+1/2}(t) - u^-_{j+1/2}(t)\right] \\
& + \frac{a_{j-1/2}(t)}{2\Delta x}\left[u^+_{j-1/2}(t) - u^-_{j-1/2}(t)\right]
\end{aligned}$$

Weighted ENO reconstruction

In each cell we need to reconstruct a polynomial approximation P_{EXACT} to the real solution from the known cell averages.
We use a second order ansatz for the polynomial

$$\begin{aligned} P_{\text{EXACT}}(x,y) &= u^n_{ij} + u^n_{ij,x}(x-x_j) + \frac{1}{2}u^n_{ij,xx}(x-x_j)^2 + \\ & \quad u^n_{ij,y}(y-y_j) + \frac{1}{2}u^n_{ij,yy}(y-y_j)^2 \end{aligned}$$

The five coefficients

$$u^n_{ij}\ ,\ \ u^n_{ij,x}\ ,\ \ u^n_{ij,xx}\ ,\ \ u^n_{ij,y}\ ,\ \ u^n_{ij,yy}$$

are determined by requiring the polynomial to conserve the cell averages

$$\bar{u}^n_{mn} \ \text{ for } \ (m,n) \in \{(i,j),(i+1,j),(i-1,j),(i,j+1),(i,j-1)\}.$$

The coefficients are given by

$$\begin{aligned} u^n_{ij} &= \bar{u}^n_{ij} - \frac{1}{24}(\bar{u}^n_{i+1,j} - 2\bar{u}^n_{ij} + \bar{u}^n_{i-1,j}) - \frac{1}{24}(\bar{u}^n_{i,j+1} - 2\bar{u}^n_{ij} + \bar{u}^n_{i,j-1}), \\ u^n_{ij,x} &= \frac{\bar{u}^n_{i+1,j} - \bar{u}^n_{i-1,j}}{2\Delta x}, \quad u^n_{ij,y} = \frac{\bar{u}^n_{i,j+1} - \bar{u}^n_{i,j-1}}{2\Delta x} \\ u^n_{ij,xx} &= \frac{\bar{u}^n_{i+1,j} - 2\bar{u}^n_{i,j} + \bar{u}^n_{i-1,j}}{\Delta x^2}, \quad u^n_{ij,yy} = \frac{\bar{u}^n_{i,j+1} - 2\bar{u}^n_{i,j} + \bar{u}^n_{i,j-1}}{\Delta y^2}. \end{aligned}$$

P_{EXACT} is a a good approximation to the real function $u(x, y; t^n)$

BUT it does not provide non-oscillatory behavior.

Solution: Weighted ENO

Discuss now construction of the interpolating polynomial for the x-direction.
Dimension-by-dimension approach

In each cell reconstruct quadratic polynomial as a convex combination of three polynomials

$$P_j(x) = w_L P_L(x) + w_R P_R(x) + w_C P_C(x),$$

with positive weights $w_i > 0$ and $\sum_i w_i = 1$, where $i \in \{L, R, C\}$.

The polynomials $P_L(x), P_R(x)$ correspond to left and right one-sided linear reconstructions, uniquely determined by requiring them to conserve the one-sided cell averages:

$$\begin{aligned}
\bar{u}_{ij} &= \int_{(i-1/2)\Delta x}^{(i+1/2)\Delta x} P_R(x)\, dx \;\text{ and }\; \bar{u}_{i,j+1} = \int_{(i+1/2)\Delta x}^{(i+3/2)\Delta x} P_R(x)\, dx \\
\bar{u}_{ij} &= \int_{(i-1/2)\Delta x}^{(i+1/2)\Delta x} P_L(x)\, dx \;\text{ and }\; \bar{u}_{i,j-1} = \int_{(i-3/2)\Delta x}^{(i-1/2)\Delta x} P_L(x)\, dx
\end{aligned}$$

The polynomial $P_C(x)$ is determined by

$$P_{\text{EXACT}}(x, y = y_j) = c_L P_L(x) + c_R P_R(x) + (1 - c_L - c_R) P_C(x)$$

Every symmetric selection of the coefficients $c_L = c_R$ will provide third-order accuracy.

Choosing $c_L = c_r = 1/4$ we obtain the polynomial $P_C(x)$ as

$$\begin{aligned} P_C(x) &= \bar{u}^n_{ij} - \frac{1}{12}(\bar{u}^n_{i+1,j} - 2\bar{u}^n_{ij} + \bar{u}^n_{i-1,j}) - \frac{1}{12}(\bar{u}^n_{i,j+1} - 2\bar{u}^n_{ij} + \bar{u}^n_{i,j-1}) \\ &+ \frac{\bar{u}^n_{i+1,j} - \bar{u}^n_{i-1,j}}{2\Delta x}(x - x_j) + \frac{1}{2}\frac{\bar{u}^n_{i+1,j} - 2\bar{u}^n_{i,j} + \bar{u}^n_{i-1,j}}{\Delta x^2}(x - x_j)^2 \end{aligned}$$

The weights w_i are used to automatically adapt the reconstruction to the smoothness of the solution. In smooth regions, they select the third-order reconstruction to provide maximum precision, whereas in the presence of discontinuities they switch to a one-sided reconstruction to guarantee the essentially non-oscillatory behavior.

The weights are taken as

$$w_i = \frac{\alpha_i}{\sum_m \alpha_m} \;, \quad \text{where} \;\; \alpha_i = \frac{c_i}{(\epsilon + IS_i)^p} \;, \quad i, m \in \{c, R.L\}$$

$$C_L = C_R = 1/4 \;, \quad C_C = 1/2$$

Smoothness indicator

$$IS_l = (\bar{u}_j^n - \bar{u}_{j-1}^n) \;, \qquad IS_l = (\bar{u}_{j+1}^n - \bar{u}_j^n)$$

$$IS_C = \frac{13}{3}(\bar{u}_{j+1}^n - 2\bar{u}_j^n + \bar{u}_{j-1}^n)^2 + \frac{1}{4}(\bar{u}_{j+1}^n - \bar{u}_{j-1}^n)^2$$

done

You won't believe it, but all this is really simple compared to Riemann solvers !!!

div B = 0 Problem

$$\partial_t \mathbf{B} + \nabla \cdot (\mathbf{u}\mathbf{B}^T - \mathbf{B}\mathbf{u}^T) = 0$$

$\nabla \cdot \mathbf{B} = 0$ at time $t = 0 \implies \nabla \cdot \mathbf{B} = 0$ at times $t > 0$

But: numerical errors $\implies \nabla \cdot \mathbf{B} \neq 0$

near shocks: $\nabla \cdot \mathbf{B} = O((\Delta x)^{-1})$

Purposes to control div B:
improve robustness and avoid unphysical effects (parallel Lorentz force)

Techniques:

- 8-wave formulation (Powell et al 1999)
  easy but
  div B not exactly zero, non-conservative, doesn't work to good for turbulence
- Constraint Transport (Evans, Hawley 1988, Dai, Woodward 1998, Balsara, Spicer 1999)
  div B = 0 up to machine precision but
  staggered formulation difficult for AMR
  needs entropy fix
  no local timestepping possible
  positivity of pressure is an issue
- Vector Potential (similar to CT, Londrillo and Del Zanna)
- Projection Method (Brackbill and Barnes)
  div B = 0, correct weak solution but
  very expensive, positivity of pressure is an issue
- divergence cleaning (Dedner et al 2002)
  easy, conservative but
  div B not exactly zero, doesn't work to good for turbulence
  pressure may become negative if energy is conserved

racoon: divergence cleaning

Divergence cleaning:

$$\begin{aligned} \partial_t \mathbf{B} + \nabla \cdot (\mathbf{u}\mathbf{B}^T - \mathbf{B}\mathbf{u}^T) + \nabla \psi &= 0 \\ \mathcal{D}(\psi) + \nabla \cdot \mathbf{B} &= 0 \end{aligned}$$

where $\mathcal{D}$ is a linear differential operator.

We obtain

$$\begin{aligned} \partial_t \nabla \cdot \mathbf{B} + \Delta \psi &= 0 \\ \partial_t \mathcal{D} \nabla \cdot \mathbf{B} + \mathcal{D} \Delta \psi &= 0 \\ \partial_t \mathcal{D}(\psi) + \partial_t \nabla \cdot \mathbf{B} &= 0 \\ \Delta \mathcal{D}(\psi) + \Delta \nabla \cdot \mathbf{B} &= 0 \end{aligned}$$

combine this

$$\begin{aligned} \partial_t \mathcal{D} \nabla \cdot \mathbf{B} - \Delta \nabla \cdot \mathbf{B} &= 0 \\ \partial_t \mathcal{D}(\psi) - \Delta \psi &= 0 \end{aligned}$$

$\Longrightarrow \nabla \cdot \mathbf{B}$ and ψ satisfy the same equation

The two important equations are:

$$\begin{aligned} \partial_t \mathcal{D}(\psi) - \Delta\psi &= 0 \\ \mathcal{D}(\psi) + \nabla \cdot \mathbf{B} &= 0 \end{aligned}$$

Case 1: $\mathcal{D}(\psi) = 0 \Longrightarrow$ Projection method

Case 2: (parabolic) $\mathcal{D}(\psi) = \frac{1}{c_p^2}\psi \Longrightarrow$

$$\begin{aligned} \partial_t \psi - c_p^2 \Delta\psi &= 0 \\ \psi + c_p^2 \nabla \cdot \mathbf{B} &= 0 \end{aligned}$$

Case 3: (hyperbolic) $\mathcal{D}(\psi) = \frac{1}{c_h^2}\partial_t\psi \Longrightarrow$

$$\begin{aligned} \partial_{tt} \psi - c_h^2 \Delta\psi &= 0 \\ \partial_t\psi + c_h^2 \nabla \cdot \mathbf{B} &= 0 \end{aligned}$$

Case 4: (mixed hyperbolic + parabolic) $\mathcal{D}(\psi) = \frac{1}{c_h^2}\partial_t\psi + \frac{1}{c_p^2}\psi \Longrightarrow$

$$\begin{aligned} \partial_{tt} \psi + \frac{c_h^2}{c_p^2}\partial_t\psi - c_h^2 \Delta\psi &= 0 \\ \partial_t\psi + \frac{c_h^2}{c_p^2}\psi + c_h^2 \nabla \cdot \mathbf{B} &= 0 \end{aligned}$$

works very good for localized structures as in FlareLab, but not in MHD turbulence

Charge and current deposition

$$\partial \mathbf{B}/\partial t = -c(\nabla \times E) \quad \Longrightarrow \quad \partial \nabla \cdot \mathbf{B}/\partial t = 0$$

Yee grid ✓

$$\partial E/\partial t = c(\nabla \times B) - 4\pi J$$

$$\frac{\partial \nabla \cdot E}{\partial t} = \cancel{c\nabla \cdot (\nabla \times B)} - 4\pi \nabla \cdot J$$

$$\frac{\partial \rho}{\partial t} = -\nabla \cdot J$$

continuity equation is again an initial condition

$$\frac{\partial \boldsymbol{E}}{\partial t} - c^2 \nabla \times \boldsymbol{B} + c^2 \nabla \Phi = -\frac{\boldsymbol{j}}{\epsilon_o} \,,$$

$$\frac{\partial \boldsymbol{B}}{\partial t} + \nabla \times \boldsymbol{E} = 0 \,,$$

$$\mathcal{D}(\Phi) + \nabla \cdot \boldsymbol{E} = \frac{\rho}{\epsilon_0} \,,$$

$$\nabla \cdot \boldsymbol{B} = 0 \,,$$

$$\mathcal{D}(\Phi) \equiv 0 \quad \Longrightarrow \quad -c^2 \nabla^2 \Phi = \frac{1}{\epsilon_0} \left(\frac{\partial \rho}{\partial t} + \nabla \cdot \boldsymbol{j} \right)$$

projection method

$$\mathcal{D}(\Phi) = \frac{\Phi}{\chi} \quad \Longrightarrow \quad \frac{\partial \Phi}{\partial t} - \chi c^2 \nabla^2 \Phi = \frac{\chi}{\epsilon_0} \left(\frac{\partial \rho}{\partial t} + \nabla \cdot \boldsymbol{j} \right)$$

parabolic

$$\mathcal{D}(\Phi) = \frac{1}{\chi^2} \frac{\partial \Phi}{\partial t} \quad \Longrightarrow \quad \frac{\partial^2 \Phi}{\partial t^2} - (\chi c)^2 \, \nabla^2 \Phi = \frac{\chi^2}{\epsilon_0} \left(\frac{\partial \rho}{\partial t} + \nabla \cdot \boldsymbol{j} \right)$$

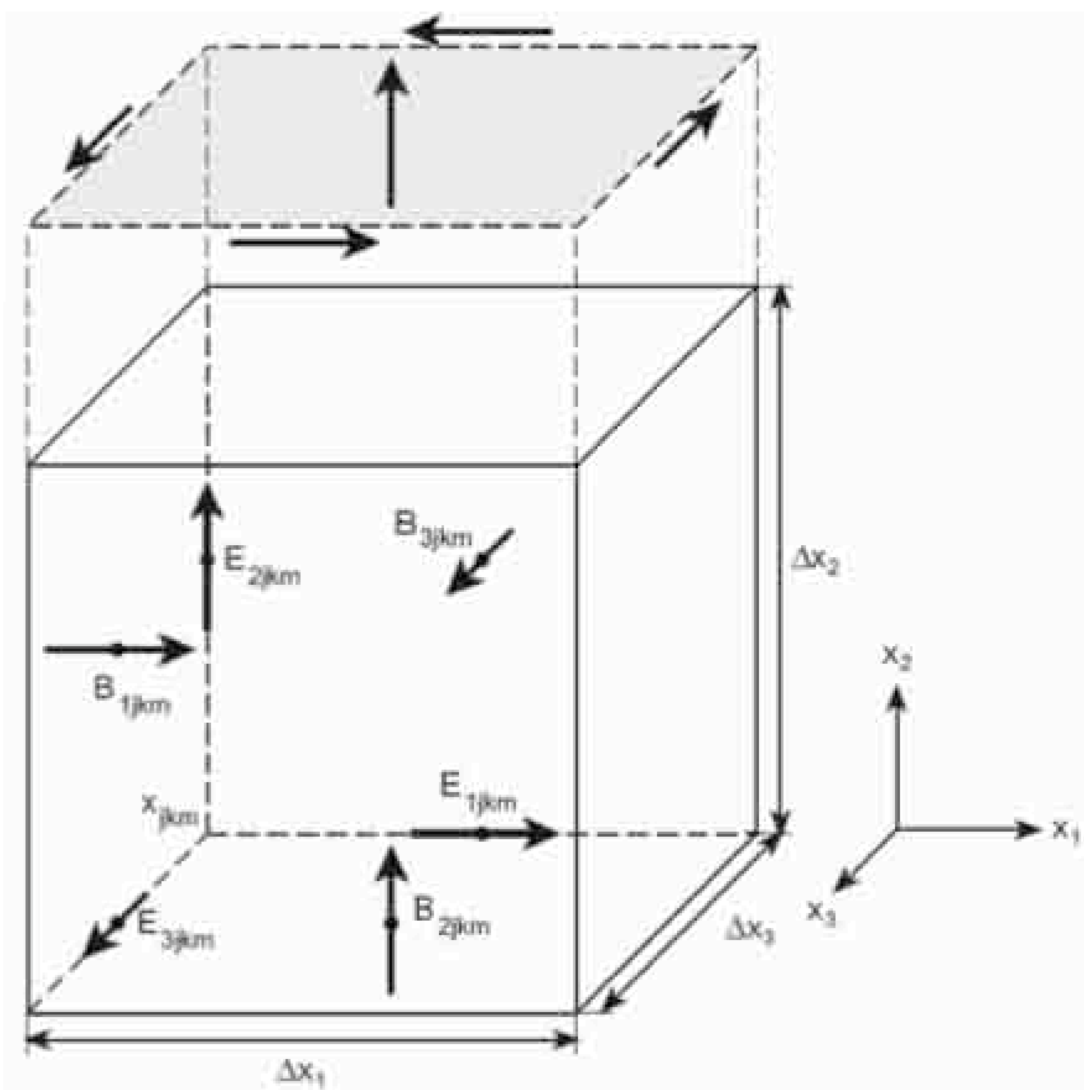
hyperbolic

Maxwell Solver: FDTD and Yee mesh (1966)

inspired by lectures by A. Spitkovsky

$$\partial \boldsymbol{E}/\partial t = c(\boldsymbol{\nabla} \times \boldsymbol{B}) - 4\pi \boldsymbol{J}\,, \qquad \boldsymbol{\nabla} \cdot \boldsymbol{E} = 4\pi\varrho\,, \quad \boldsymbol{\nabla} \cdot \boldsymbol{B} = 0$$

$$\partial \boldsymbol{B}/\partial t = -c(\boldsymbol{\nabla} \times \boldsymbol{E})\,, \qquad \frac{d}{dt}\gamma m\mathbf{v} = q(\mathbf{E} + \frac{\mathbf{v}}{c} \times \mathbf{B})$$



FDTD: second order in space and

$$\boldsymbol{E}^{n+1/2} = \boldsymbol{E}^{n-1/2} + \Delta t[c(\boldsymbol{\nabla} \times \boldsymbol{B}^n) - 4\pi \boldsymbol{J}^n]$$

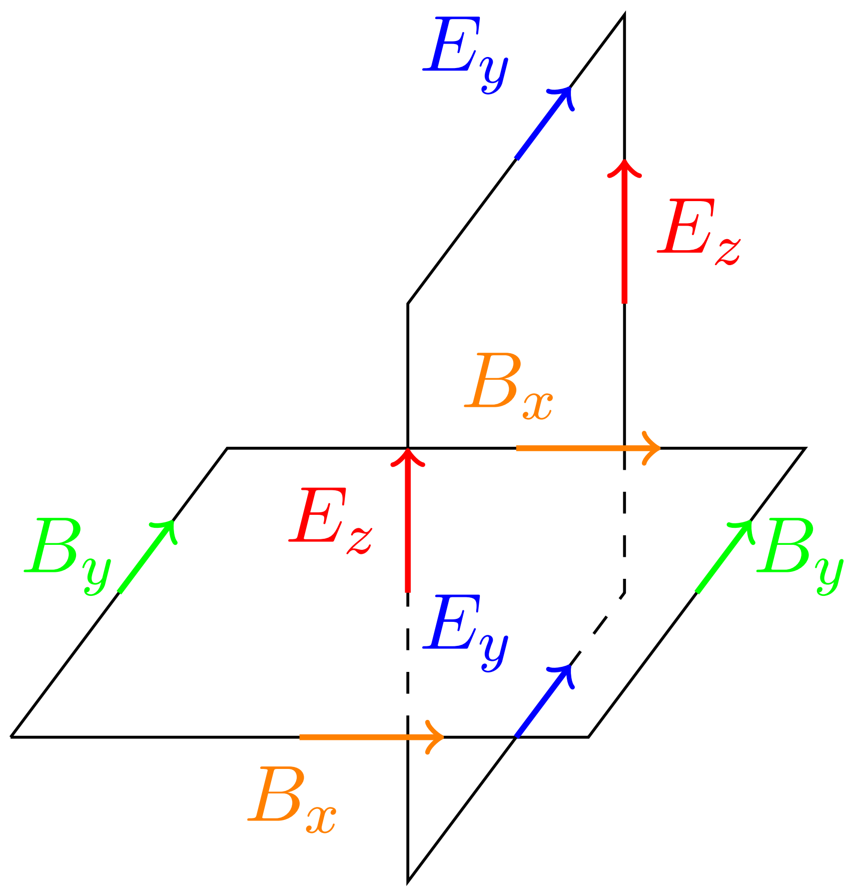
$$\boldsymbol{B}^{n+1} = \boldsymbol{B}^n - c\Delta t \boldsymbol{\nabla} \times \boldsymbol{E}^{n+1/2}$$

Yee mesh: div B

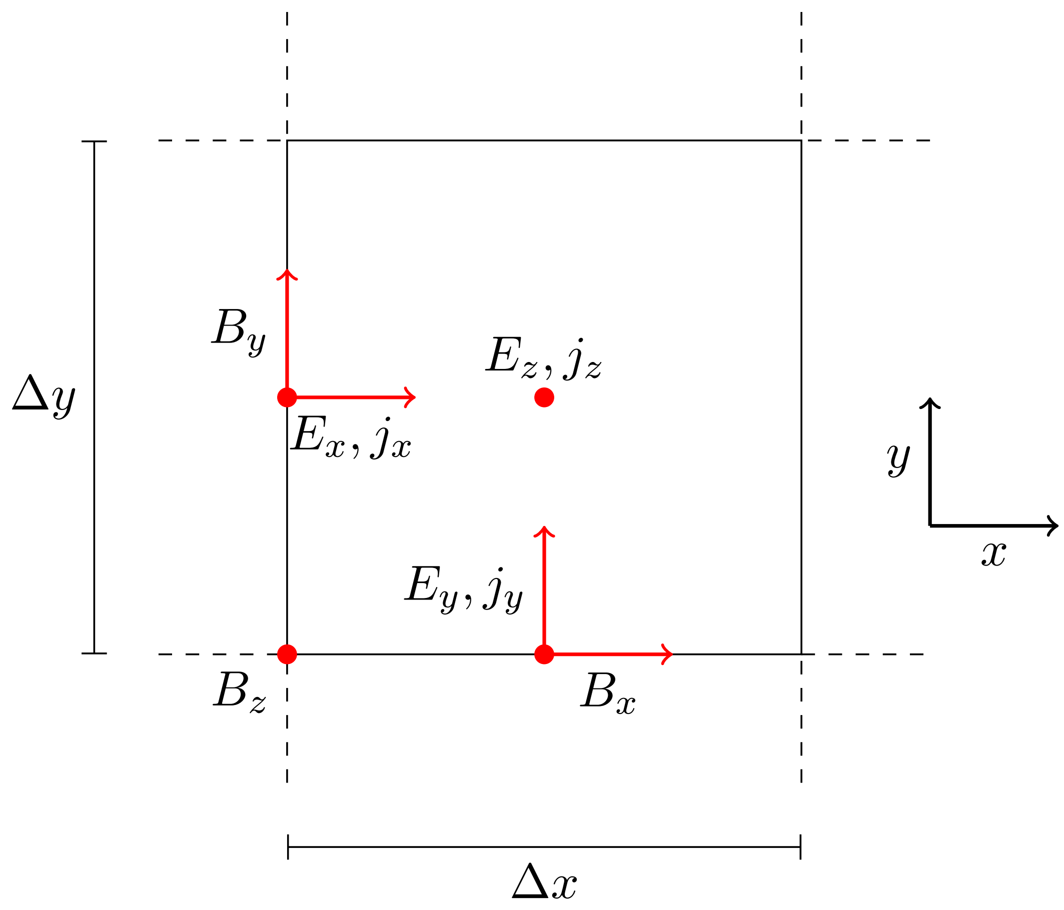
Yee mesh motivated by integral form:

$$\partial_t \int_\Sigma \mathbf{B} \cdot \mathrm{d}\mathbf{S} = -\oint_{\partial\Sigma} \mathbf{E} \cdot \mathrm{d}\mathbf{l}$$

$$\partial_t \int_\Sigma \mathbf{E} \cdot \mathrm{d}\mathbf{S} = -c^2 \int_\Sigma \mathbf{j} \cdot \mathrm{d}\mathbf{S} + c^2 \oint_{\partial\Sigma} \mathbf{B} \cdot \mathrm{d}\mathbf{l}$$



2D by projection

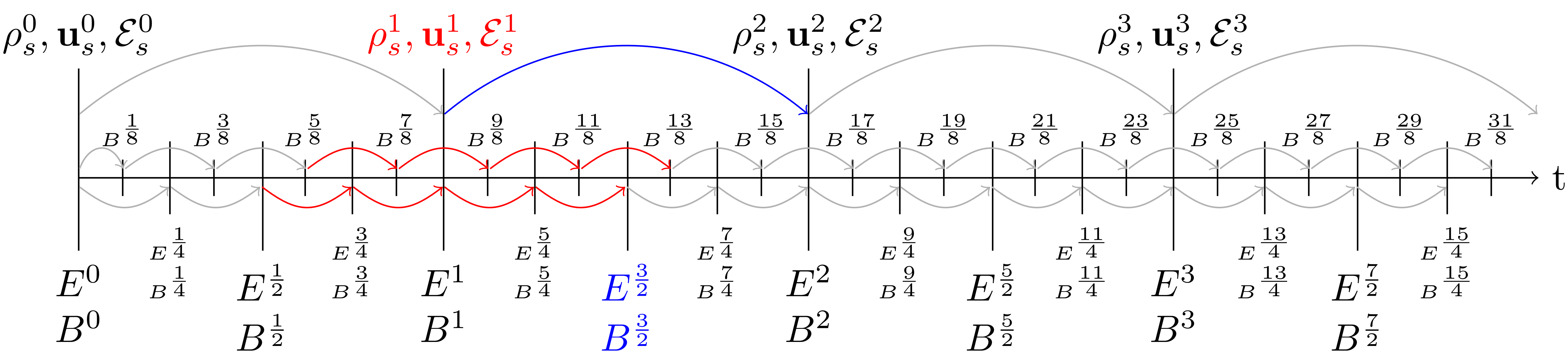


Coupling FDTD- and CWENO Method

Fluid: strongly stable TVD Runge Kutta (Shu-Osher 1988)

$$
\begin{aligned}
v' &= v^n + \frac{\Delta t}{6} f\left(v^n, t^n\right) \\
v'' &= v' + \frac{\Delta t}{6} f\left(6v' - 5v^n, t^n + \Delta t\right) \\
v^{n+1} &= v'' + \frac{2\Delta t}{3} f\left(\tfrac{3}{2}v'' - \tfrac{1}{2}v^n, t^n + \tfrac{1}{2}\Delta t\right)
\end{aligned}
$$

subcycling and interpolation



2D Simulations: GEM Setup

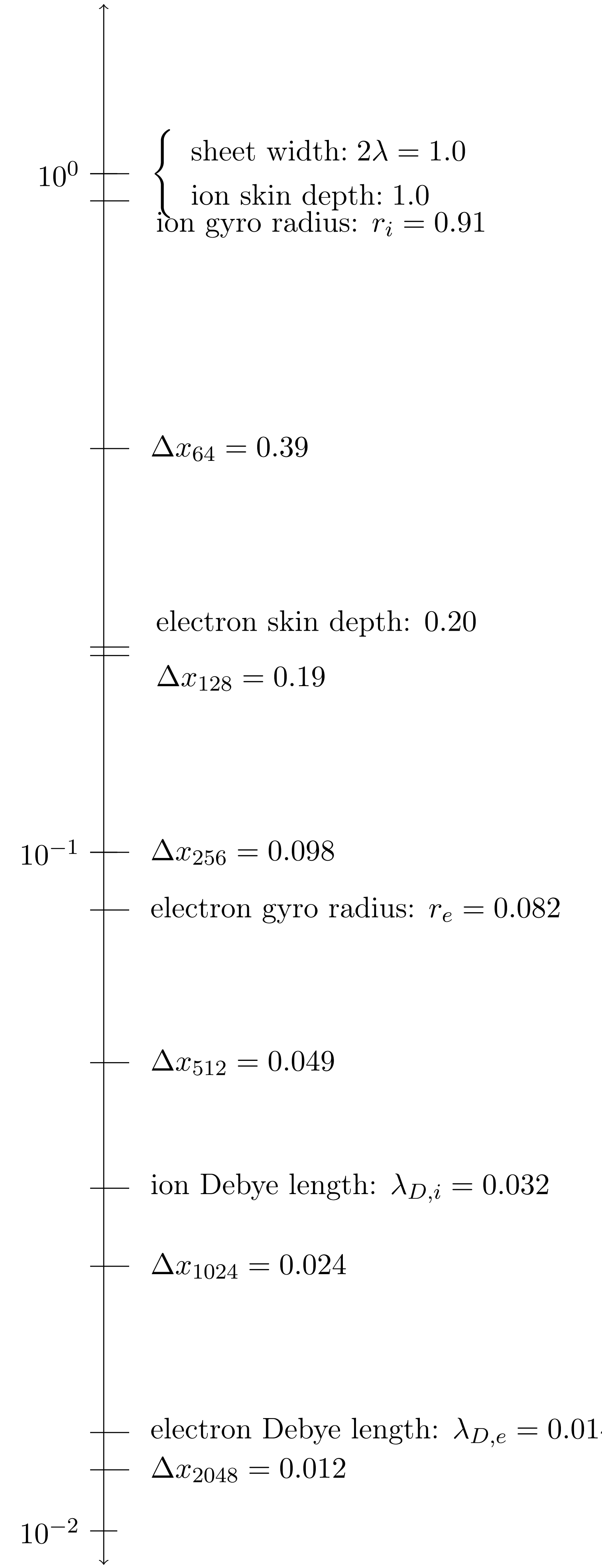
Parameters:

$\frac{m_i}{m_e} = 25$ | $\frac{T_i}{T_e} = \sqrt{\frac{m_i}{m_e}} = 5$ | $\lambda = 0.5$

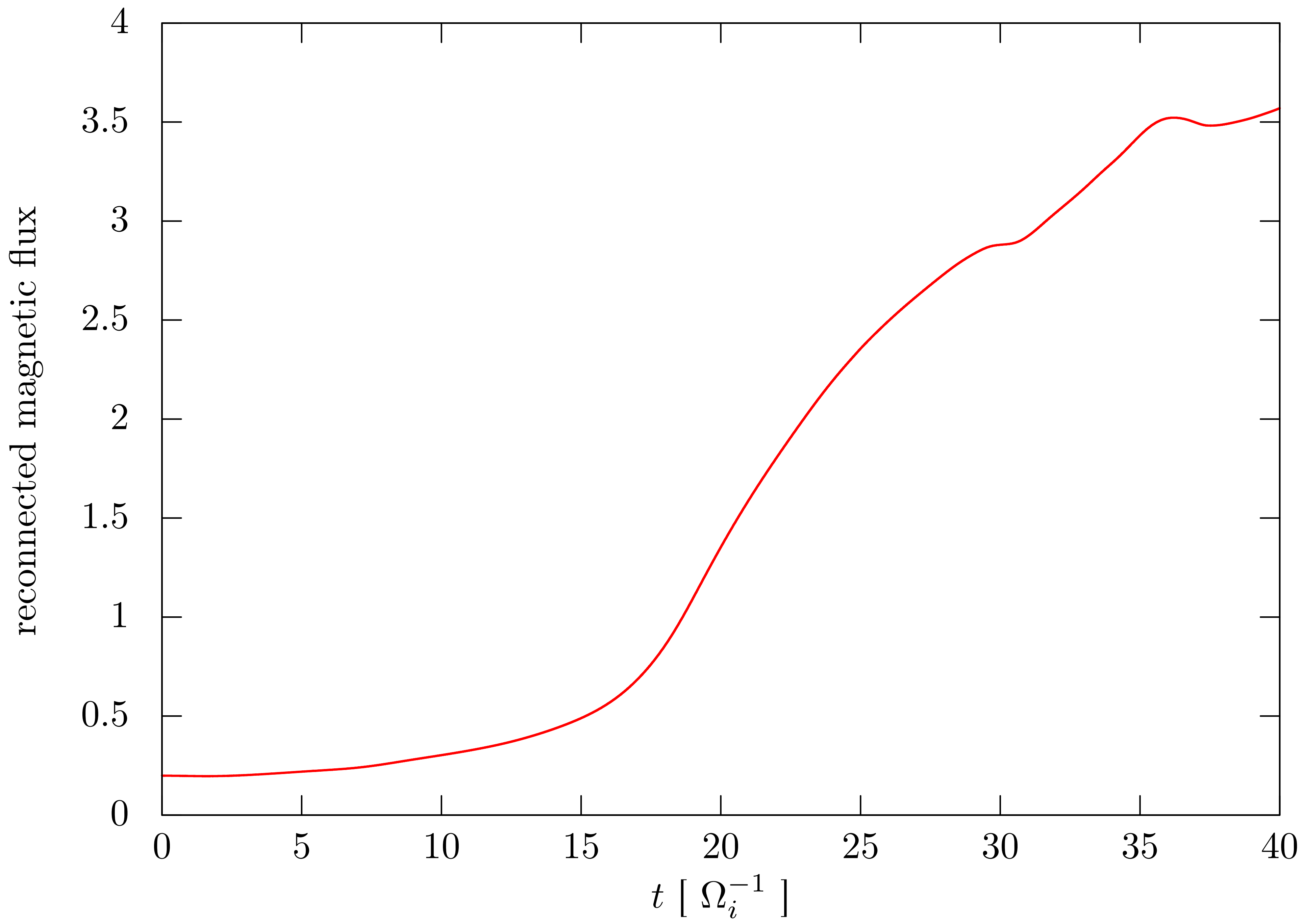
$n_0 = 1$ | $n_\infty = 0.2$ | $B_0 = 1$

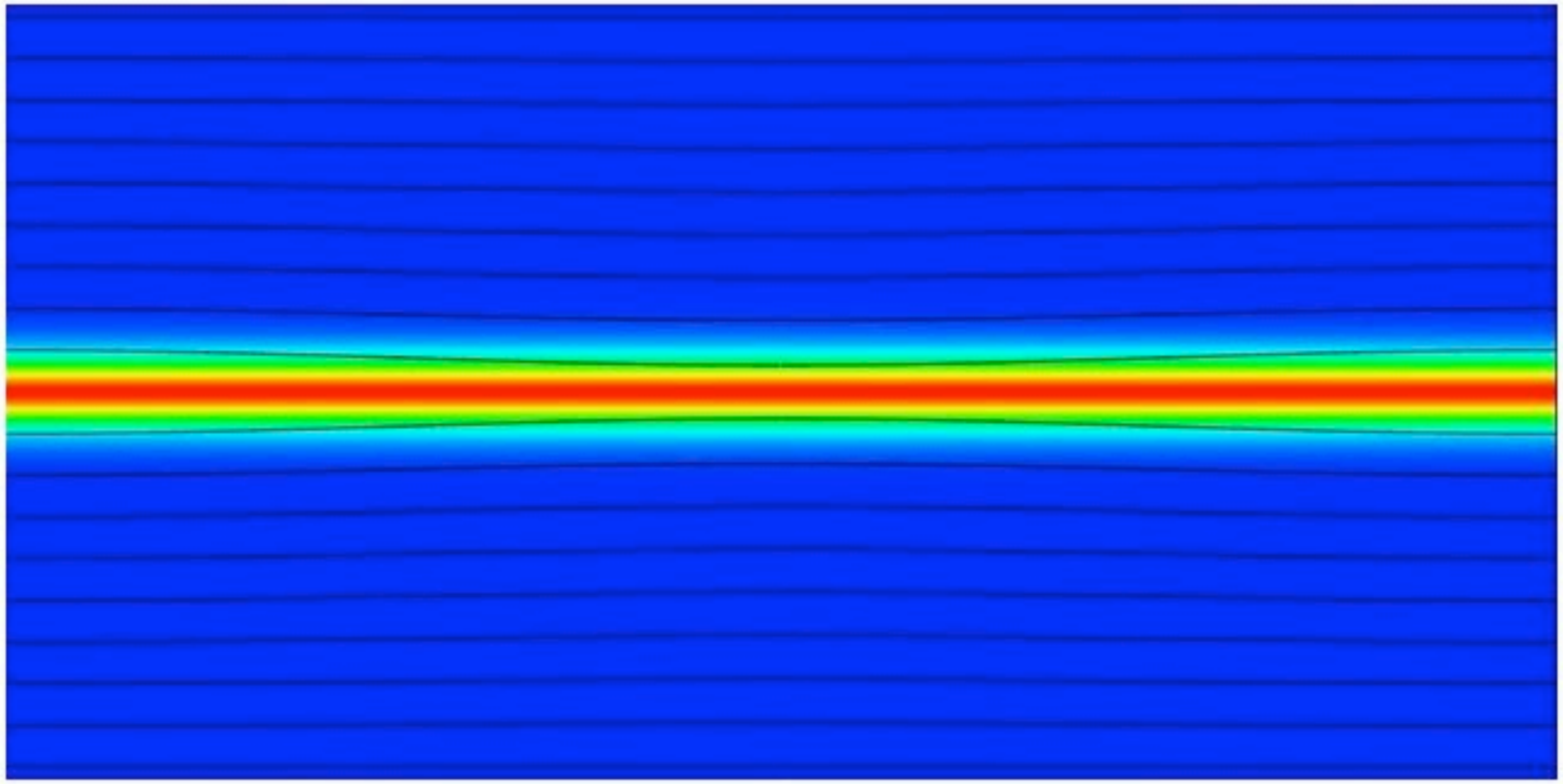
$\psi_0 = 0.1$ | $L_x = 8\pi$ | $L_y = 4\pi$

Name	Expression	Electrons	Ions
thermal velocity	$v_{\mathrm{th},s} = \sqrt{2T_{0,s}}\sqrt{\frac{m_i}{m_s}}$	2.0	0.91
plasma frequency	$\omega_{\mathrm{p},s} = c\sqrt{\frac{m_i}{m_s}}\sqrt{n_{0,s}}$	100	20
gyro frequency	$\Omega_s = \frac{m_i}{m_s}B_0$	25	1
Larmor radius	$r_s = \sqrt{2T_{0,s}}\sqrt{\frac{m_s}{m_i}}\frac{1}{B_0}$	0.082	0.91
Debye length	$\lambda_{\mathrm{D},s} = \frac{1}{c}\sqrt{\frac{T_{0,s}}{n_{0,s}}}$	0.014	0.032
skin depth/inertial length	$\delta_s = \sqrt{\frac{m_s}{m_i}}\frac{1}{\sqrt{n_{0,s}}}$	0.2	1

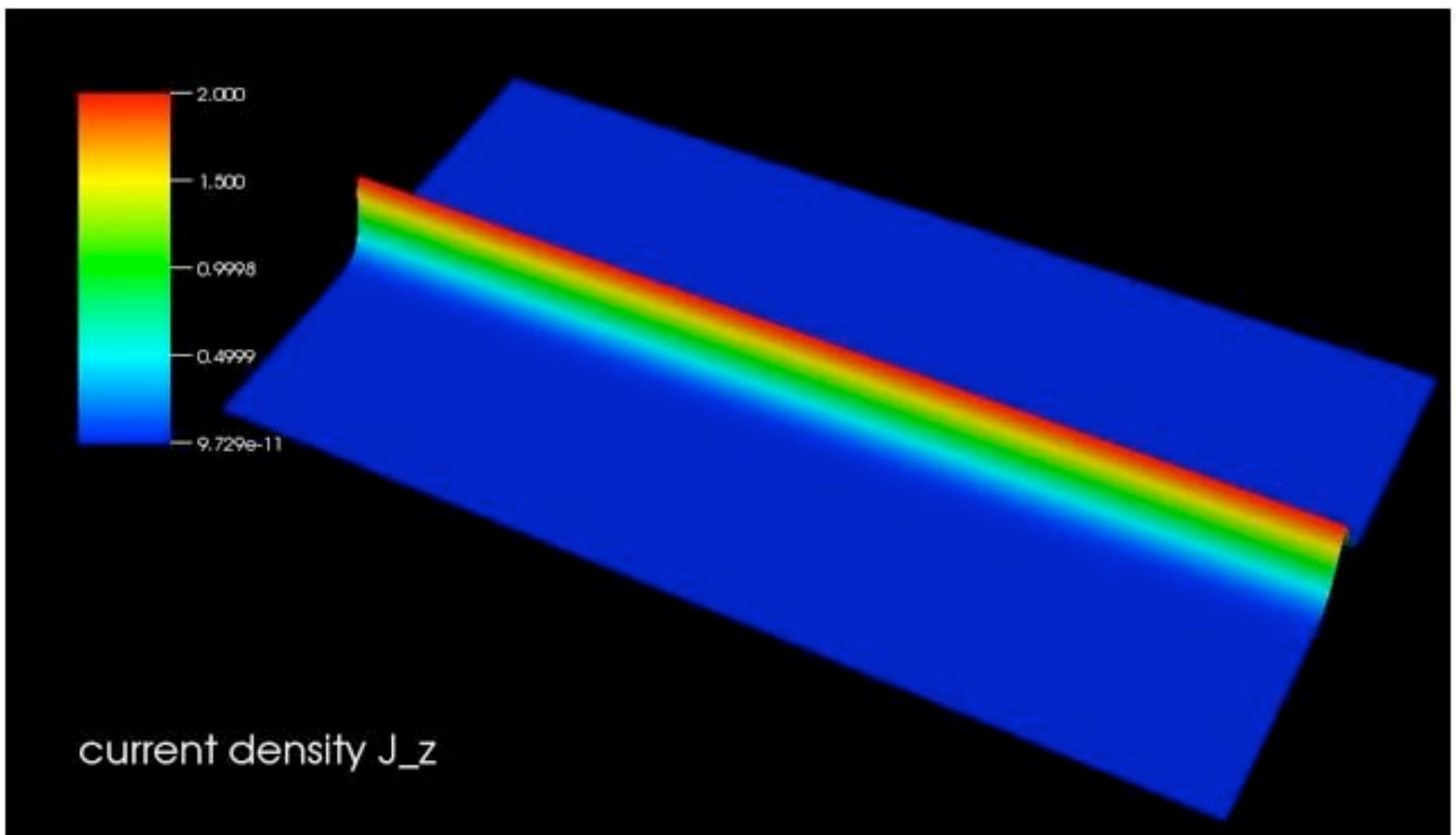


Reconnected flux





electron density



current density j_z

Ok, now we have a fluid code !

Let's do Vlasov

Vlasov simulations

collisionless Plasma: Vlasov equation

$$\frac{\partial f_k}{\partial t} + \mathbf{v} \cdot \nabla_{\mathbf{x}} f_k + \frac{q_k}{m_k} (\mathbf{E} + \mathbf{v} \times \mathbf{B}) \cdot \nabla_{\mathbf{v}} f_k = 0$$

$+$ Maxwell, $k = e, i$

important: positive conservative scheme, semi-Lagrangian, Boris, backsubstitution method

(Filbet, Sonnendrücker, Bertrand 2001)

Darwin-Approximation

CFL-condition too restrictive
$\Longrightarrow$ Darwin approximation

electric field split into *longitudinal* and *transversal* part

$$\mathbf{E} = \mathbf{E}_L + \mathbf{E}_T \quad \text{mit } \nabla \cdot \mathbf{E}_T = 0 \text{und } \nabla \times \mathbf{E}_L = 0$$

Maxwell equations

$$\nabla \times \mathbf{E} \quad = -\frac{\partial \mathbf{B}}{\partial t} \qquad\qquad \nabla \cdot \mathbf{E} \quad = \frac{\rho}{\varepsilon_0}$$

$$\nabla \times \mathbf{B} = \mu_0 \left(\varepsilon_0 \frac{\partial \mathbf{E}}{\partial t} + \mathbf{j} \right) \qquad\qquad \nabla \cdot \mathbf{B} = 0$$

Darwin-Approximation

CFL-condition too restrictive
$\Longrightarrow$ Darwin approximation

electric field split into *longitudinal* and *transversal* part

$$\mathbf{E} = \mathbf{E}_L + \mathbf{E}_T \quad \text{mit } \nabla \cdot \mathbf{E}_T = 0 \text{und } \nabla \times \mathbf{E}_L = 0$$

Maxwell equations

$$\nabla \times \mathbf{E_T} = -\frac{\partial \mathbf{B}}{\partial t} \qquad \nabla \cdot \mathbf{E_L} = \frac{\rho}{\varepsilon_0}$$

$$\nabla \times \mathbf{B} = \mu_0 \left(\varepsilon_0 \frac{\partial \mathbf{E}}{\partial t} + \mathbf{j} \right) \qquad \nabla \cdot \mathbf{B} = 0$$

Darwin-Approximation

CFL-condition too restrictive
$\Longrightarrow$ Darwin approximation

electric field split into *longitudinal* and *transversal* part

$$\mathbf{E} = \mathbf{E}_L + \mathbf{E}_T \quad \text{mit } \nabla \cdot \mathbf{E}_T = 0 \text{und } \nabla \times \mathbf{E}_L = 0$$

Maxwell equations with Darwin approximation

$$\nabla \times \mathbf{E_T} = -\frac{\partial \mathbf{B}}{\partial t} \qquad \nabla \cdot \mathbf{E_L} = \frac{\rho}{\varepsilon_0}$$

$$\nabla \times \mathbf{B} = \mu_0 \left(\varepsilon_0 \frac{\partial \mathbf{E_L}}{\partial t} + \mathbf{j} \right) \qquad \nabla \cdot \mathbf{B} = 0$$

no timestep restriction by the speed of light, but 8 elliptic equations

Semi-Lagrangean scheme

Consider $\partial_t f + \partial_x \left(v(t,x) f \right) = 0$

The characteristics of this PDE are given by:

$$\begin{aligned} \frac{dX}{ds}(s) &= v(s, X(s)) \\ X(t) &= x \end{aligned}$$

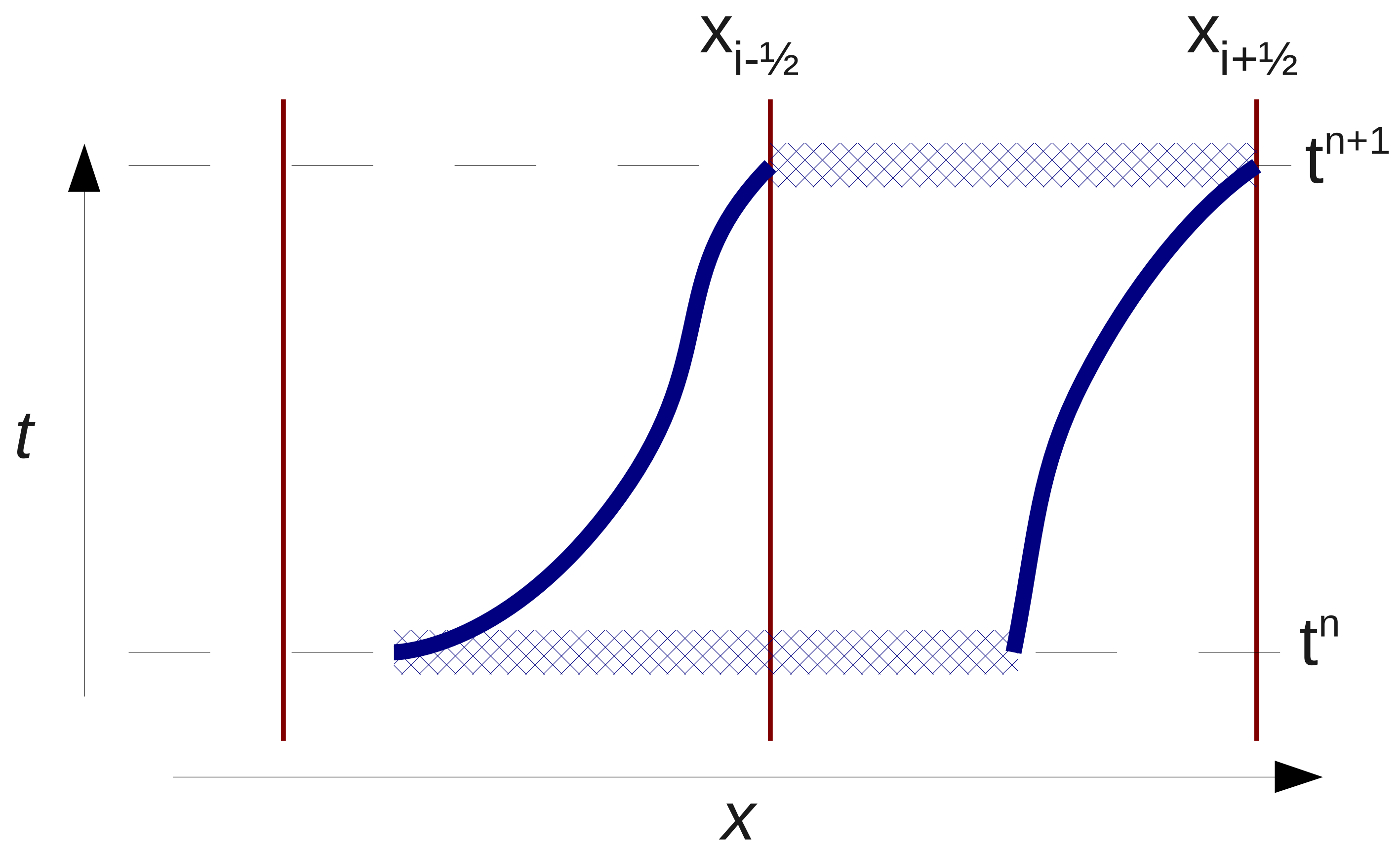
Denote the solution as $X(s,t,x)$

Since $\frac{df}{ds} = 0$ (r.h.s. of the PDE), we have

$$\int_{x_1}^{x_2} f(t,x)dx = \int_{X(s,t,x_1)}^{X(s,t,x_2)} f(s,x)dx$$

With this we can update the cell-average of f in the ith cell:

$$\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} f(t^{n+1},x)dx = \int_{X(t^n,t^{n+1},x_{i-\frac{1}{2}})}^{X(t^n,t^{n+1},x_{i+\frac{1}{2}})} f(t^n,x)dx$$



The integral of f over the hatched area is conserved. “This part of the fluid will always stay between the two characteristics.”

$$\int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} f(t^{n+1},x)dx = \int_{X(t^n,t^{n+1},x_{i-\frac{1}{2}})}^{X(t^n,t^{n+1},x_{i+\frac{1}{2}})} f(t^n,x)dx$$

Let $\bar{f}_i^n$ denote the cell-average in the ith cell at time t^n.

$$\begin{aligned}
\bar{f}_i^{n+1} &= \bar{f}_i^n + \Phi_{i-\frac{1}{2}} - \Phi_{i+\frac{1}{2}} \\
&= \bar{f}_i^n + \int_{X(t^n,t^{n+1},x_{i-\frac{1}{2}})}^{x_{i-\frac{1}{2}}} f(t^n,x)\mathrm{d}x - \int_{x_{i+\frac{1}{2}}}^{X(t^n,t^{n+1},x_{i+\frac{1}{2}})} f(t^n,x)\mathrm{d}x
\end{aligned}$$

Strategy:

- Follow the Characteristics ending at the cell borders backwards in time and find their footpoint
- Reconstruct the integral of f from the footpoint to the cell border
- Update with $\bar{f}_i^{n+1} = \bar{f}_i^n + \Phi_{i-\frac{1}{2}} - \Phi_{i+\frac{1}{2}}$

This will lead to a conservative scheme.

Developed by *Filbet, Sonnendrücker, Bertrand (JCP 2001)*
PFC = *Positive Flux-Conservative*

Let's consider the simple second-order scheme for positive velocities: Approximate the primit function of f in the interval $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}]$ (again, $\bar{f}_i$ denotes the cell average):

$$F(x) = \int_{-\infty}^{x} f(x)\mathrm{d}x$$

by

$$\tilde{F}(x) = w_{i-1} + (x - x_{i-\frac{1}{2}})\bar{f}_i + \frac{1}{2}(x - x_{i-\frac{1}{2}})(x - x_{i+\frac{1}{2}})\frac{\bar{f}_{i+1} - \bar{f}_i}{\Delta x}$$

Now we can reconstruct f itself:

$$\tilde{f}(x) = \frac{\mathrm{d}F}{\mathrm{d}x}(x) = \bar{f}_i + (x - x_i)\frac{\bar{f}_{i+1} - \bar{f}_i}{\Delta x}$$

However this scheme can cause negative reconstructed $\tilde{f}$. To avoid this, one can introduce a slope-limiter ϵ to ensure that the reconstruction lies between 0 and f_∞:

$$\epsilon_i = \begin{cases} \min(1; 2\bar{f}_i/(\bar{f}_{i+1} - \bar{f}_i)) & \text{if } \bar{f}_{i+1} > \bar{f}_i \\ \min(1; -2(f_\infty - \bar{f}_i)/((\bar{f}_{i+1} - \bar{f}_i)) & \text{if } \bar{f}_{i+1} < \bar{f}_i, \end{cases}$$

to obtain

$$f_h(x) = \bar{f}_i + \epsilon_i(x - x_i)\frac{\bar{f}_{i+1} - \bar{f}_i}{\Delta x}$$

Let's denote the distance from the footpoint of the characteristic to the cell-boundary by α. Integrating f_h then gives the flux through the boundary at $x_{i+\frac{1}{2}}$:

$$\begin{aligned} \Phi_{i+\frac{1}{2}} &= \int_{x_{i+\frac{1}{2}-\alpha}}^{x_{i+\frac{1}{2}}} f_h(x)\mathrm{d}x \\ &= \alpha\left(\bar{f}_i + \frac{\epsilon_i}{2}\left(1 - \frac{\alpha}{\Delta x}\right)(\bar{f}_{i+1} - \bar{f}_i)\right) \end{aligned}$$

Some remarks:

- This scheme can be extended to higher orders. We use the third order one.
- A similar derivation produces the scheme for negative velocities.
- The length of the characteristics can be arbitrarily large with only a minor change in the derivation.
- The accuracy in time depends only on how good the characteristics can be calculated.

The Vlasov equation

$$\partial_t f_s + \mathbf{v} \cdot \nabla_{\mathbf{x}} f_s + \frac{q_s}{m_s} (\mathbf{E} + \mathbf{v} \times \mathbf{B}) \cdot \nabla_{\mathbf{v}} f_s = 0$$

We want to solve this PDE using a one-dimensional semi-Lagrangian scheme.
Why? Becase one-dimensional schemes can have fancy limiters, conservation-properties and efficient implementations that are difficult to generalise to higher dimensions.
Remember: The Vlasov equation is a conservative, hyperbolic PDE in 6 dimension (plus time)

One way to do this is *splitting*.

Splitting

Consider $\partial_t f = \mathcal{A} f + \mathcal{B} f$, where $\mathcal{A}$ and $\mathcal{B}$ are linear operators (with no time dependance).

The formal solution to this is

$$f(t) = \exp\big((\mathcal{A} + \mathcal{B})t\big) f_0$$

If $\mathcal{A}$ and $\mathcal{B}$ commute, we can also write:

$$f(t) = \exp(\mathcal{B}t) \exp(\mathcal{A}t) f_0$$

This means we can just solve $\partial_t f = \mathcal{A} f$, use the result as an initial value for $\partial_t f = \mathcal{B} f$ and still get the correct solution!

Godunov splitting

What happens when $\mathcal{A}$ and $\mathcal{B}$ do *not* commute?
Let's look at the *Zassenhaus* formula (A variation on *Baker-Campbell-Hausdorff*):

$$\exp\left((\mathcal{A}+\mathcal{B})t\right) = \exp\left(\mathcal{B}t\right)\exp\left(\mathcal{A}t\right)\exp\left([\mathcal{A},\mathcal{B}]\frac{t^2}{2}\right)\exp\left(\mathcal{O}(t^3)\right)$$

So now we have:

$$f(t) = \exp(\mathcal{B}t)\exp(\mathcal{A}t)f_0 + \mathcal{O}(t^2)$$

We still get an approximate solution accurate to first order in time.
This is called *Godunov* splitting or *Lie-Trotter* splitting

Strang splitting

Can we do better?

A scheme accurate to second order in time is the *Strang-Splitting*:

$$f(t) = \exp(\mathcal{B}t/2)\exp(\mathcal{A}t)\exp(\mathcal{B}t/2)f_0 + \mathcal{O}(t^3)$$

By manipulating the *Baker-Campbell-Hausdorff* formula, splitting schemes of arbitrary order can be constructed.

However, the *Sheng-Suzuki theorem* states that all splitting schemes better than second order will have at least one negative exponent (i.e. negative time-steps).

Strang splitting and the Vlasov equation

We will now use Strang splitting on the Vlasov equation:

$$\partial_t f_s + \underbrace{\mathbf{v} \cdot \nabla_{\mathbf{x}}}_{\mathcal{A}} f_s + \underbrace{\frac{q_s}{m_s} \left(\mathbf{E} + \mathbf{v} \times \mathbf{B}\right) \cdot \nabla_{\mathbf{v}}}_{\mathcal{B}} f_s = 0$$

$$f_s(t^{n+1}) = \exp(\mathcal{B}t/2) \exp(\mathcal{A}t) \exp(\mathcal{B}t/2) f_s(t^n) + \mathcal{O}(t^3)$$

This means we update the velocity-part of f_s over one half time-step,
then update the position-part over one full time-step,
then update the velocity-part again over one half time-step.

This is equivalent to the *Leapfrog* or *Strömer-Verlet* schemes in PIC simulations!

The position update

We want to solve

$$\partial_t f_s + \mathbf{v} \cdot \nabla_{\mathbf{x}} f_s = 0$$

Let's rewrite this equation to

$$\partial_t f_s + \partial_x v_x f_s + \partial_y v_y f_s + \partial_z v_z f_s = 0$$

Since $\mathbf{v}$ is just a variable and does not depend on $\mathbf{x}$, we can write this in a conservative form. Now we have three linear operators that all commute!

We can just solve each step seperately and the solution is still exact. By using a conservative numerical scheme, the conservation property of the Vlasov equation is kept.

The velocity update

The velocity part is not that easy.

$$
\begin{aligned}
\partial_t f_s + \frac{q_s}{m_s}\left(\mathbf{E} + \mathbf{v} \times \mathbf{B}\right) \cdot \nabla_{\mathbf{v}} f_s = \\
\partial_t f_s + \frac{q_s}{m_s} \partial_{v_x} (E_x + v_y B_z - v_z B_y) f_s \\
+ \frac{q_s}{m_s} \partial_{v_y} (E_y + v_z B_x - v_x B_z) f_s \\
+ \frac{q_s}{m_s} \partial_{v_z} (E_z + v_x B_y - v_y B_x) f_s = 0
\end{aligned}
$$

We can still rewrite this in a conservative way, but the three operators do not commute because of the velocity in the $\mathbf{v} \times \mathbf{B}$ term.

The velocity update

Can we use Strang splitting?

If we denote the individual operators by $\mathcal{V}_x$, $\mathcal{V}_y$, and $\mathcal{V}_z$ we will have

$$
\begin{aligned}
f(t^{n+1}) = {} & \exp(\mathcal{V}_x t/4)\exp(\mathcal{V}_y t/2)\exp(\mathcal{V}_x t/4) \\
& \times \exp(\mathcal{V}_z t) \\
& \times \exp(\mathcal{V}_x t/4)\exp(\mathcal{V}_y t/2)\exp(\mathcal{V}_x t/4) f(t^n) + \mathcal{O}(t^3)
\end{aligned}
$$

This means 7 steps for the velocity update and we have a numerically preferred direction.

Backsubstitution

What we really want is:

- Just one step per operator
- No splitting error in time

Equations of motion:

$$\frac{d}{dt} m\mathbf{v} = q(\mathbf{E} + \mathbf{v} \times \mathbf{B})$$
$$\frac{d}{dt}\mathbf{x} = \mathbf{v}$$

looks implicit

leap-frog

$$\frac{\mathbf{v}^{n+1/2} - \mathbf{v}^{n-1/2}}{\Delta t} = \frac{q}{m}\left(\mathbf{E}^n + \frac{1}{2}(\mathbf{v}^{n+1/2} + \mathbf{v}^{n-1/2}) \times \mathbf{B}^n\right)$$

Solution: Boris (1970)

$$\mathbf{v}^{n-1/2} = \mathbf{v}^{-} - \frac{q\mathbf{E}^n}{m}\frac{\Delta t}{2}$$

$$\mathbf{v}^{n+1/2} = \mathbf{v}^{+} + \frac{q\mathbf{E}^n}{m}\frac{\Delta t}{2}$$

$$\frac{\mathbf{v}^{+} - \mathbf{v}^{-}}{\Delta t} = \frac{q}{2m}(\mathbf{v}^{+} + \mathbf{v}^{-}) \times \mathbf{B}$$

explicit

$$\mathbf{v}^{-} = \mathbf{v}^{n-1/2} + \frac{q\Delta t\mathbf{E}^n}{2m}$$

$$\mathbf{v}' = \mathbf{v}^{-} + \mathbf{v}^{-} \times \mathbf{t}^n$$

$$\mathbf{v}^{+} = \mathbf{v}^{-} + \mathbf{v}' \times \frac{2\mathbf{t}^n}{1 + \mathbf{t}^n \cdot \mathbf{t}^n}$$

$$\mathbf{v}^{n+1/2} = \mathbf{v}^{+} + \frac{q\Delta t\mathbf{E}^n}{2m}$$

$$\text{with } \mathbf{t}^n = \frac{q\Delta t\mathbf{B}^n}{2m}$$

Proof:

We know:

$$\mathbf{v}^+ - \mathbf{v}^- = \frac{q\Delta t}{2m}(\mathbf{v}^+ + \mathbf{v}^-) \times \mathbf{B}$$

We want to proof:

$$\mathbf{v}^+ - \mathbf{v}^- = \mathbf{v}' \times \mathbf{s}$$

$$\mathbf{v}' = \mathbf{v}^- + \mathbf{v}^- \times \mathbf{t} \ , \quad \mathbf{t} = \frac{q\Delta t}{2m}\mathbf{B} \ , \quad \mathbf{s} = \frac{2\mathbf{t}}{1+\mathbf{t}^2}$$

thus:

$$\mathbf{v}^+ - \mathbf{v}^- = \boxed{\mathbf{v}^- \times \mathbf{s}} + \boxed{(\mathbf{v}^- \times \mathbf{t}) \times \mathbf{s}}$$

$$\begin{aligned} \boxed{\mathbf{v}^- \times \mathbf{s}} &= \mathbf{v}^- \times \mathbf{B}\frac{q\Delta t}{2m}\frac{2}{1+t^2} \\ &= -\mathbf{v}^+ \times \mathbf{B}\frac{q\Delta t}{2m}\frac{2}{1+t^2} + (\mathbf{v}^+ - \mathbf{v}^-)\frac{2}{1+t^2} \\ &= \boxed{-(\mathbf{v}^+ - \mathbf{v}^-) \times \mathbf{B}\frac{q\Delta t}{2m}\frac{1}{1+t^2}} + (\mathbf{v}^+ - \mathbf{v}^-)\frac{1}{1+t^2} \end{aligned}$$

$$
\begin{aligned}
\boxed{(\mathbf{v}^- \times \mathbf{t}) \times \mathbf{s}} &= (\mathbf{v}^- \times \mathbf{s}) \times \mathbf{t} \\
&= -\left(\frac{q\Delta t}{2m}\right)^2 \frac{1}{1+t^2}[(\mathbf{v}^+ - \mathbf{v}^-) \times \mathbf{B}] \times \mathbf{B} + \boxed{\frac{q\Delta t}{2m}\frac{1}{1+t^2}(\mathbf{v}^+ - \mathbf{v}^-) \times \mathbf{B}} \\
\Longrightarrow \mathbf{v}^+ - \mathbf{v}^- &= (\mathbf{v}^+ - \mathbf{v}^-)\frac{1}{1+t^2} + \left(\frac{q\Delta t}{2m}\right)^2 \frac{1}{1+t^2}\mathbf{B} \times [(\mathbf{v}^+ - \mathbf{v}^-) \times \mathbf{B}]
\end{aligned}
$$

$$(\mathbf{v}^+ - \mathbf{v}^-) \times \mathbf{B} = \frac{q\Delta t}{2m}[(\mathbf{v}^+ + \mathbf{v}^-) \times \mathbf{B}] \times \mathbf{B} =: \mathbf{C} \times \mathbf{B}$$

$$\mathbf{B} \times (\mathbf{C} \times \mathbf{B}) = \mathbf{C}B^2 - \mathbf{B}\cancel{\mathbf{B} \cdot \mathbf{C}} = \frac{q\Delta t}{2m}[(\mathbf{v}^+ + \mathbf{v}^-) \times \mathbf{B}]B^2$$

$$\Longrightarrow t^2(\mathbf{v}^+ - \mathbf{v}^-) = \left(\frac{q\Delta t}{2m}\right)^2 B^2[(\mathbf{v}^+ + \mathbf{v}^-) \times \mathbf{B}]\frac{q\Delta t}{2m}$$

$$t^2 = \left(\frac{q\Delta t}{2m}\right)^2 B^2 \Longrightarrow$$ ✓

PIC

I) $$\frac{x^{n+1/2} - x^{n-1/2}}{\Delta t} = v^n$$

II) $$j^n = \sum v_\alpha^n S(x^*)$$
$$x^* = \left(\frac{x_\alpha^{n+1/2} + x_\alpha^{n-1/2}}{2}\right) = x^n + O(\Delta t^2)$$

III) $$\frac{E^{n+1/2} - E^{n-1/2}}{\Delta t} = \nabla \times B^n - j^n$$

IV) $$\frac{B^{n+1} - B^n}{\Delta t} = -\nabla \times E^{n+1/2}$$

V) Boris

$$\frac{v^{n+1} - v^n}{\Delta t} = \frac{q}{m}\left[E^{n+1/2}(x^{n+1/2}) + \frac{v^{n+1} + v^n}{2} \times B^*(x^{n+1/2})\right]$$
$$B^* = \frac{B^{n+1} + B^n}{2} = B^{n+1/2} + O(\Delta t^2)$$

Vlasov

$$\hat{f}^{n+1}(x^{n+1/2}, v^n) = \Lambda_x(\Delta t)\tilde{f}^n(x^{n-1/2}, v^n)$$

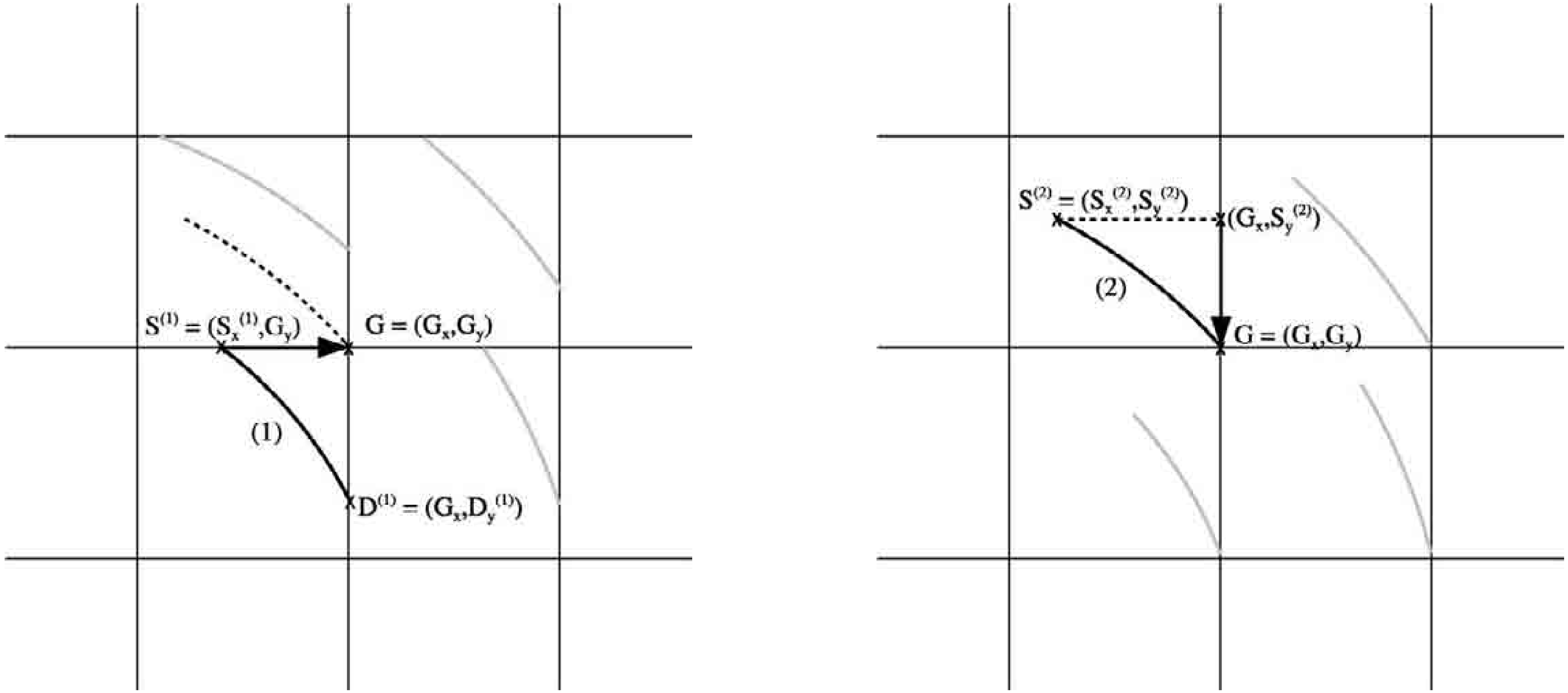
$$j^n = \sum v_\alpha^n f_\alpha^*(x^n, v^n)$$
$$f^*(x^n, v^n) = \left(\frac{\tilde{f}^n(x^{n-1/2}, v^n) + \hat{f}^n(x^{n+1/2}, v^n)}{2}\right)$$
$$= f^n(x^n, n^v) + O(\Delta t^2)$$

Boris

$$\tilde{f}^{n+1}(x^{n+1/2}, v^{n+1}) = \Lambda_v(\Delta t)\hat{f}(x^{n+1/2}, v^n)$$

So let's revisit what the semi-Lagrangian scheme does (for simplicity in 2D).
A full two-dimensional scheme would transport the value of f along the black characteristic.

would like to have: $f^{\mathrm{new}}(D_x, D_y) = f^{\mathrm{old}}(S_x, S_y)$



Splitting: $f^{\mathrm{inter}}(G_x, G_y) = f^{\mathrm{old}}(S_x^{(1)}, G_y)$

f^{old} is lossed, only have f^{inter}

$f^{\mathrm{new}}(G_x, G_y) = f^{\mathrm{inter}}(G_x, S_y^{(2)})$

assuming correct interpolation $f^{\mathrm{inter}}(G_x, S_y^{(2)}) = f^{\mathrm{old}}(S_x^{(2)}, S_y^{(2)})$

$$\Longrightarrow \quad f^{\mathrm{new}}(G_x, G_y) = f^{\mathrm{old}}(S_x^{(2)}, S_y^{(2)}) \quad \checkmark$$

Backsubstitution for the velocity update

The characteristics for the velocity update can be calculated by the *Boris* scheme. Define

$$\mathbf{k} = \frac{\Delta t}{2}\frac{q_s}{m_s}\mathbf{B} \qquad\qquad \mathbf{s} = \frac{2\mathbf{k}}{1+k^2}$$

Now the backward in time *Boris* scheme is given by:

$$\begin{aligned}
\mathbf{v}^{+} &= \mathbf{v}^{n+1} - \frac{\Delta t}{2}\frac{q_s}{m_s}\mathbf{E} \\
\tilde{\mathbf{v}} &= \mathbf{v}^{+} - \mathbf{v}^{+} \times \mathbf{k} \\
\mathbf{v}^{-} &= \mathbf{v}^{+} - \tilde{\mathbf{v}} \times \mathbf{s} \\
\mathbf{v}^{n} &= \mathbf{v}^{-} - \frac{\Delta t}{2}\frac{q_s}{m_s}\mathbf{E}
\end{aligned}$$

This formula has to be brought into this form:

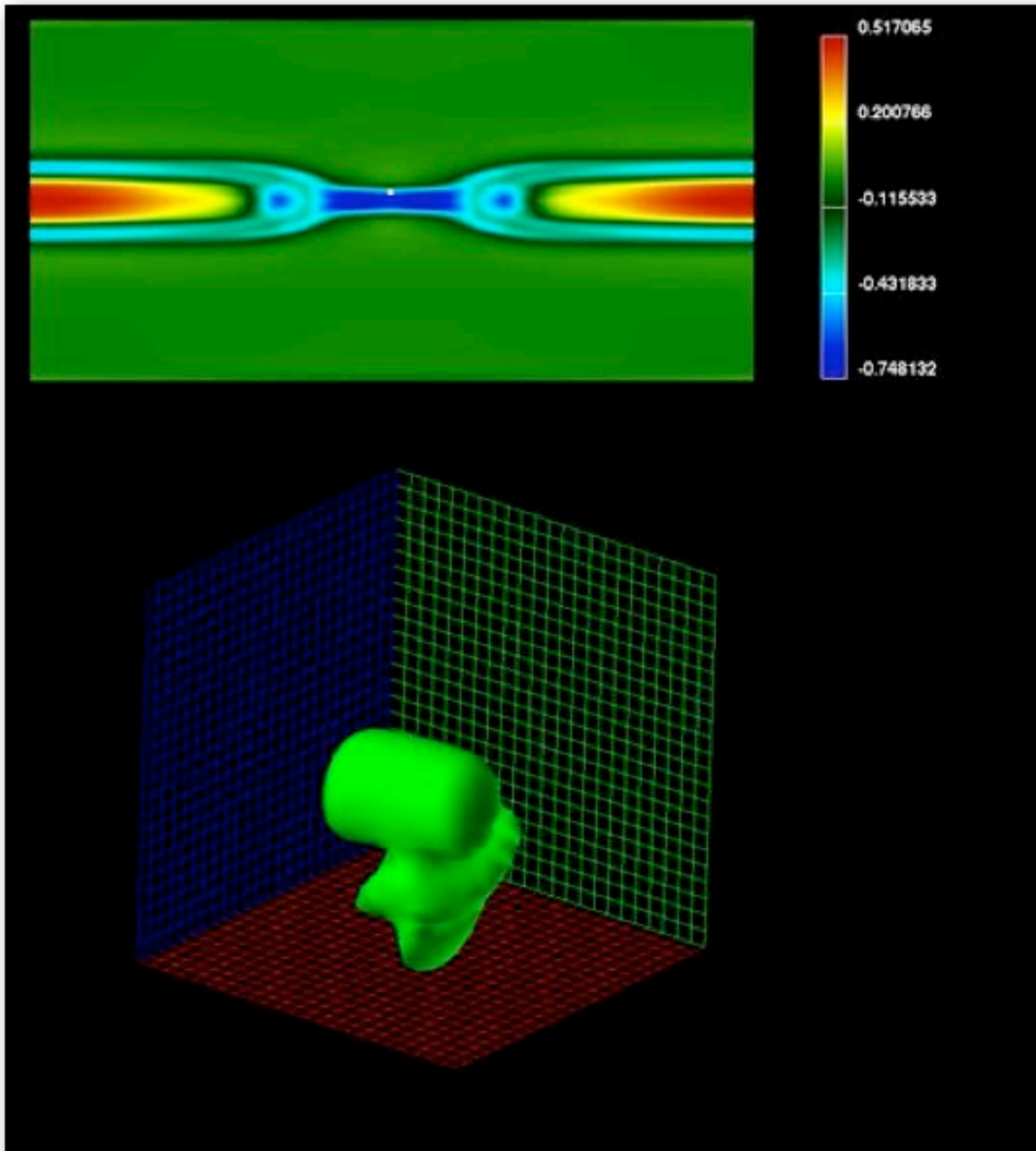
$$\begin{aligned}
v_x^n &= v_x^n(v_x^{n+1}, v_y^n, v_z^n) \\
v_y^n &= v_y^n(v_x^{n+1}, v_y^{n+1}, v_z^n) \\
v_z^n &= v_z^n(v_x^{n+1}, v_y^{n+1}, v_z^{n+1})
\end{aligned}$$

Backsubstitution for the velocity update

$$
\begin{aligned}
v_x^n &= v_x^n(v_x^{n+1}, v_y^n, v_z^n) \\
v_y^n &= v_y^n(v_x^{n+1}, v_y^{n+1}, v_z^n) \\
v_z^n &= v_z^n(v_x^{n+1}, v_y^{n+1}, v_z^{n+1})
\end{aligned}
$$

The last equation (3) is given simply by the z-component of Boris' scheme.
To find (2) we solve (3) for v_z^{n+1} and substitute this into the y-component of Boris' scheme.
Equation (1) can be found by using the x-component of the *forward* in time Boris scheme and solving for v_x^n.

Example: magnetic reconnection with DSDV I



Electron out of plane current

Electron distribution function

New Code: DSDV II (Martin Rieke)

- full Maxwell Solver
- parallel CUDA

Hardware and CUDA performance

The *DaVinci-cluster* at the Ruhr-Universität Bochum consists of 17 nodes with a total of

- 16320 cores and 272 GB RAM on GPUs (68~NVidia Tesla S1070 cards with 240 cores and 4 GB RAM each)
- 136 respectively 272 (with HT) cores and 408 GB on CPUs (34 Xeon E5530 Quad Core CPUs (2.4 GHz) with 8 cores respectively 16 cores (with HT) and 12~GB RAM each)

system	resolution	duration of run
CPUs (Schmitz, Grauer)	$256 \times 128 \times 30^3$	~ 150 h
GPUs (this work)	$256 \times 128 \times 32^3$	~ 8 h

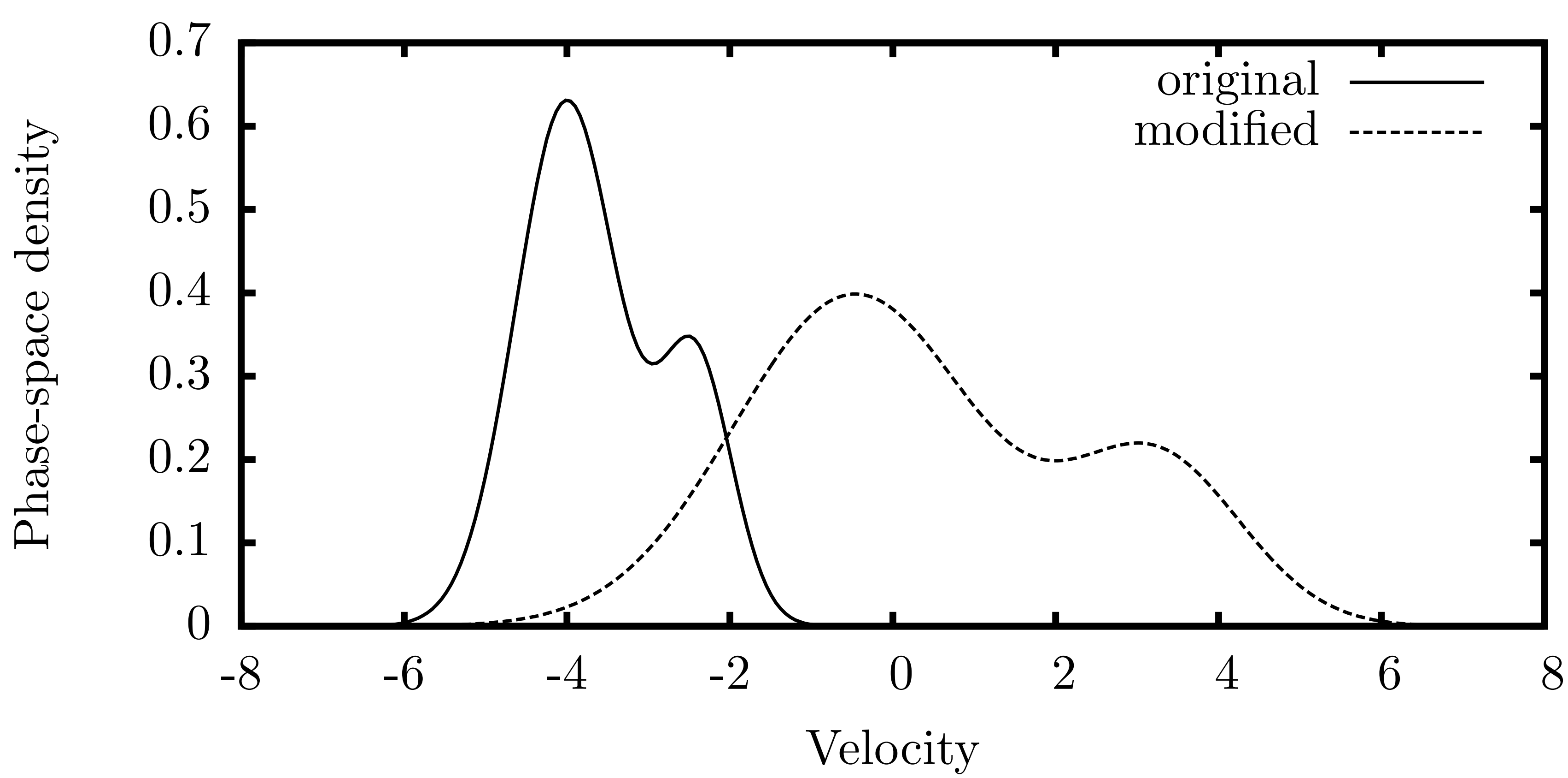
Comparison of the time necessary to simulate one quarter of the GEM setup until $t = 40\Omega_i^{-1}$.

Ok, now we have a Vlasov code !

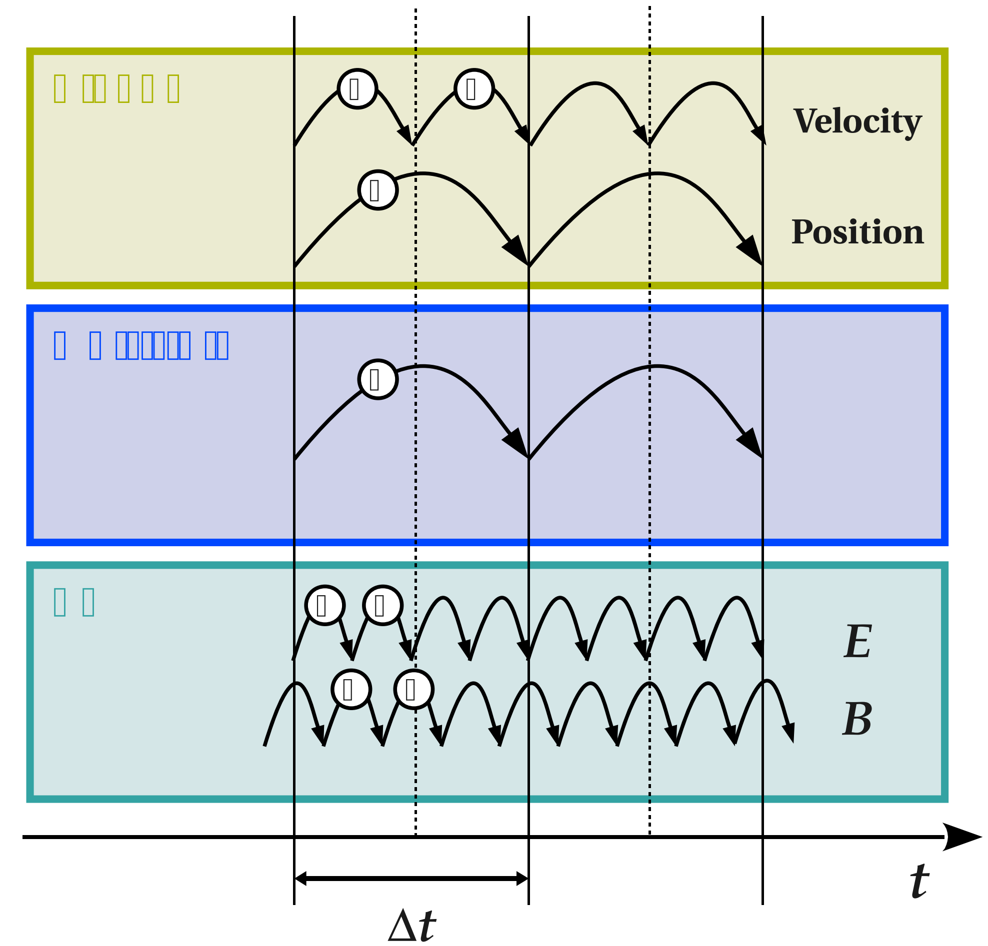
Let’s do the coupling

Multifluid and Vlasov blocks communicate via exchange of ghostcells.

- In a first step, the phase-space density is extrapolated into the ghostcells.
  This is of coarse not correct but respects phase space structure.
- Next, it is modified to match the moments given by the fluid in the respective cell by rescaling, translating, and squeezing.
  This is implemented as advection along suitably chosen characteristics.



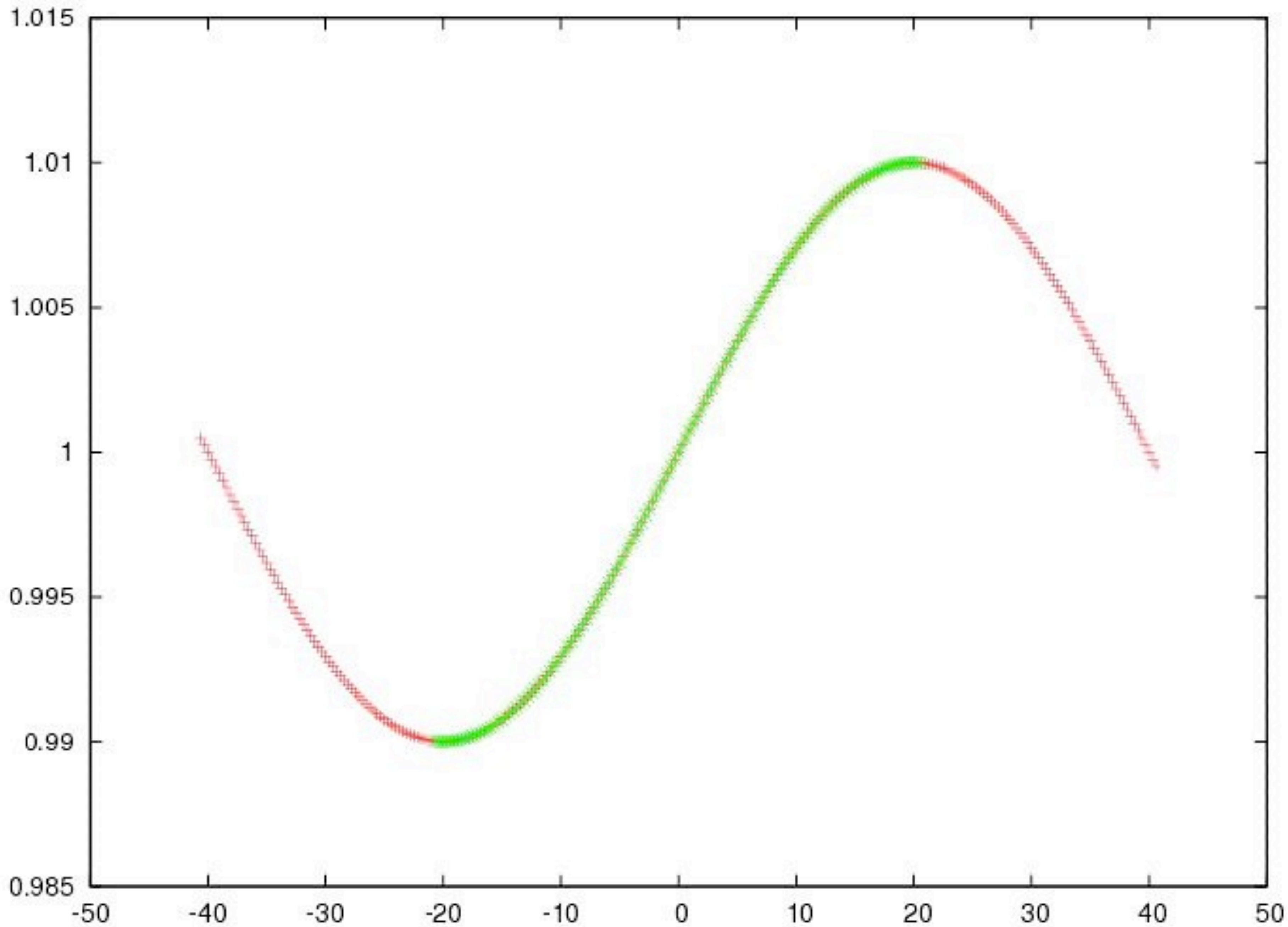
- The multifluid ghostcells are filled with the moments calculated from the phase-space density of the Vlasov simulation.
Because the RK scheme is a multi-stage method, these moments are interpolated linearly in time.



- Both codes calculate the current density in their respective regions. These are collected and used to integrate Maxwell's equations globally.

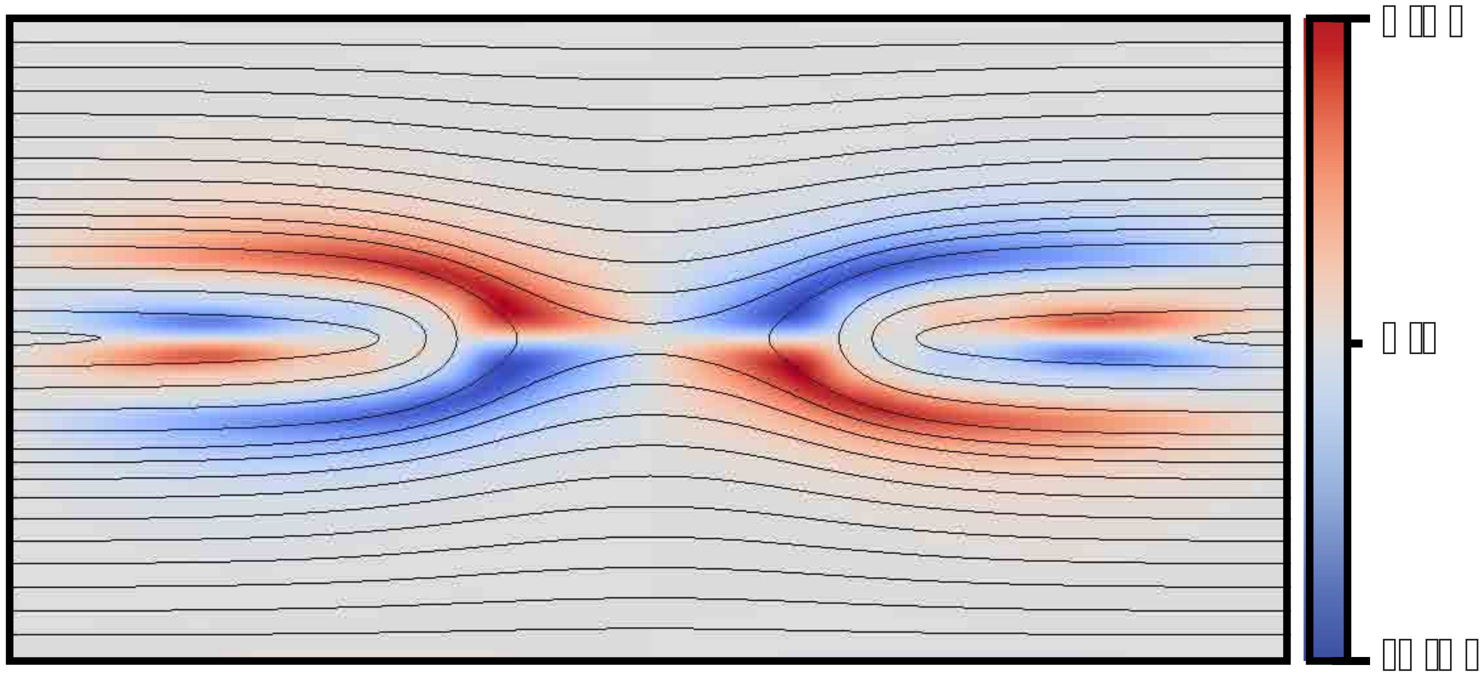
- Each code can be executed on its own, or coupled to the other via MPI. This concept is known as *Multiple Program Multiple Data* (MPMD).

Ion Sound Waves

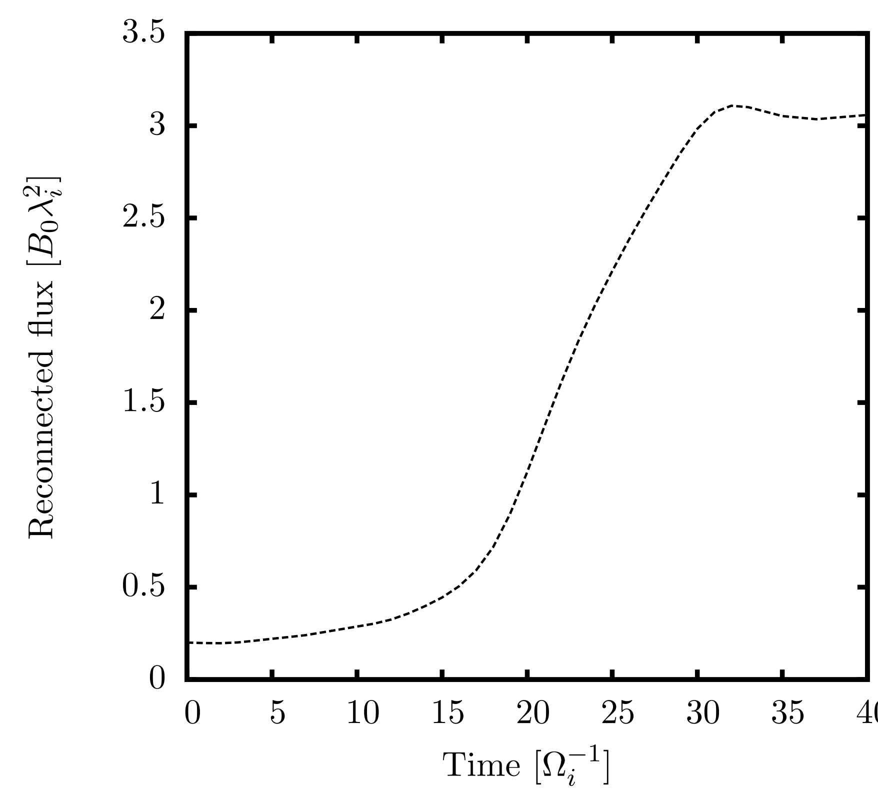


Results/Examples

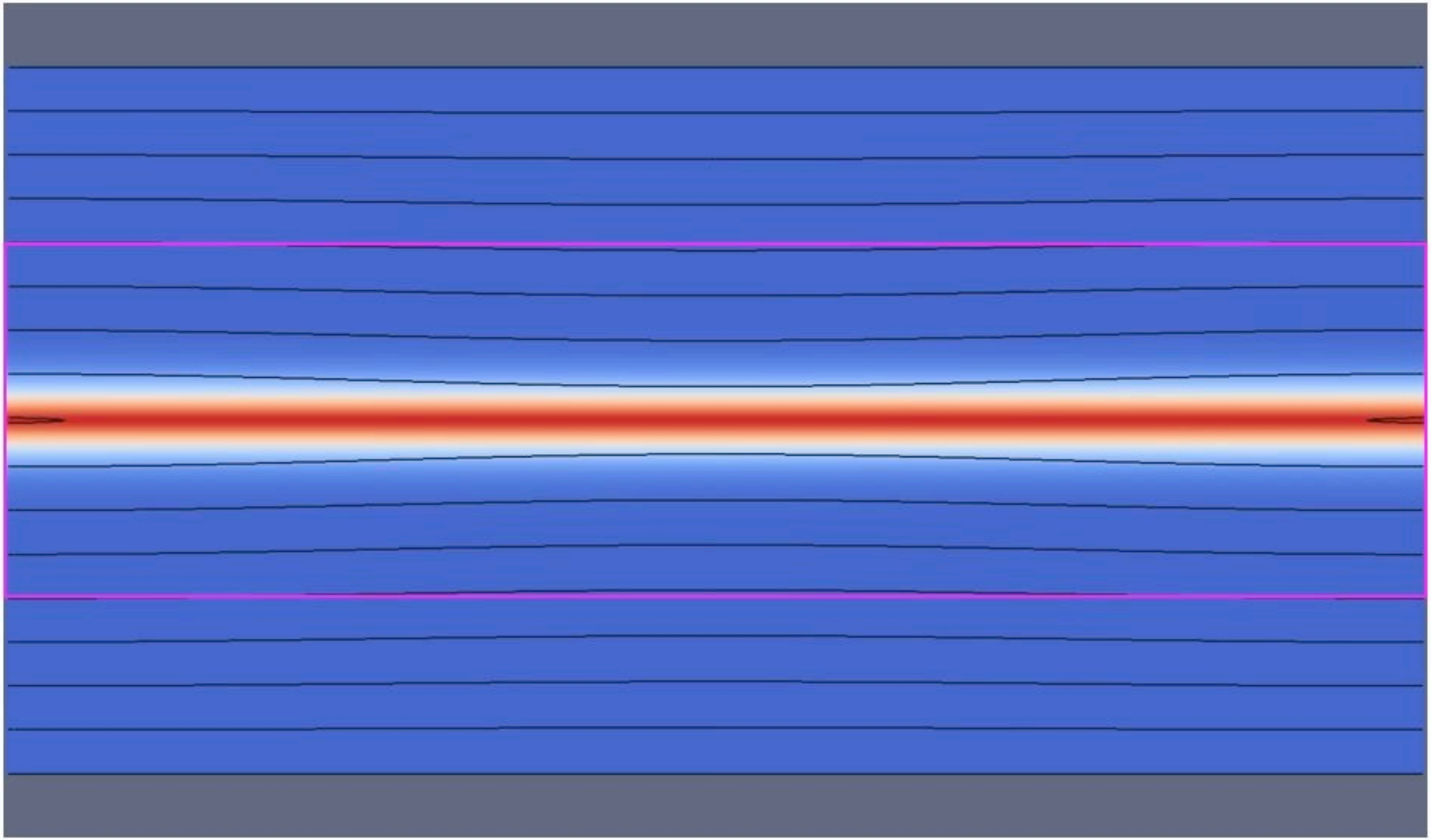
The GEM reconnection challenge (2001), where there is a clearly localized area of interest at the current sheet, was simulated as a test case.



B_z together with magnetic field lines at time of peak reconnection rate.



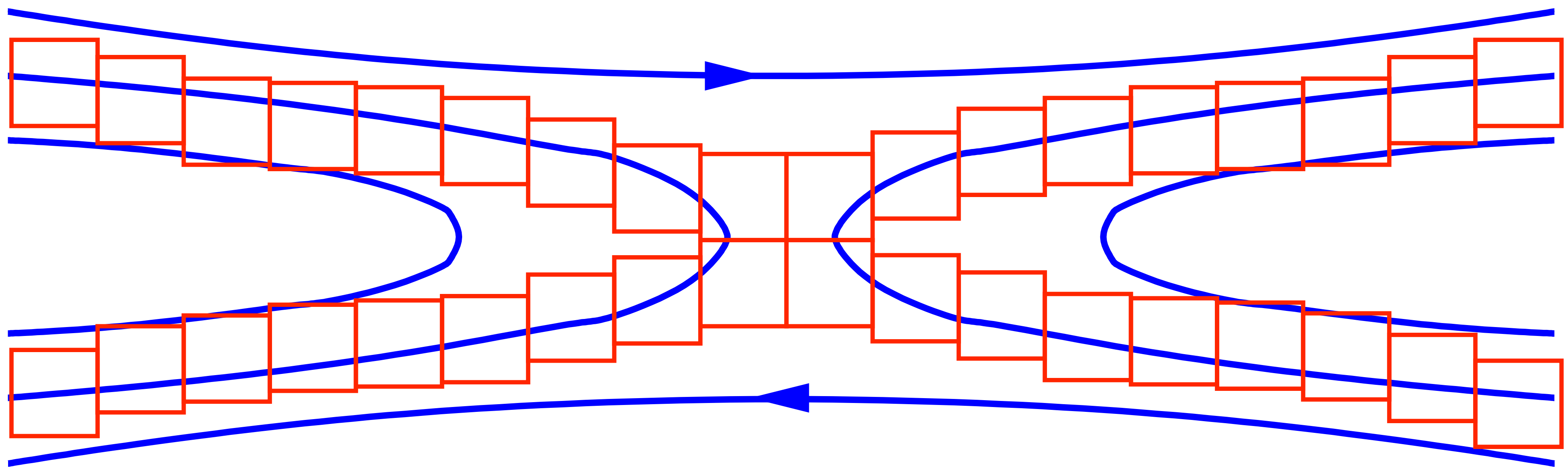
Reconnected magnetic flux over time.



Where is fluid and where is the kinetic region ?

Future dreams:

- adaptive fluid-kinetic coupling



indicator: difference between 5- and 10-moment model

- multiscale-multiphysics
  MHD -> Hall-MHD (Ohms law) -> 5 moment 2 fluid -> kinetic

Thank You