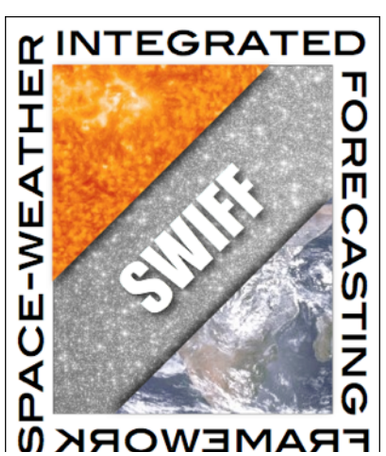
The Multi Level Multi Domain (MLMD) method:

a semi-implicit adaptive method for Particle In Cell plasma simulations

M.E. Innocenti[1], A. Beck[2], S. Markidis[3], G. Lapenta[1,4]

1: Centre for Plasma Astrophysics, KULeuven, Belgium
2: Laboratoire Leprince-Ringuet, Ecole Polytechnique, CNRS-IN2P3, France
3: HPCViz Department, KTH Royal Institute of Technology, Stockholm, Sweden
4: Exascience Intel Lab Europe, Kapeldreef 75, B-3001 Leuven, Belgium

Efficient solution of large systems of non-linear PDEs in science - Lyon
October 8, 2013



Outline

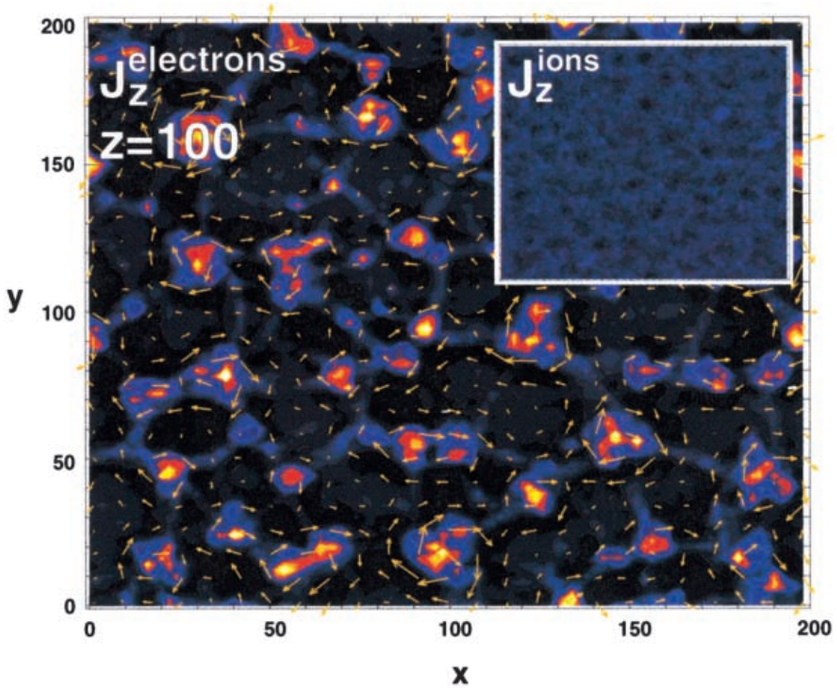
- WHAT do we want to simulate?
- HOW do we want to simulate it?
- WHY do we need a (Multi Level Multi Domain) implicit code?
- the Multi Level Multi Domain method and Parsek2D-MLMD, a parallel C++ implicit adaptive code
- conclusions and future work

WHAT do we want to simulate?

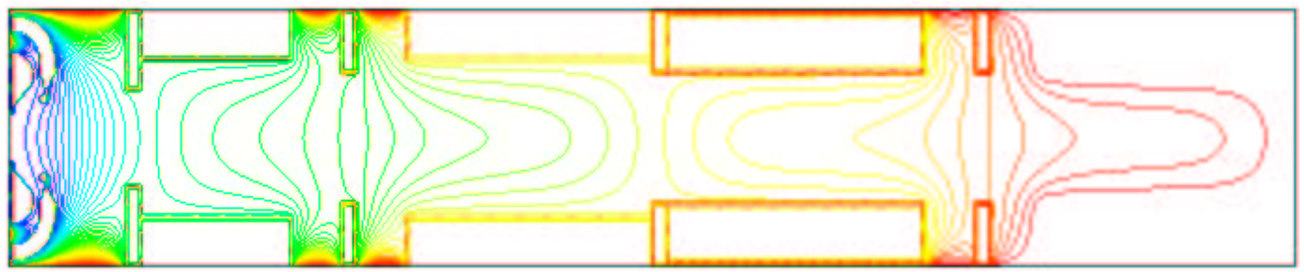
Plasmas,

in different fields of applications

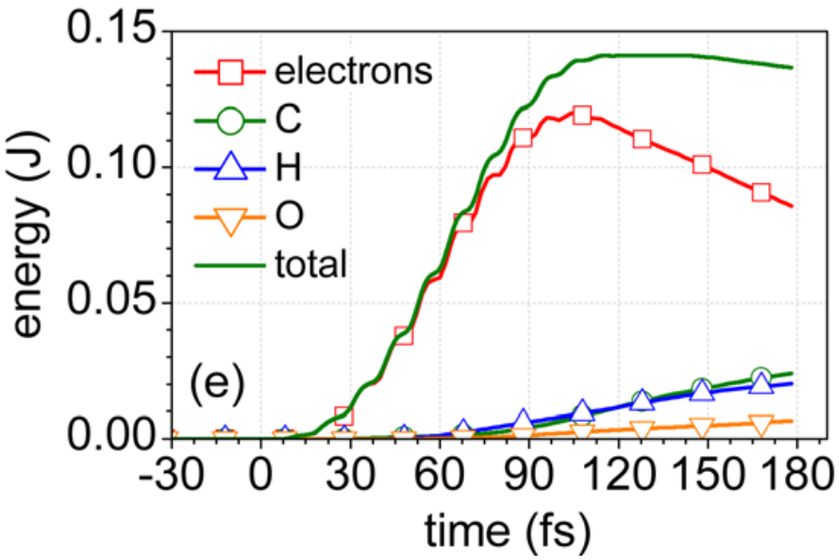
Relativistic shocks



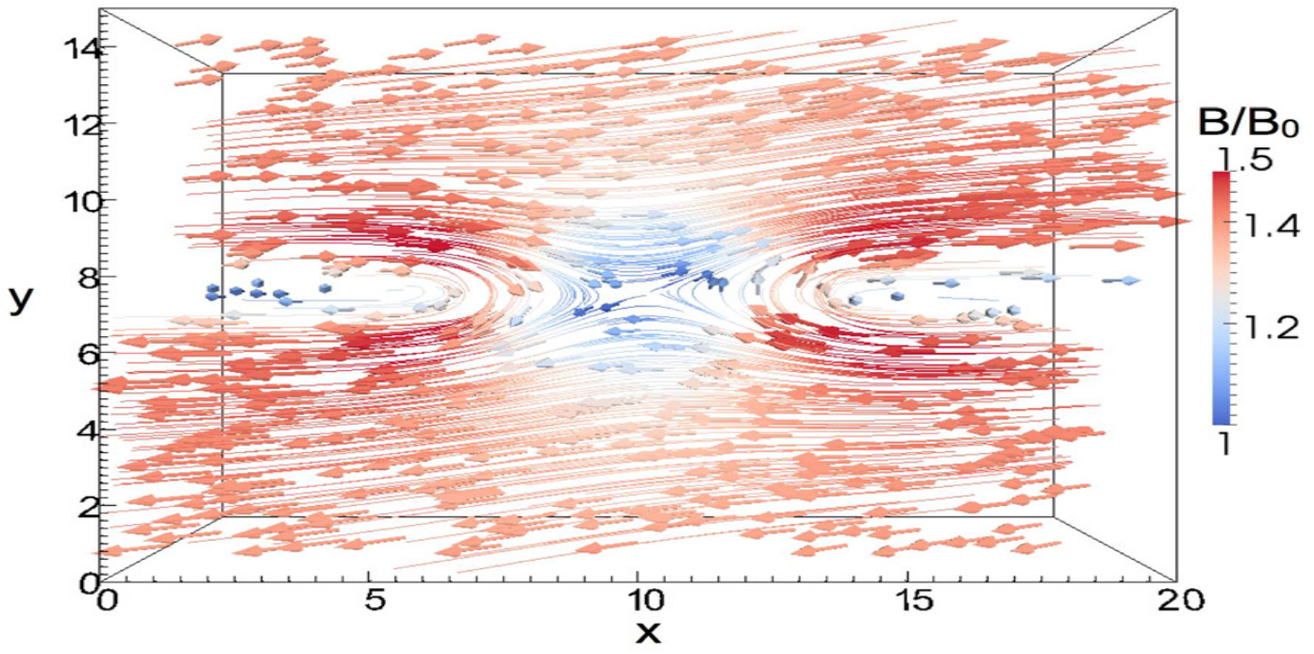
Accelerators



Laser-target interaction



Magnetic reconnection



Frederiksen04, McCorquondale04, Petrov11, Markidis12

HOW do we want to simulate it?

Different physical descriptions

increasing physical details...

fluid

the plasma must be:
- quasi neutral
- with negligible particle scale processes
- in thermodynamic equilibrium

hybrid
(ions as particles, electrons as fluid)

the plasma must be:
- quasi neutral
- with negligible electron scale processes

accessible:
- ion gyroradius and ion skin depth processes

full particle
(ions and electrons as particles)

accessible:
- all particle processes

... at an increasing computational cost

Ledvina08

Where does the computational cost come from?

Particle In Cell simulations: two sets of equations to solve

$$\nabla \cdot \mathbf{E} = 4\pi\rho$$
$$\nabla \cdot \mathbf{B} = 0$$
$$\nabla \times \mathbf{E} = -\frac{1}{c}\frac{\partial \mathbf{B}}{\partial t}$$
$$\nabla \times \mathbf{B} = \frac{4\pi}{c}\mathbf{J} + \frac{1}{c}\frac{\partial \mathbf{E}}{\partial t}$$

Solved at grid points g

$$\frac{d\mathbf{x}_p}{dt} = \mathbf{v}_p$$
$$\frac{d\mathbf{v}_p}{dt} = \frac{q_p}{m_p}\left(\mathbf{E}_p + \frac{\mathbf{v}_p \times \mathbf{B}_p}{c}\right)$$

Solved for each particle

Interpolation function $W(\mathbf{x}_g\text{-}\mathbf{x}_p)$ to exchange info between particles and grid points and vice versa

With explicit time discretization, three stability constraints:

$c\Delta t < \Delta x$ Courant condition on the light wave

$\omega_{pe}\Delta t < 2$ to properly model particle response

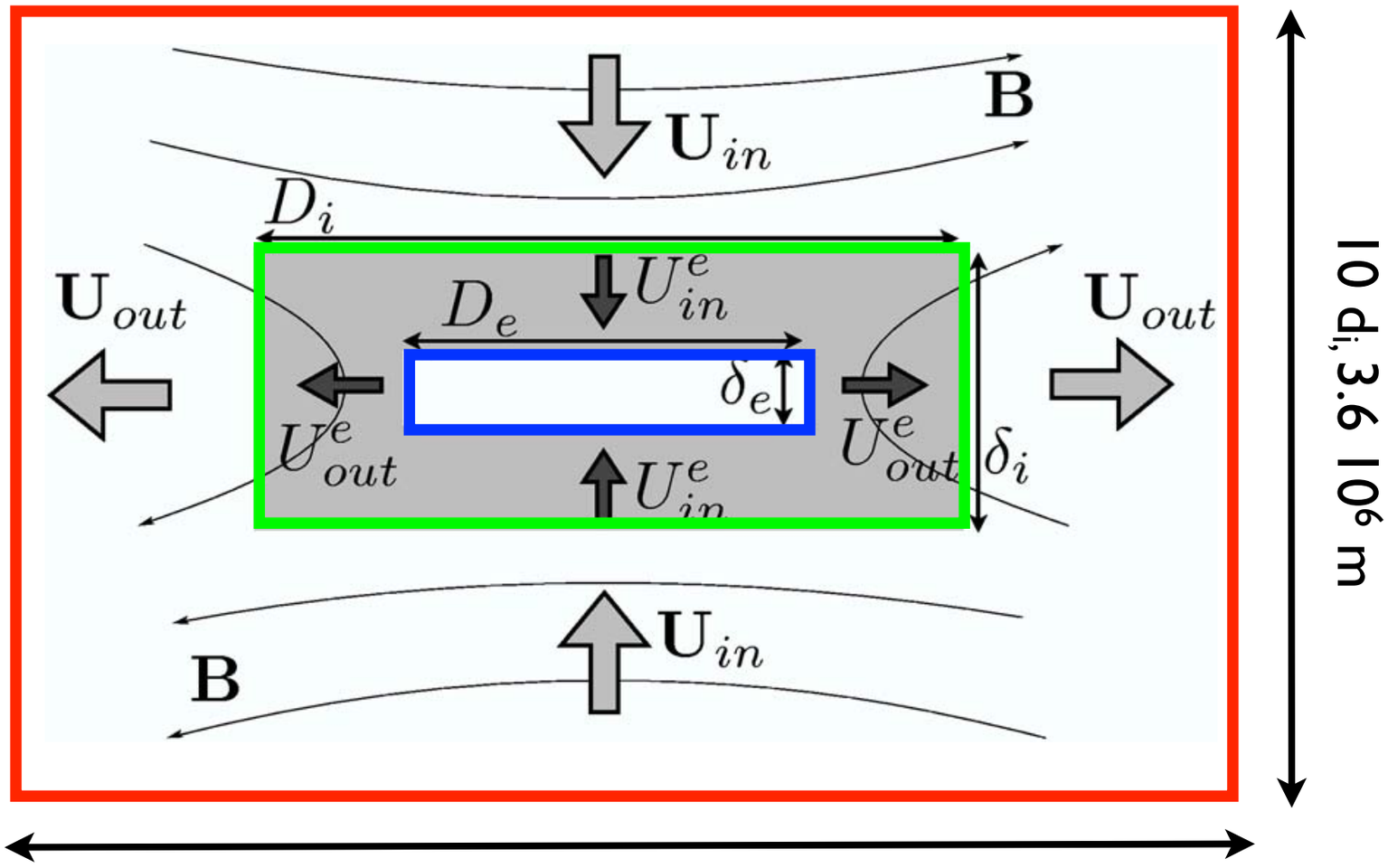
$\Delta x < \varsigma\lambda_D$ to avoid numerical grid instabilities

ω_{pe} electron plasma frequency

λ_D Debye length

Birdsall04, Hockney88

What does this imply for a magnetic reconnection simulation?



explicit PIC:

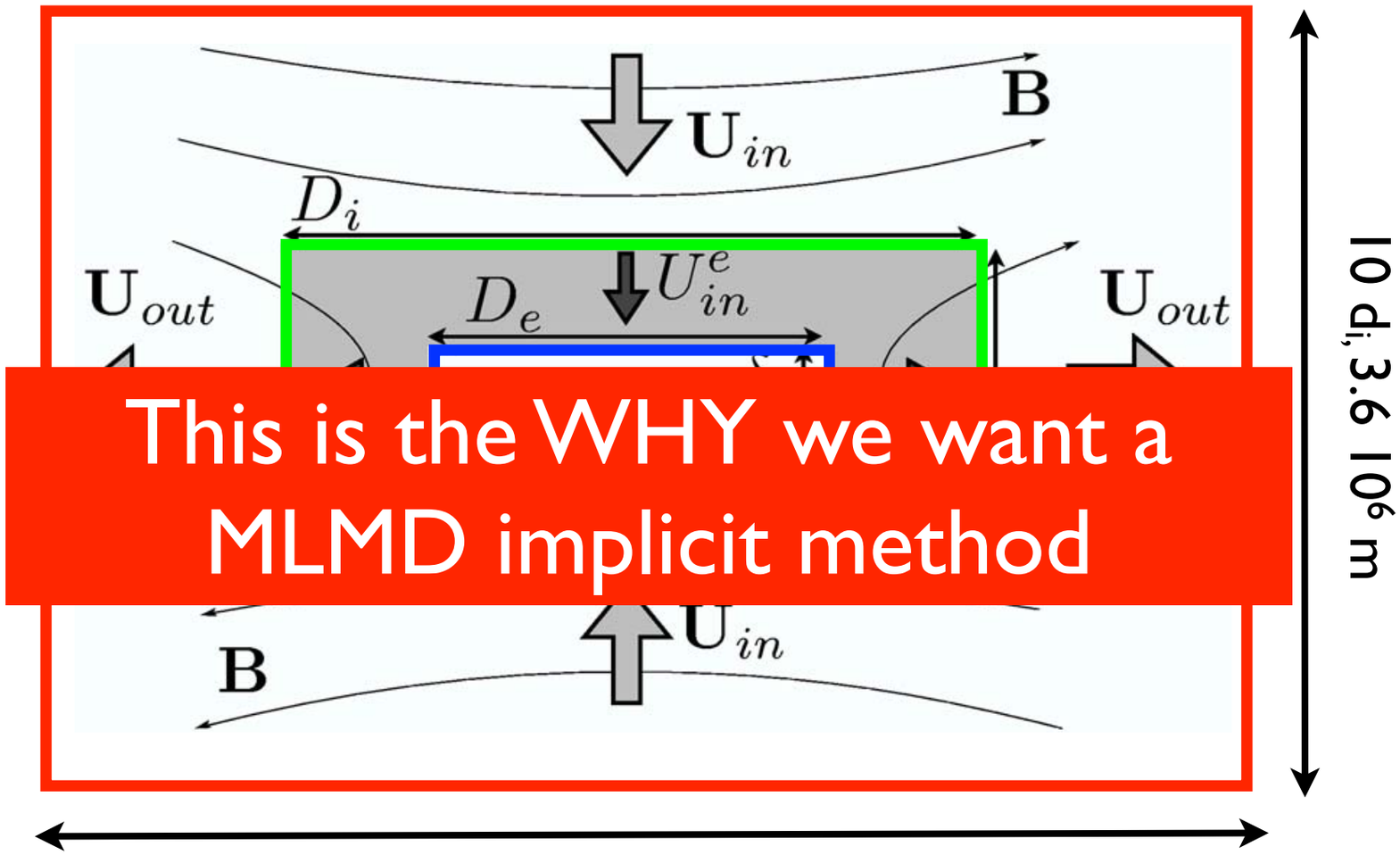
$\delta = \lambda_{D,e} = 217$ m

$\Delta t = 0.1/\omega_{pe} = 10^{-4}$ s

assuming that a core simulates 100 cycle for 4096 cells in a min, the simulation of 1 physical second takes 100 mins on 135 000 cores

Daughton06

What does this imply for a magnetic reconnection simulation?



This is the WHY we want a MLMD implicit method

explicit PIC:

$\delta = \lambda_{D,e} = 217$ m

$\Delta t = 0.1/\omega_{pe} = 10^{-4}$ s

assuming that a core simulates 100 cycle for 4096 cells in a min, the simulation of 1 physical second takes 100 mins on 135 000 cores

Daughton06

How to bypass this stability constraints?

(Semi) implicit methods,
divided in three big families

Fully implicit methods

Chen et al., 2011
Markidis et Lapenta, 2011

Direct implicit methods

Cohen et al., 1989
Friedman, 1990
Hewett and Langdon, 1987
Tanaka, 1988
Gibbons and Hewitt, 1995

Implicit Moment Method (IMM)

Mason, 1987

Vu and Brackbill, 1992
Lapenta et al., 2006

How to bypass this stability constraints?

(Semi) implicit methods,
divided in three big families

Fully implicit methods

Chen et al., 2011
Markidis et Lapenta, 2011

Direct implicit methods

Cohen et al., 1989
Friedman, 1990
Hewett and Langdon, 1987
Tanaka, 1988
Gibbons and Hewitt, 1995

They all share a more lenient stability constraint calculated on the average particle velocity

For the IMM:
$0.01 < v_{th,e}\Delta t/\Delta x < 1$

Implicit Moment Method (IMM)

Mason, 1987

Vu and Brackbill, 1992
Lapenta et al., 2006

The Implicit Moment Method (IMM)

discretization of Maxwell's equations in terms of the decentering parameters θ

$$\mathbf{E}^{n+\theta} - (c\theta\Delta t)^2 \nabla^2 \mathbf{E}^{n+\theta} =$$

$$\mathbf{E}^n + c\theta\Delta t \left(\nabla \times \mathbf{B}^n - \frac{4\pi}{c}\mathbf{J}^{n+\theta}\right) - (c\theta\Delta t)^2 4\pi\nabla\rho^{n+\theta}$$

Crank-Nicholson Predictor Corrector scheme to time-discretize particle motion

$$\mathbf{x}_p^{n+1} = \mathbf{x}_p^n + \bar{\mathbf{v}}_p^k \Delta t$$

$$\mathbf{v}_p^{n+1} = \mathbf{v}_p^n + \frac{q_p \Delta t}{m_p}\left(\mathbf{E}_p^{n+\theta}(\bar{\mathbf{x}}^k) + \frac{\bar{\mathbf{v}}_p^k \times \mathbf{B}_p^n(\bar{\mathbf{x}}^k)}{c}\right)$$

the equation coupling is solved by approximating particle moments around the particle positions at the previous time step

$$\rho^{n+\theta} = \sum_p q_p S(\mathbf{x} - \mathbf{x}^{n+\theta}) = \sum_p q_p \left[S(\mathbf{x} - \mathbf{x}^n) - (\mathbf{x}^{n+\theta} - \mathbf{x}^n)\nabla S(\mathbf{x} - \mathbf{x}^n) + \frac{1}{2}(\mathbf{x}^{n+\theta} - \mathbf{x}^n)^2 \nabla\nabla S(\mathbf{x} - \mathbf{x}^n) + \mathcal{O}(\mathbf{x} - \mathbf{x}^n)^3\right]$$

$$(\mathbf{x}^{n+\theta} - \mathbf{x}^n) = f(\mathbf{E}^{n+\theta})$$

final expression for the time-advanced field

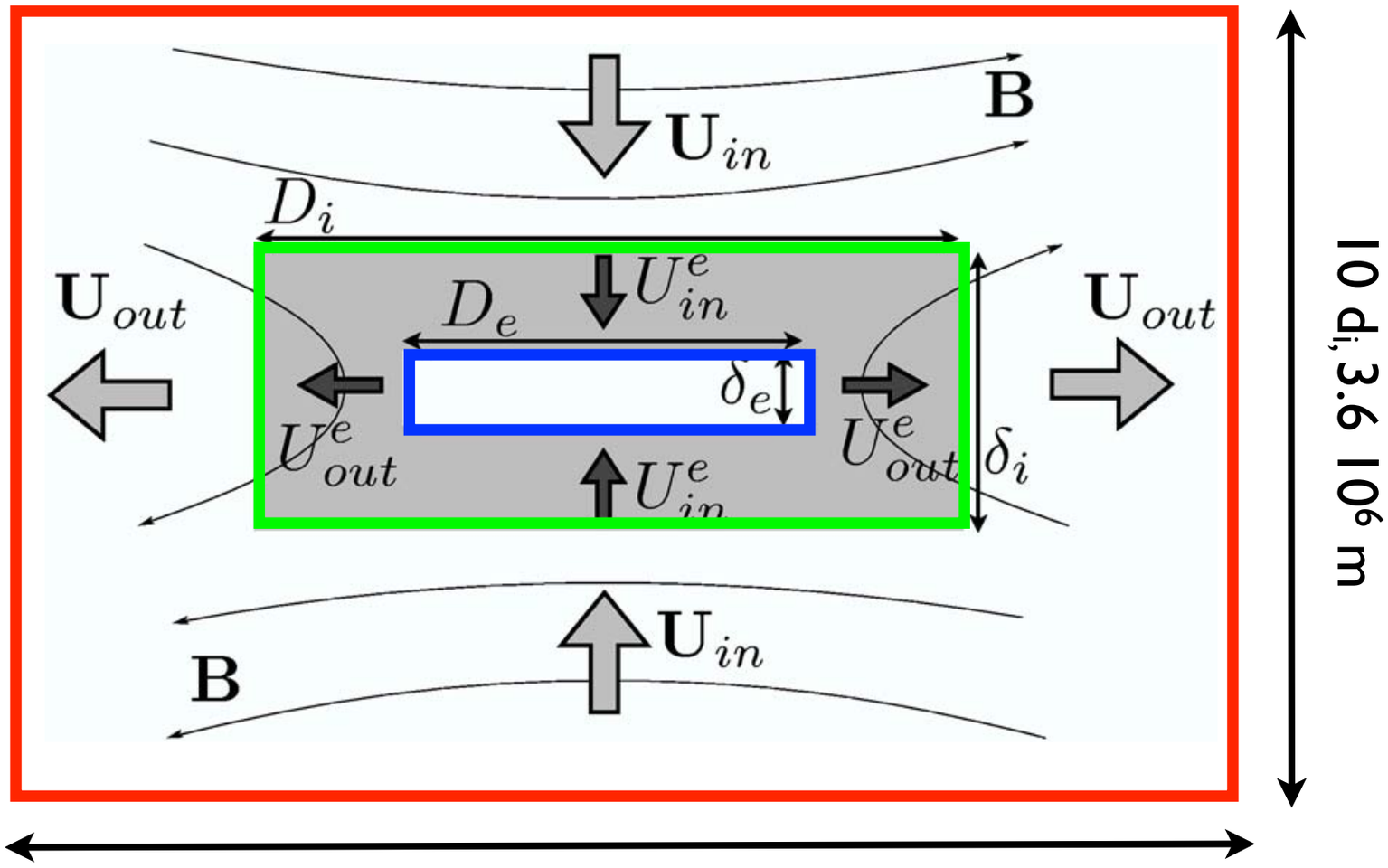
$$(\mathbf{I} + \mu) \cdot \mathbf{E}^{n+\theta} - (c\theta\Delta t)^2 \left(\nabla^2 \mathbf{E}^{n+\theta} + \nabla\nabla \cdot (\mu \cdot \mathbf{E}^{n+\theta})\right) = \mathbf{E}^n + c\theta\Delta t \left(\nabla \times \mathbf{B}^n - \frac{4\pi}{c}\hat{\mathbf{J}}^n\right) - (c\theta\Delta t)^2 4\pi\nabla\hat{\rho}^n$$

solved with a matrix free GMRES

Single level simulations DO NOT need preconditioning, MLMD simulations DO

Saad03

What about magnetic reconnection simulations NOW?



IMM PIC:

$\delta = d_e/2 = 3.76\ 10^3$ m

$\Delta t = 0.1/\omega_{pi} = 5\ 10^{-3}$ s

assuming that a core simulates 100 cycle for 4096 cells in a min, the simulation of 1 physical second takes 2 mins on 451 cores

Can we do even better?

Adaptivity:

if different resolution is needed in different parts of the domain, locally adapt the grid resolution to the physics of interest to save resources in the areas where high resolution is not needed

Moving Mesh Adaptation

Brackbill, 1993
Lapenta, 2011
Chacon et al., 2011

Adaptive Mesh Refinement

Vay et al., 2004
Fujimoto et Sydora, 2008

Multi Level Multi Domain (MLMD)

Innocenti et al., 2013
Beck et al., submitted

Can we do even better?

Adaptivity:

if different resolution is needed in different parts of the domain, locally adapt the grid resolution to the physics of interest to save resources in the areas where high resolution is not needed

Moving Mesh Adaptation

Brackbill, 1993
Lapenta, 2011
Chacon et al., 2011

Adaptive Mesh Refinement

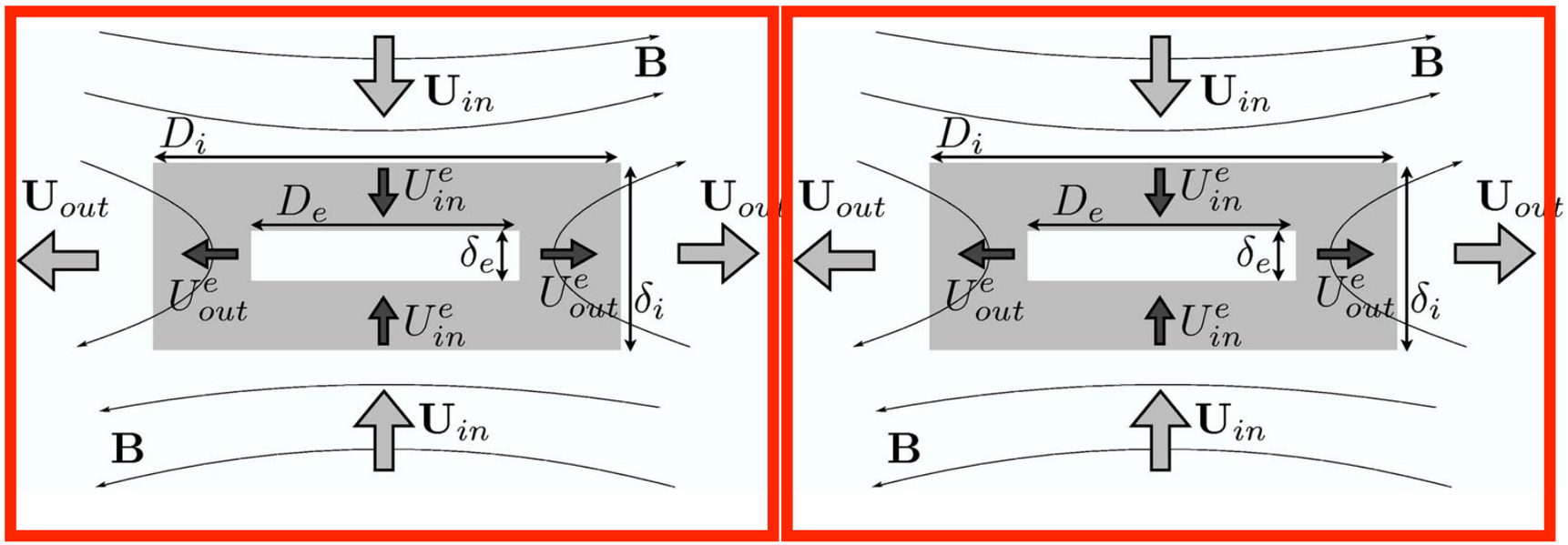
Vay et al., 2004
Fujimoto et Sydora, 2008

we propose a semi-implicit adaptive method
for Particle In Cell plasma simulations
→combines the benefits of implicit and adaptive methods

Multi Level Multi Domain (MLMD)

Innocenti et al., 2013
Beck et al., submitted

What can we do for magnetic reconnection problems with the MLMD method?



Outer Region
20 d_i x 10 d_i
7.2 10^6 m x 3.6 10^6 m

Ion Diffusion Region
5 d_i x 2 d_i
1.8 10^6 m x 7.2 10^5 m

Electron Diffusion Region
5 d_e x 2 d_e
3.76 10^4 m x 7.52 10^3

explicit PIC

$\delta = \lambda_{D,e} = 217$ m
$\Delta t = 0.1/\omega_{pe} = 10^{-4}$ s

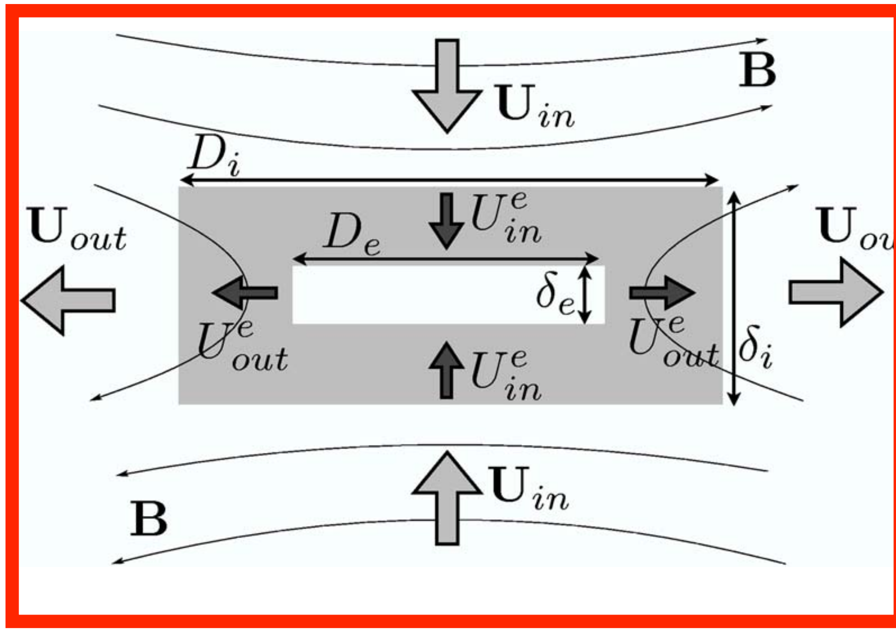
the simulation of 1 physical second takes
100 mins on 135 000 cores

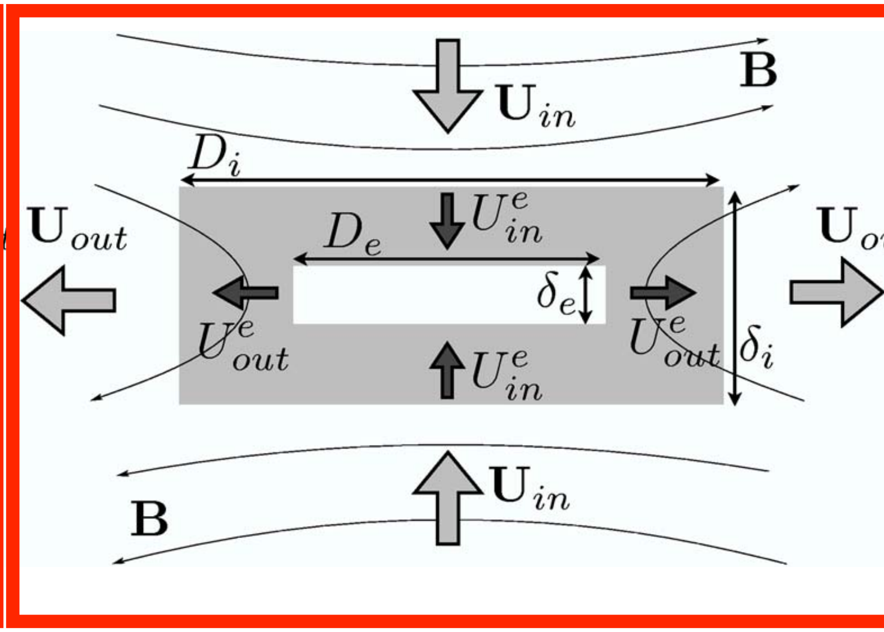
IMM PIC

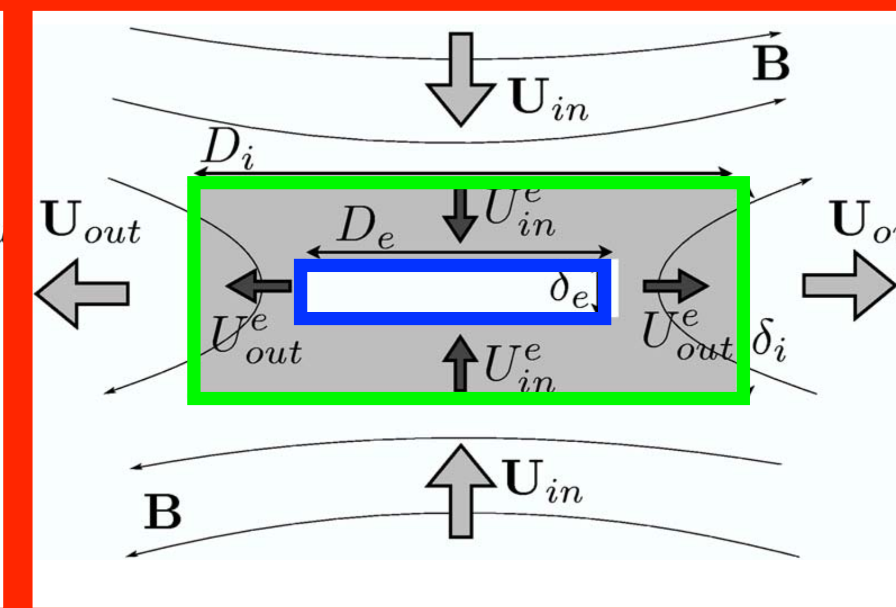
$\delta = d_e/2 = 3.76\ 10^3$ m
$\Delta t = 0.1/\omega_{pi} = 5\ 10^{-3}$ s

the simulation of 1 physical second takes
2 mins on 451 cores

What can we do for magnetic reconnection problems with the MLMD method?







Outer Region

$20\ d_i \times 10\ d_i$

$7.2\ 10^6$ m x $3.6\ 10^6$ m

explicit PIC

$\delta = \lambda_{D,e} = 217$ m

$\Delta t = 0.1/\omega_{pe} = 10^{-4}$ s

the simulation of 1 physical second takes 100 mins on 135 000 cores

Ion Diffusion Region

$5\ d_i \times 2\ d_i$

$1.8\ 10^6$ m x $7.2\ 10^5$ m

IMM PIC

$\delta = d_e/2 = 3.76\ 10^3$ m

$\Delta t = 0.1/\omega_{pi} = 5\ 10^{-3}$ s

the simulation of 1 physical second takes 2 mins on 451 cores

Electron Diffusion Region

$5\ d_e \times 2\ d_e$

$3.76\ 10^4$ m x $7.52\ 10^3$

MLMD IMM PIC

$\delta = d_e/10 = 7.52\ 10^2$ m

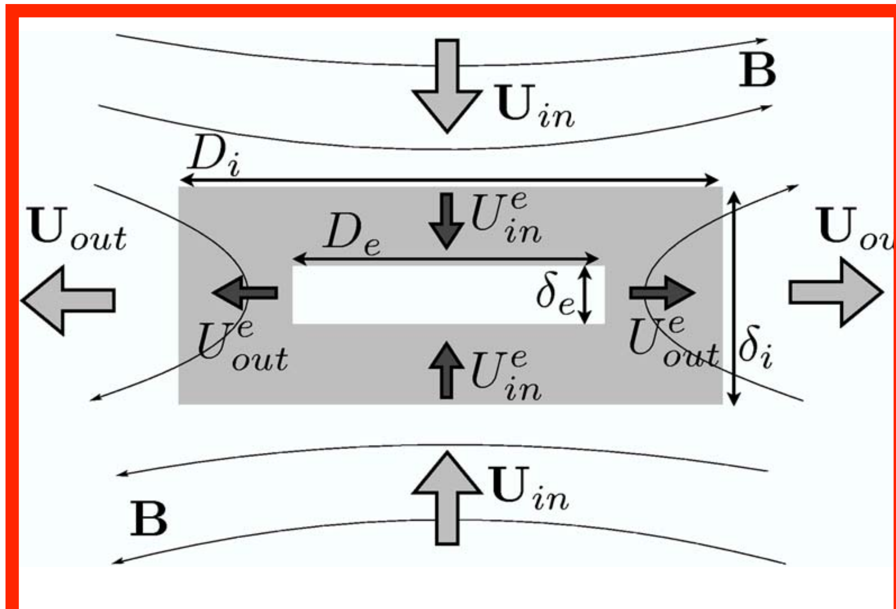
$\delta = d_i/10 = 3.6\ 10^4$ m

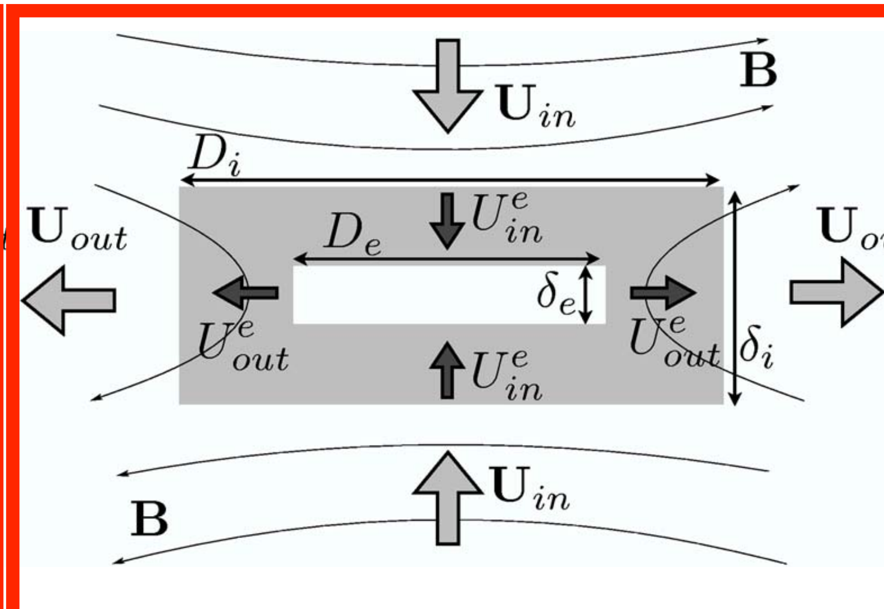
$\delta = d_i/2 = 1.8\ 10^5$ m

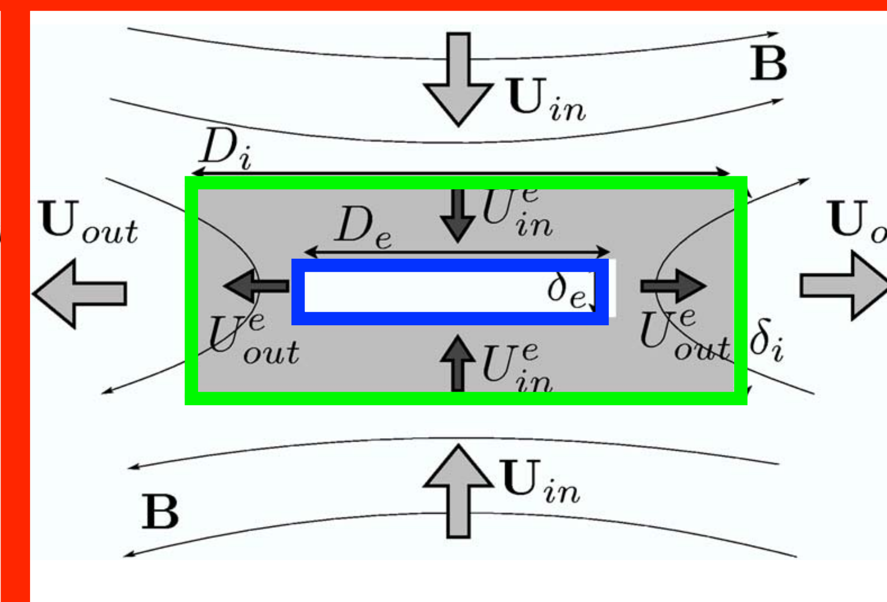
$\Delta t = 0.1/\omega_{pi} = 5\ 10^{-3}$ s

the simulation of 1 physical second takes 2 mins on 1 core

What can we do for magnetic reconnection problems with the MLMD method?







Outer Region

20 d_i x 10 d_i

7.2 10^6 m x 3.6 10^6 m

explicit PIC

$\delta = \lambda_{D,e}$ = 217 m

$\Delta t = 0.1/\omega_{pe} = 10^{-4}$ s

the simulation of 1 physical second takes 100 mins on 135 000 cores

Ion Diffusion Region

5 d_i x 2 d_i

1.8 10^6 m x 7.2 10^5 m

IMM PIC

$\delta = d_e/2$ = 3.76 10^3 m

$\Delta t = 0.1/\omega_{pi} = 5\ 10^{-3}$ s

the simulation of 1 physical second takes 2 mins on 451 cores

Electron Diffusion Region

5 d_e x 2 d_e

3.76 10^4 m x 7.52 10^3

MLMD IMM PIC

$\delta = d_e/10$ = 7.52 10^2 m

$\delta = d_i/10$ = 3.6 10^4 m

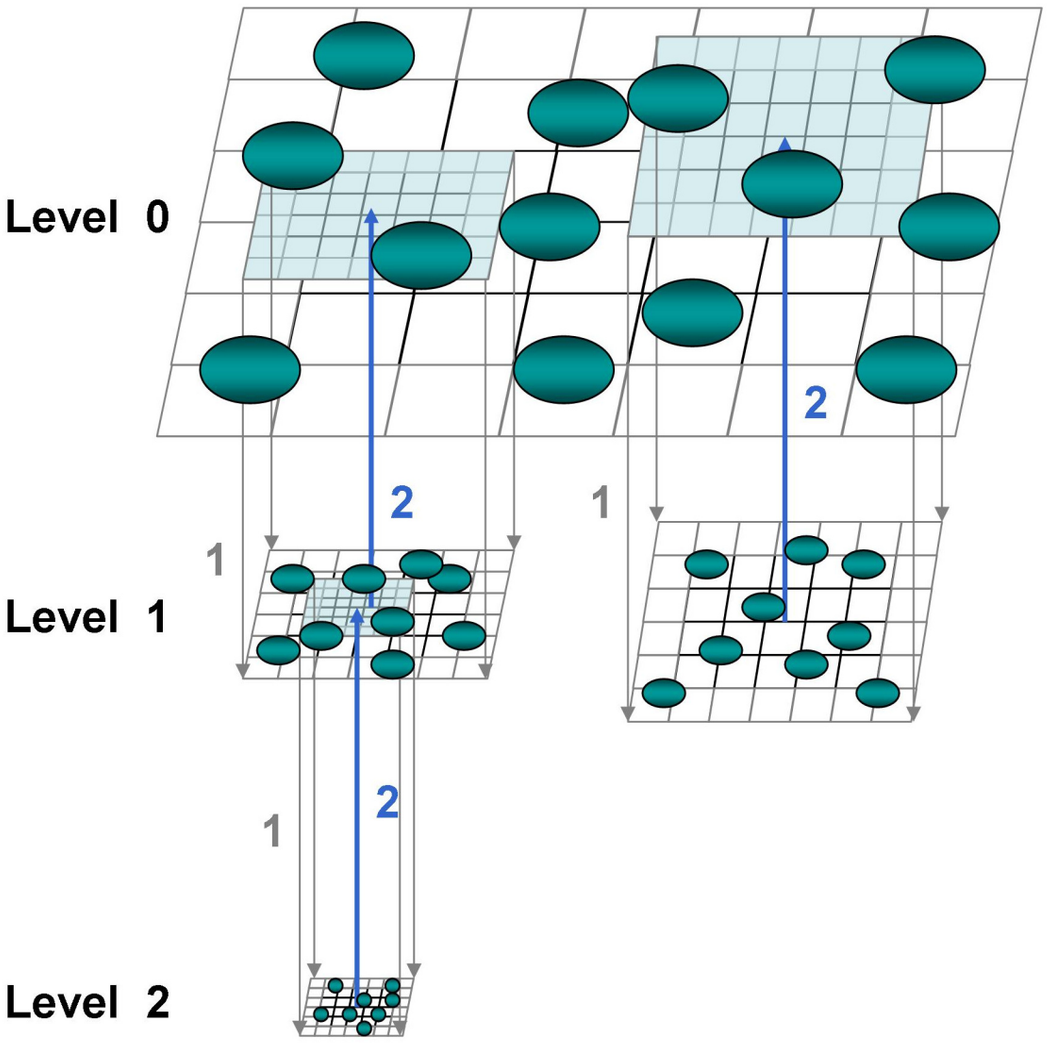
$\delta = d_i/2$ = 1.8 10^5 m

$\Delta t = 0.1/\omega_{pi} = 5\ 10^{-3}$ s

the simulation of 1 physical second takes 2 mins on 1 core

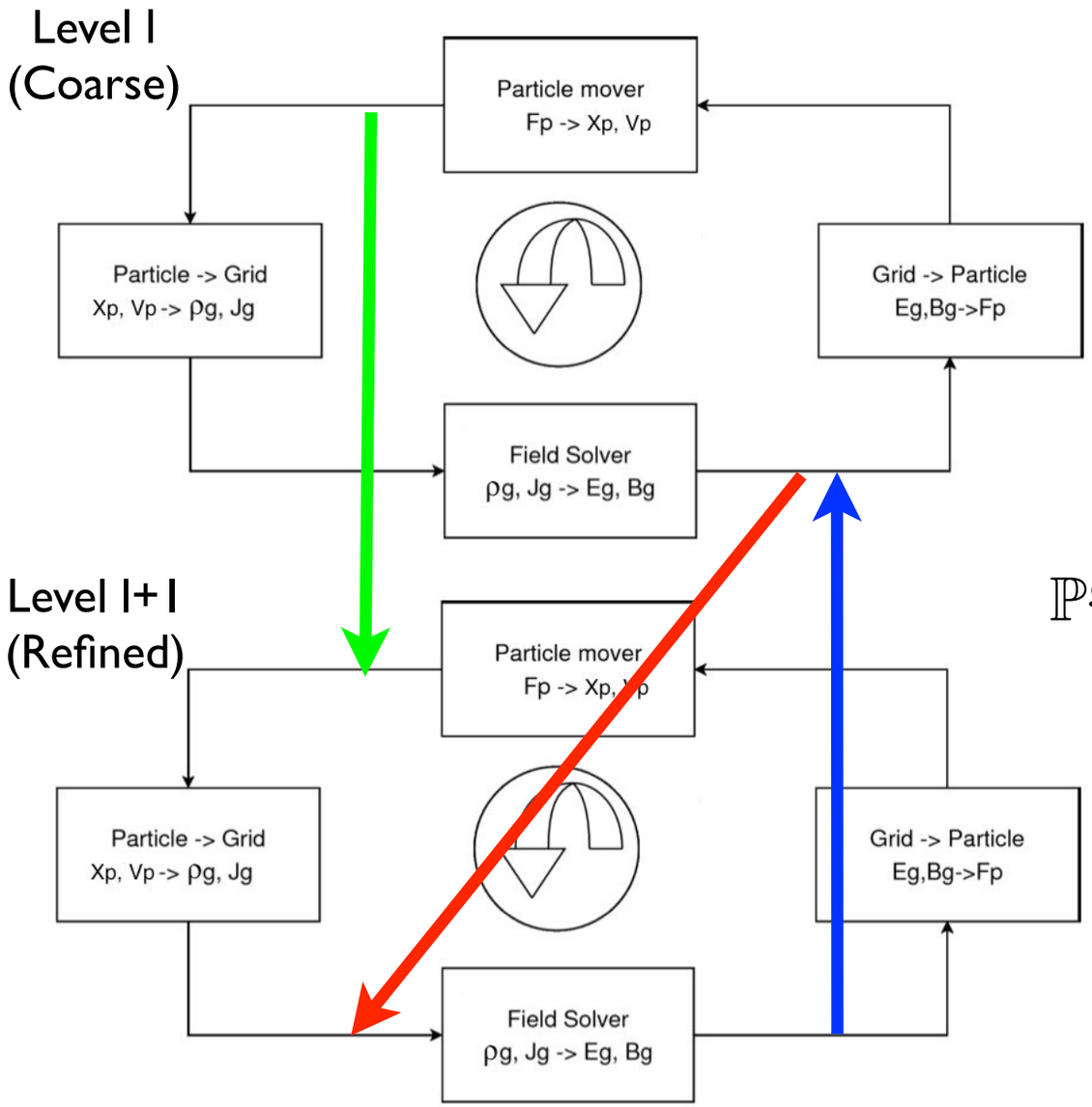
... we are quite far from this ...

The Multi Level Multi Domain (MLMD) method



- if only a portion of the total domain requires high resolution, different levels are simulated with the resolution locally required
- the IMM is used as baseline algorithm to bypass the strict stability constraints of explicit PIC, to have increased freedom in the choice of the Refinement Ratio (RR) between the grids and also to take advantage of its damping properties
- all levels are simulated fully with field are particles: boundary refined grid particles are created with a splitting algorithm from the corresponding coarse grid particles, refined grid particles are lost when they exit the refined domain
- the native coarse grid solution in the overlap area is not discarded, but contributes to the final solution

The Multi Level Multi Domain (MLMD) method



1. Boundary condition interpolation

2. Refined field projection

3. Refined particle repopulation

Information exchange steps between the grids:

1. boundary condition interpolation

$$\Xi_{I,g_{l+1}} = \sum_{g_l} \Xi_{N,g_l} W_{g_l}(\mathbf{x}_{g_l} - \mathbf{x}_{g_{l+1}})$$

2. refined field projection; average for momentum conservation purposes

$$\mathbf{E}_{P,g_l} = \frac{1}{2}\left(\mathbf{E}_{N,g_l} + \mathbb{P}^{g_{l+1}\to g_l}\left(\mathbf{E}_{N,g_l+1}\right)\right)$$

$$\mathbb{P}^{g_{l+1}\to g_l}\left(\mathbf{E}_{N,g_l+1}\right) = \frac{\sum_{g_{l+1}} \mathbf{E}_{N,g_{l+1}} W_{g_l}(\mathbf{x}_{g_l} - \mathbf{x}_{g_{l+1}})}{\sum_{g_{l+1}} W_{g_l}(\mathbf{x}_{g_l} - \mathbf{x}_{g_{l+1}})}$$

3. refined particle repopulation; splitting algorithm for optimal particle BC at the grid interface

$$q^{n+1}_{p_{g_{l+1}}} = q^{n+1}_{p_{g_l}} / RF^D$$

$$\mathbf{v}^{n+1}_{p_{g_{l+1}}} = \mathbf{v}^{n+1}_{p_{g_l}}$$

$$\mathbf{x}^{n+1}_{p_{g_{l+1}},i} = \mathbf{x}^{n+1}_{p_{g_l}} - \frac{\Delta\mathbf{x}_{g_l}}{2} + \Delta\mathbf{x}_{g_{l+1}}\left(\frac{1}{2} + i\right) - \mathbf{x}_{0,l+1}$$

i= 0 : RF - 1, per direction

This communication steps between the levels introduce bottlenecks in the parallel execution

To sum up, before showing test cases

Is the MLMD method "TOFU complaint"?

Implicit

Adaptive

Multi scale

To sum up, before showing test cases

Is the MLMD method "TOFU complaint"?

Implicit ✓

Adaptive

Multi scale

To sum up, before showing test cases

Is the MLMD method "TOFU complaint"?

Implicit ✓

Adaptive ✓

Multi scale

To sum up, before showing test cases

Is the MLMD method "TOFU complaint"?

Implicit ✓

Adaptive ✓

Multi scale ✓

(in the IMM, the scales not resolved are damped, not suppressed, and can be recovered with appropriate Dx and Dt)

To sum up, before showing test cases

Is the MLMD method "TOFU complaint"?

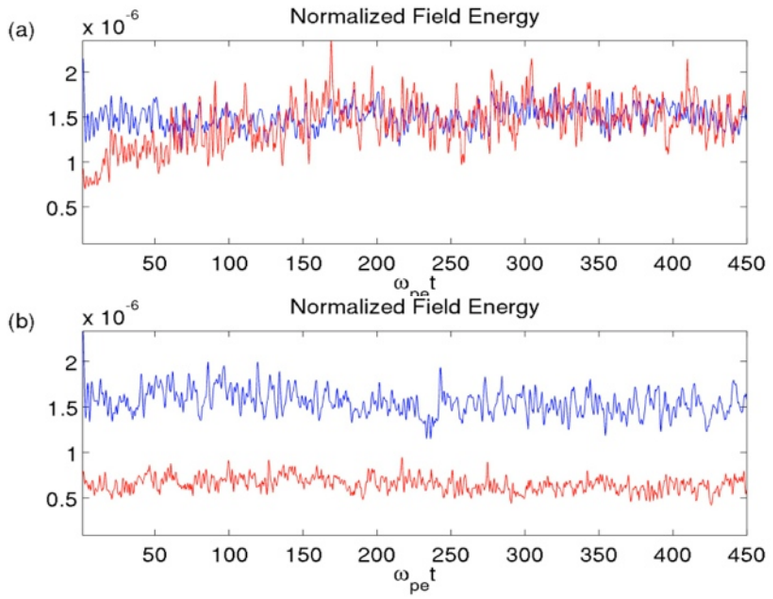
Implicit ✓

Adaptive ✓

Multi scale ✓

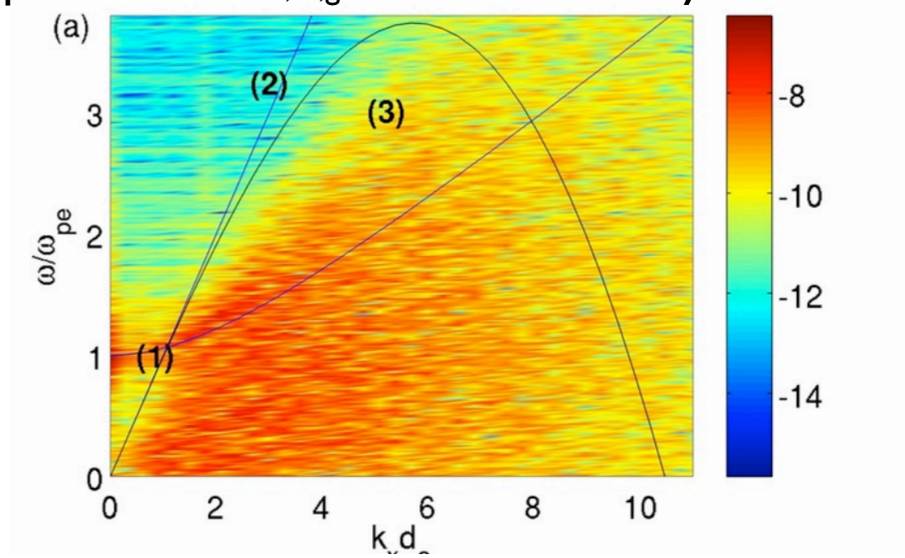
(in the IMM, the scales not resolved are damped, not suppressed, and can be recovered with appropriate Dx and Dt)

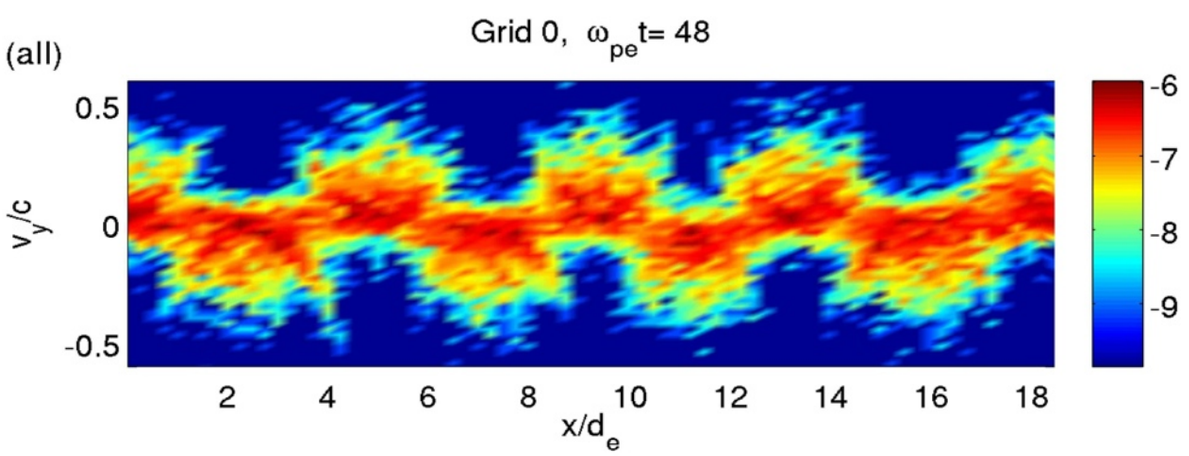
MLDM test cases: 1D simulations



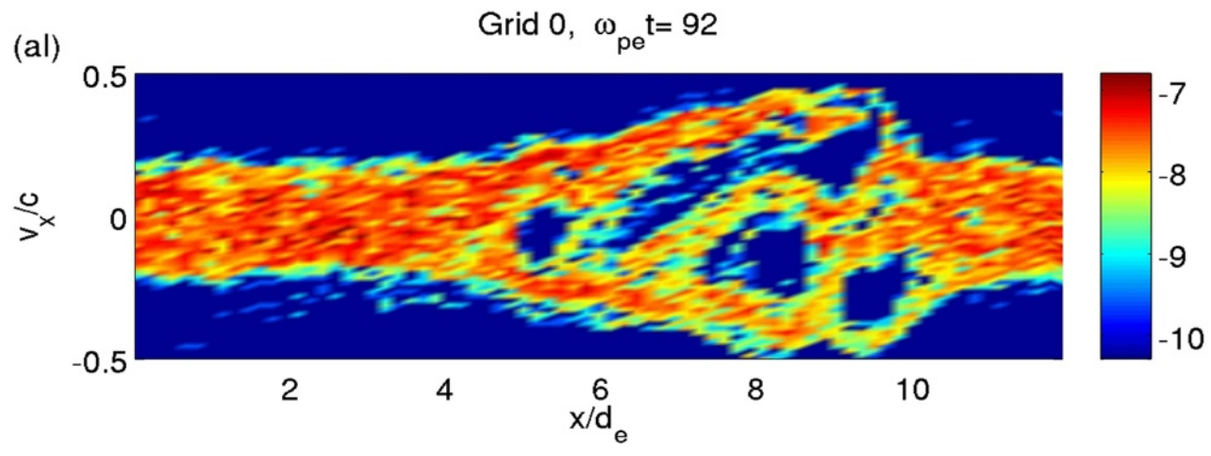
Some noise is indeed injected from the coarse into the refined grid,
but it does not damage the physical significance of the simulation

Spectrum of $E_{x,N,gII}$ in the MLMD system



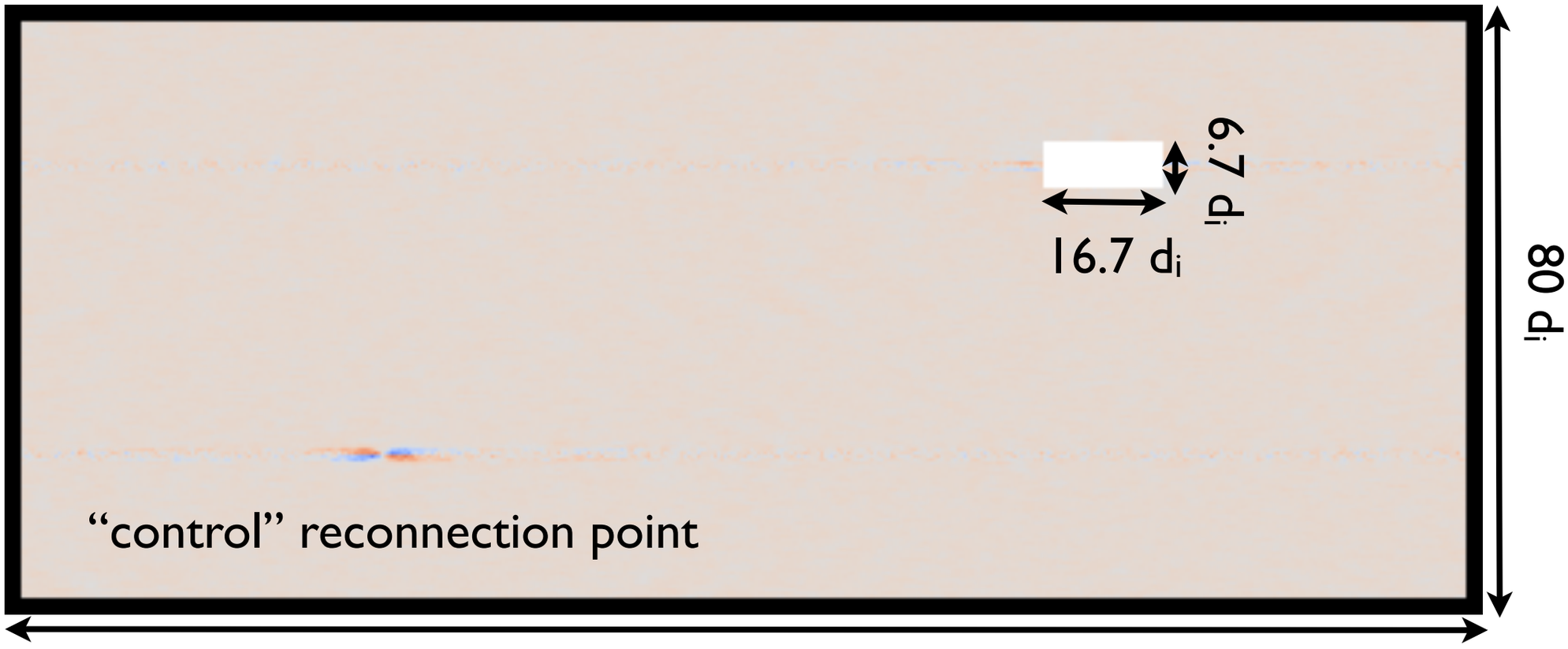


Phase space in the perpendicular direction for the Weibel instability in a 1D3V setting after saturation



Phase space in the longitudinal direction for the two stream instability in a 1D3V setting in the electron hole merging phase

MLMD test cases:
a big-domain high-resolution magnetic reconnection problem simulated with Parsek2D-MLMD



Mass Ratio: 256, Refinement Ratio between the levels: 12

Coarse Level

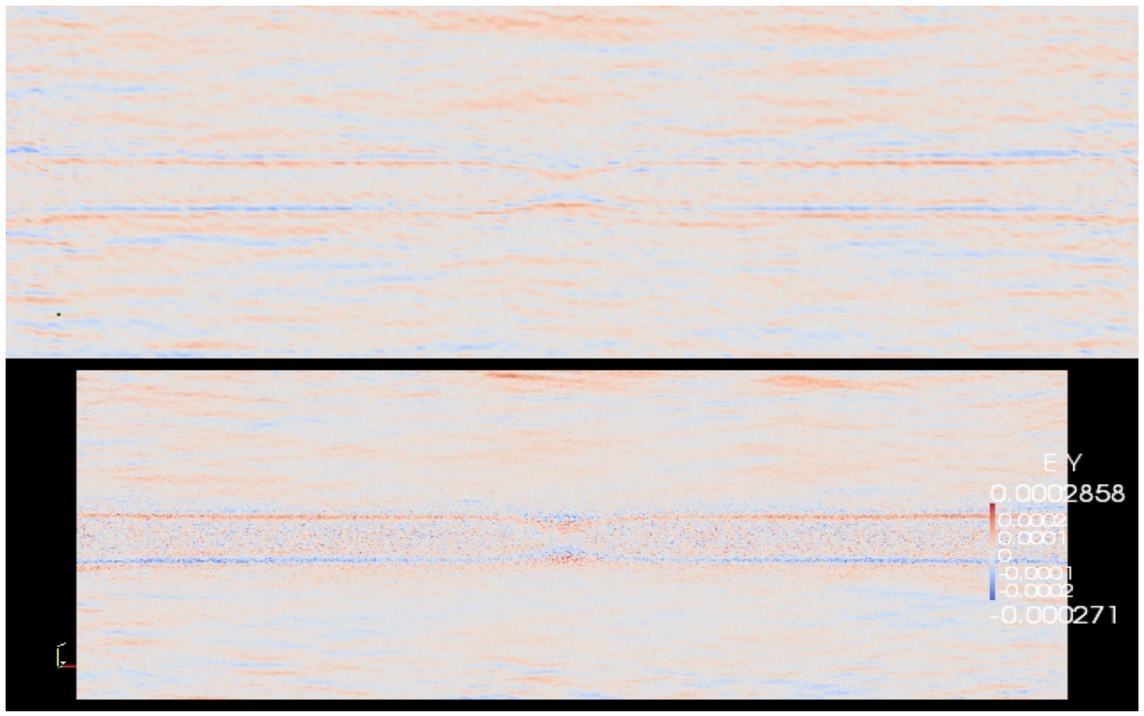
2560 x 1024 cells, 1.5 10^8 particles
$\Delta x = \Delta y = 0.078\ d_i \sim$ 1/10 d_i

Refined Level

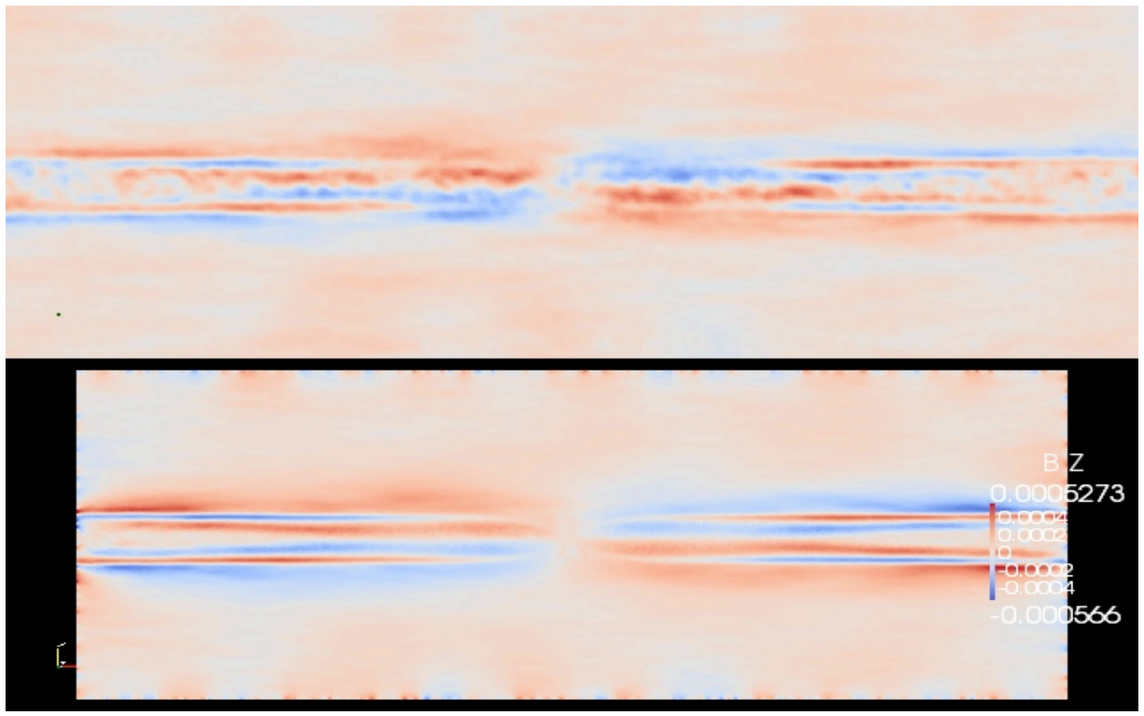
2560 x 1024 cells, 1.5 10^8 particles
$d_e = 0.0625\ d_i$
$\Delta x = \Delta y = 0.0065\ d_i \sim$ 1/10 d_e

MLMD test cases:
a big-domain high-resolution magnetic reconnection problem simulated with Parsek2D-MLMD

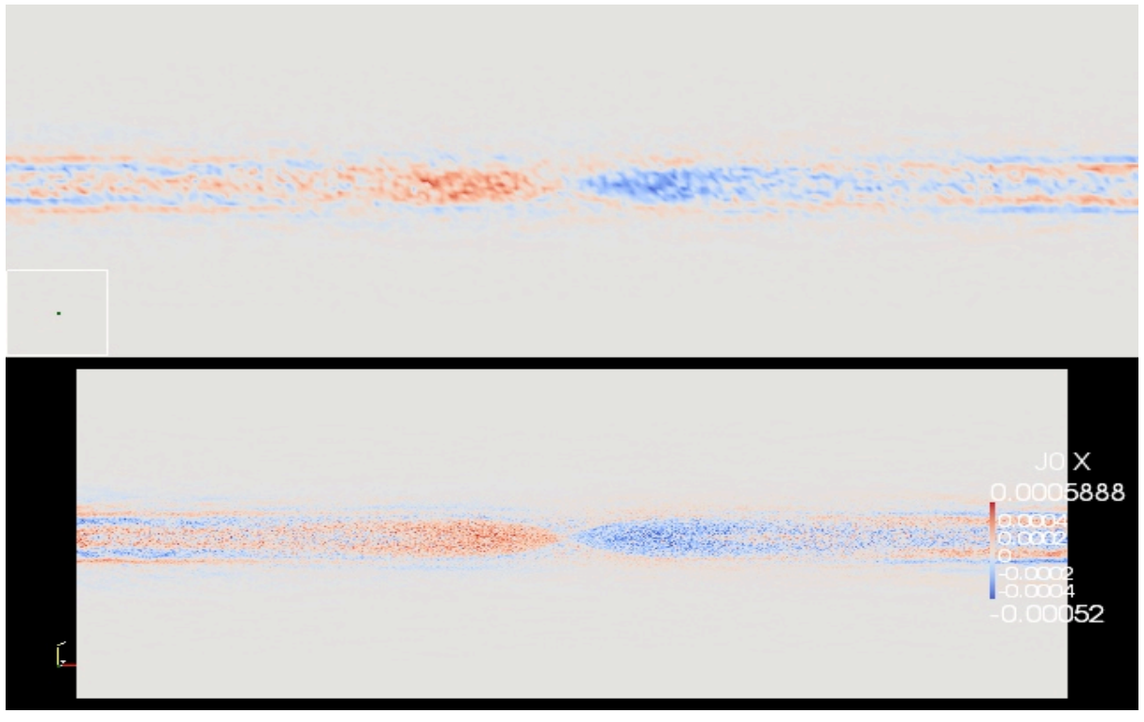
Time: 1.36 Ω_{ci}



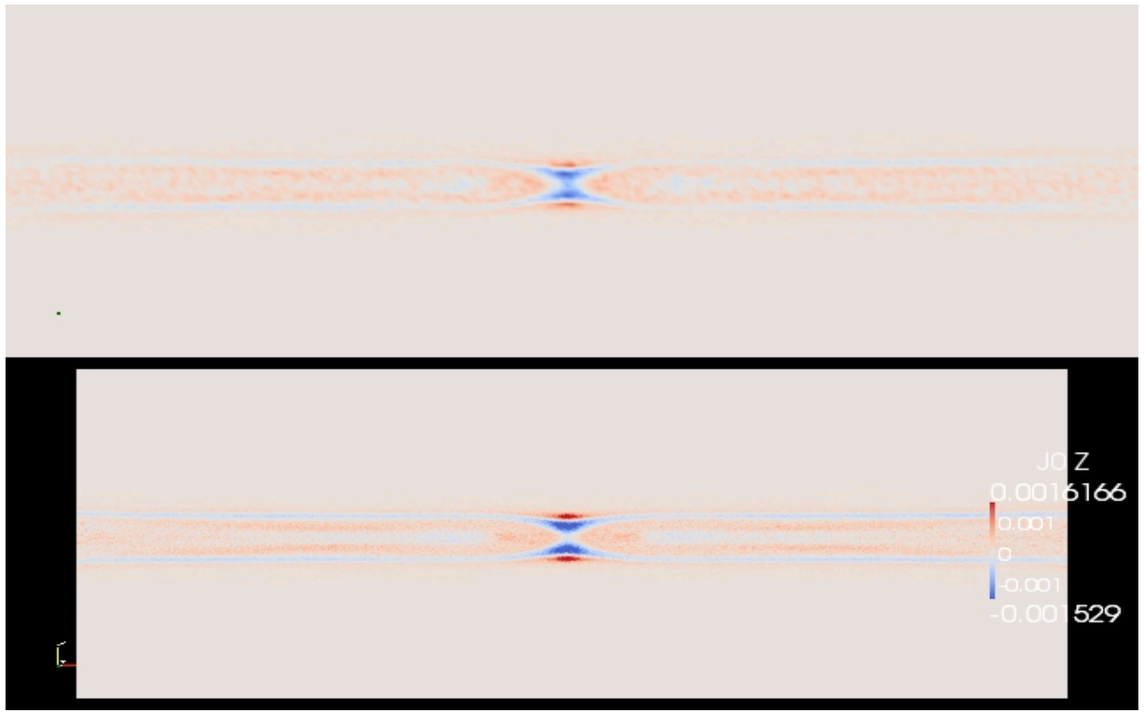
← Ey



Bz →



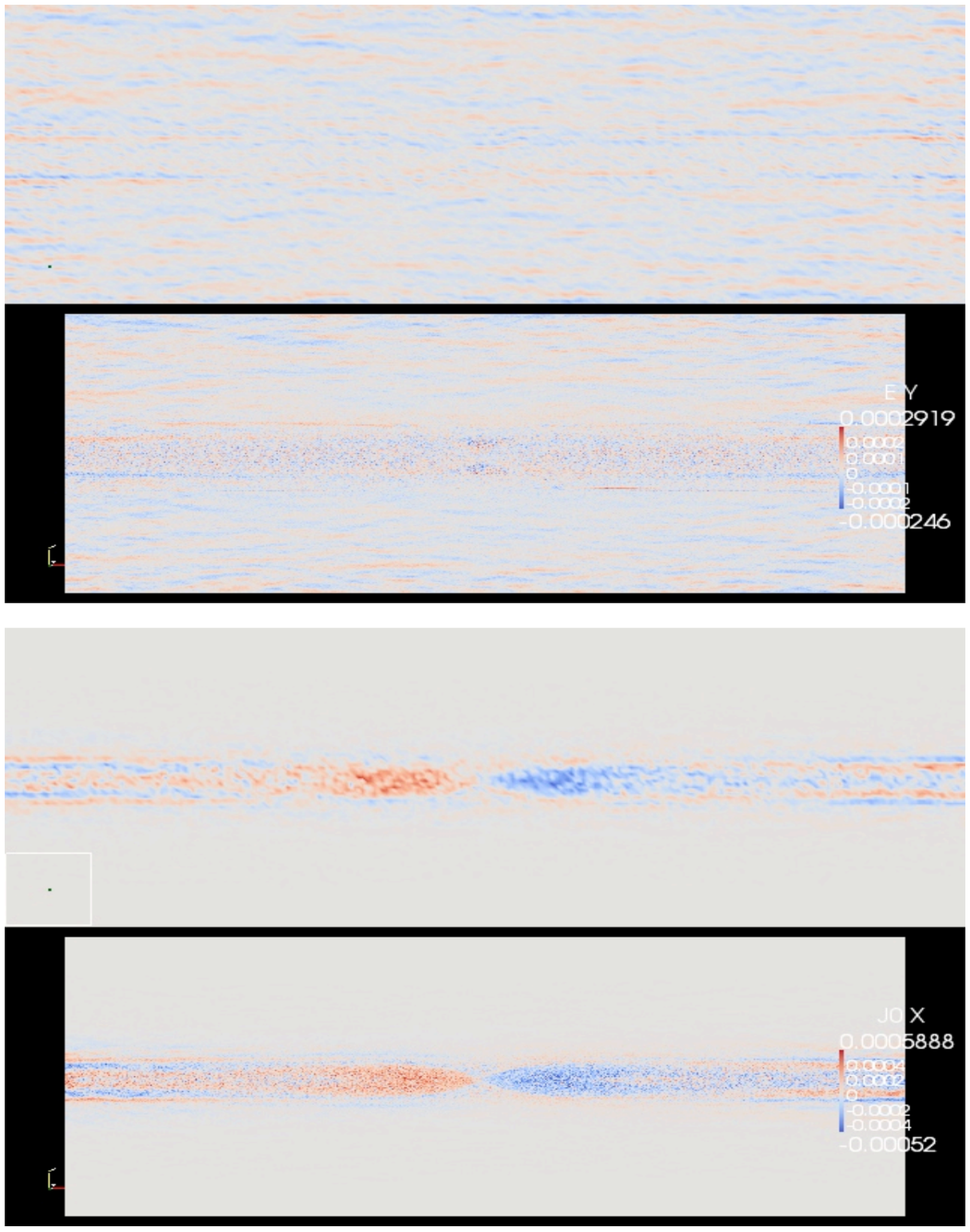
← J0x



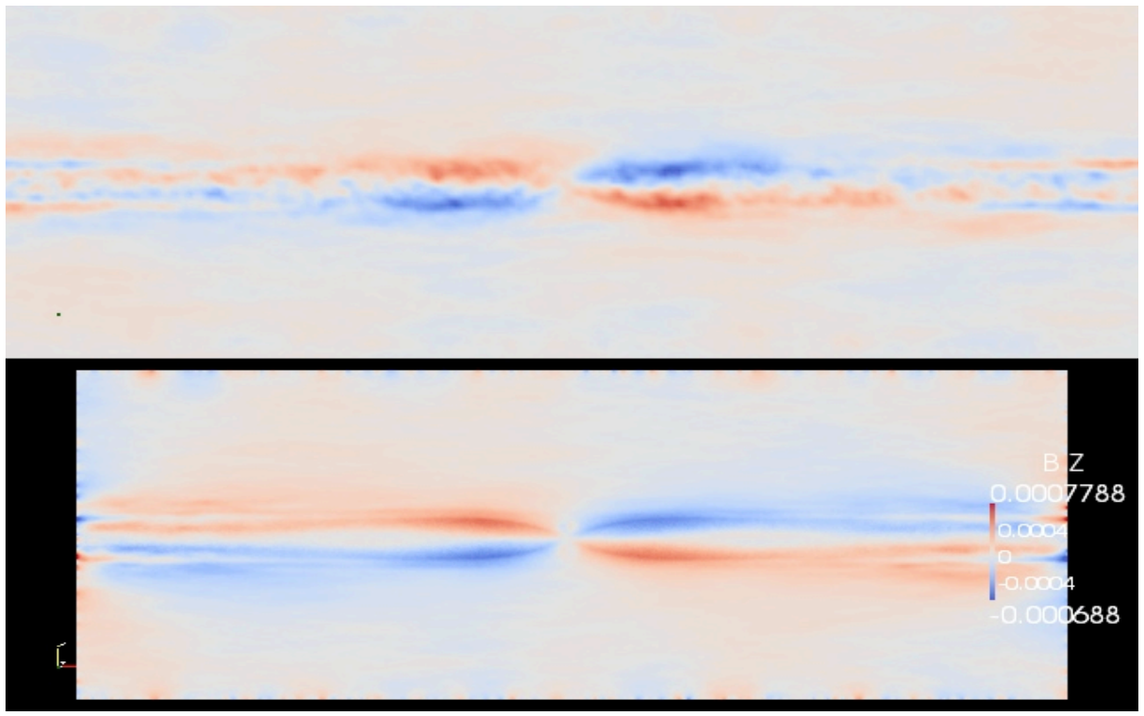
J0z →

MLMD test cases:
a big-domain high-resolution magnetic reconnection problem simulated with Parsek2D-MLMD

Time: 2.28 Ω_{ci}

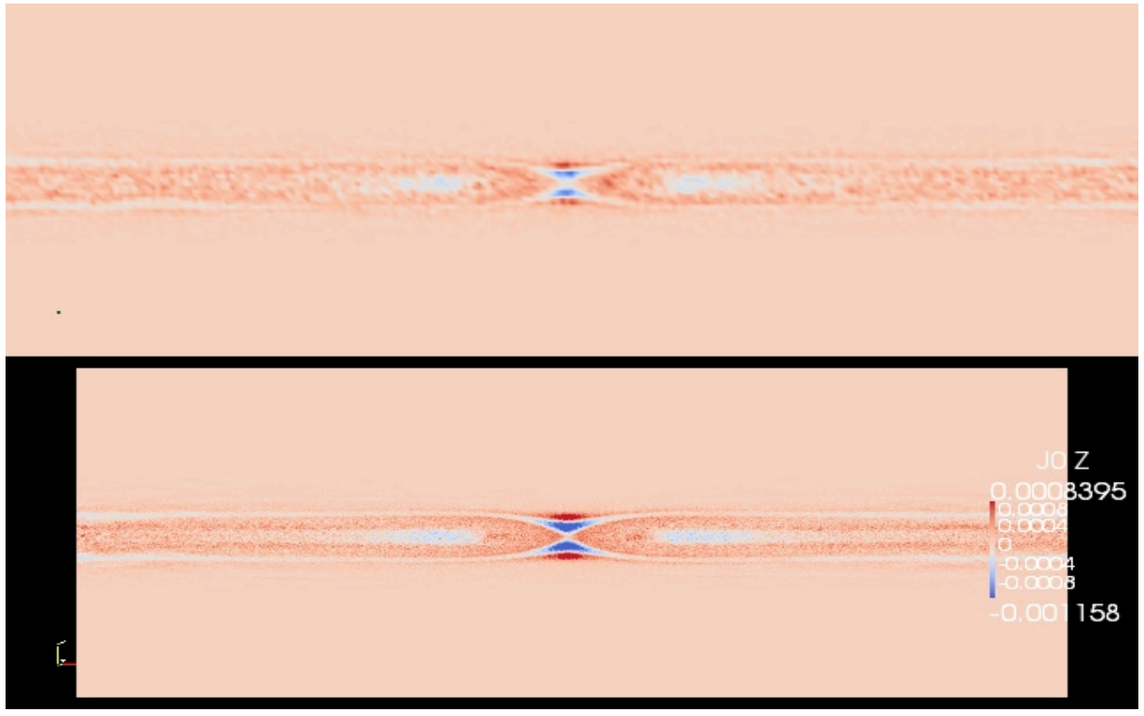


← Ey



Bz →

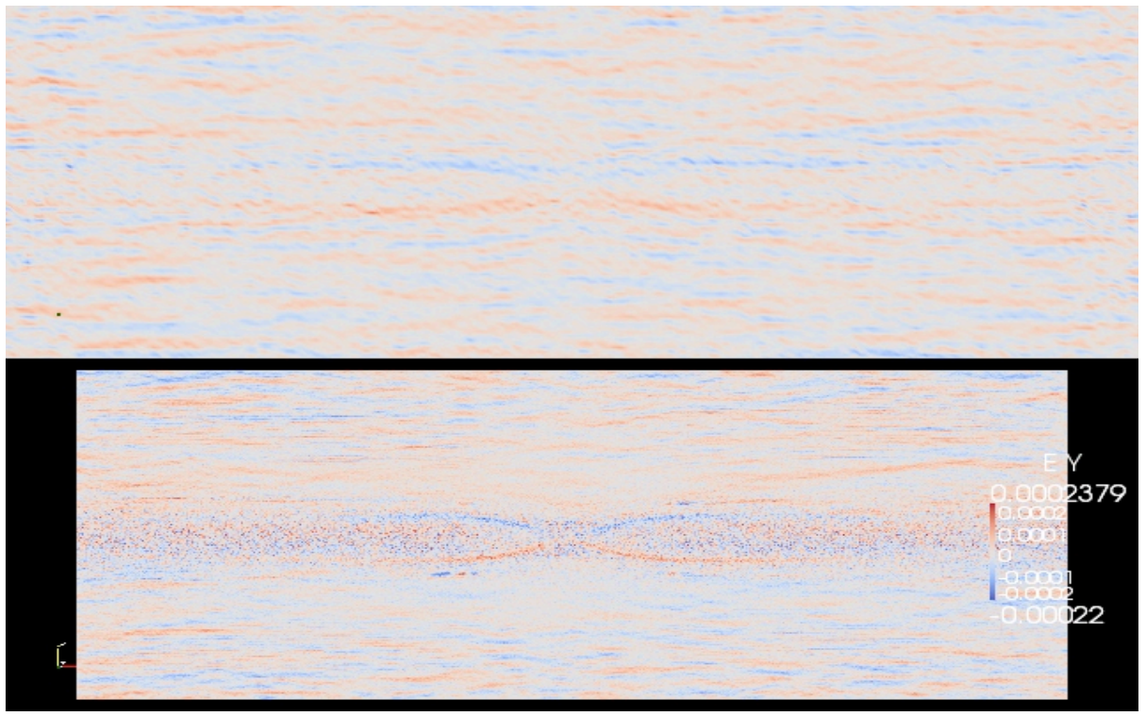
← J0x



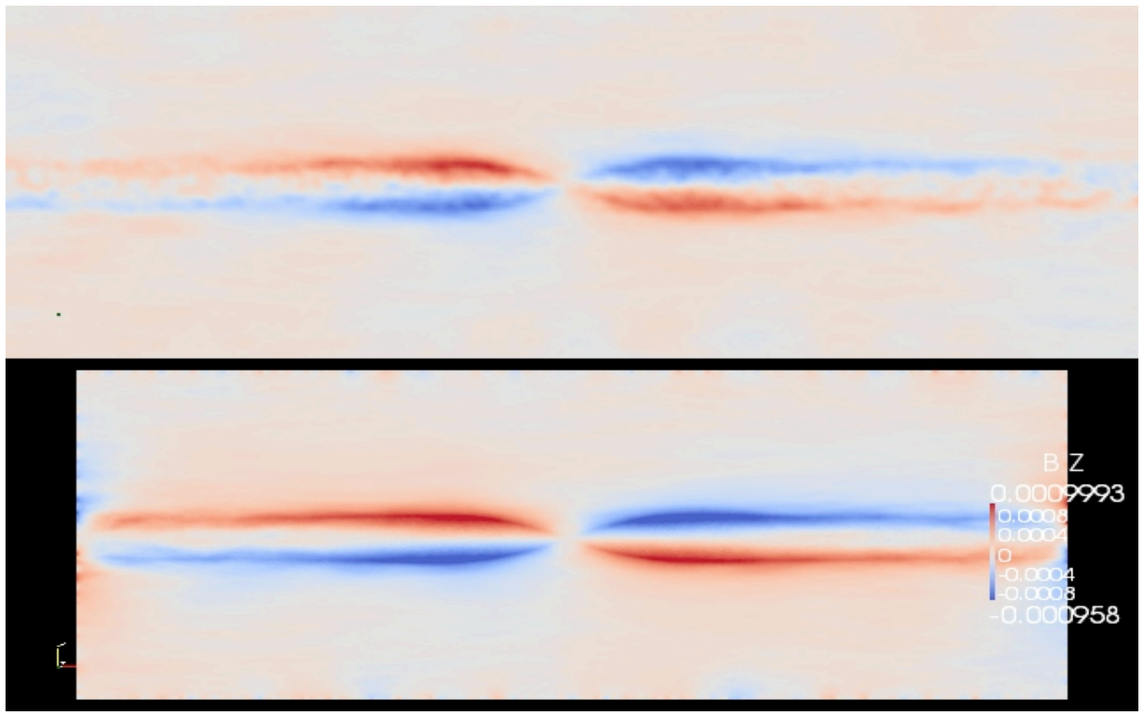
J0z →

MLMD test cases:
a big-domain high-resolution magnetic reconnection problem simulated with Parsek2D-MLMD

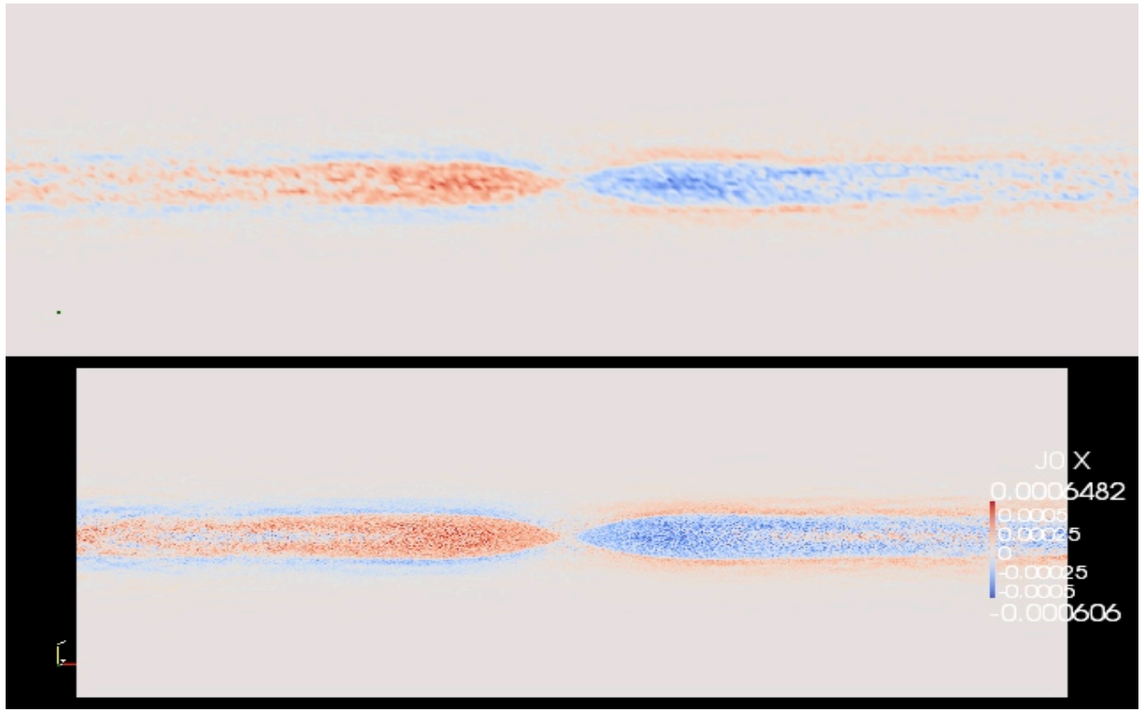
Time: 3.65 Ω_{ci}



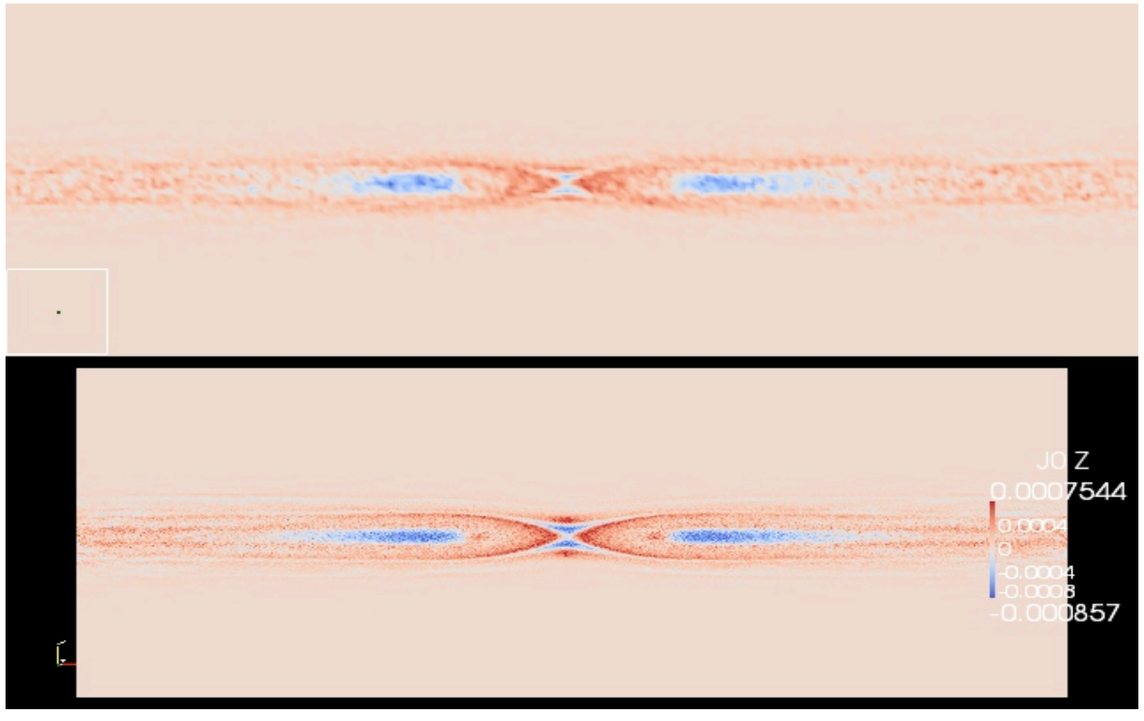
← Ey



Bz →



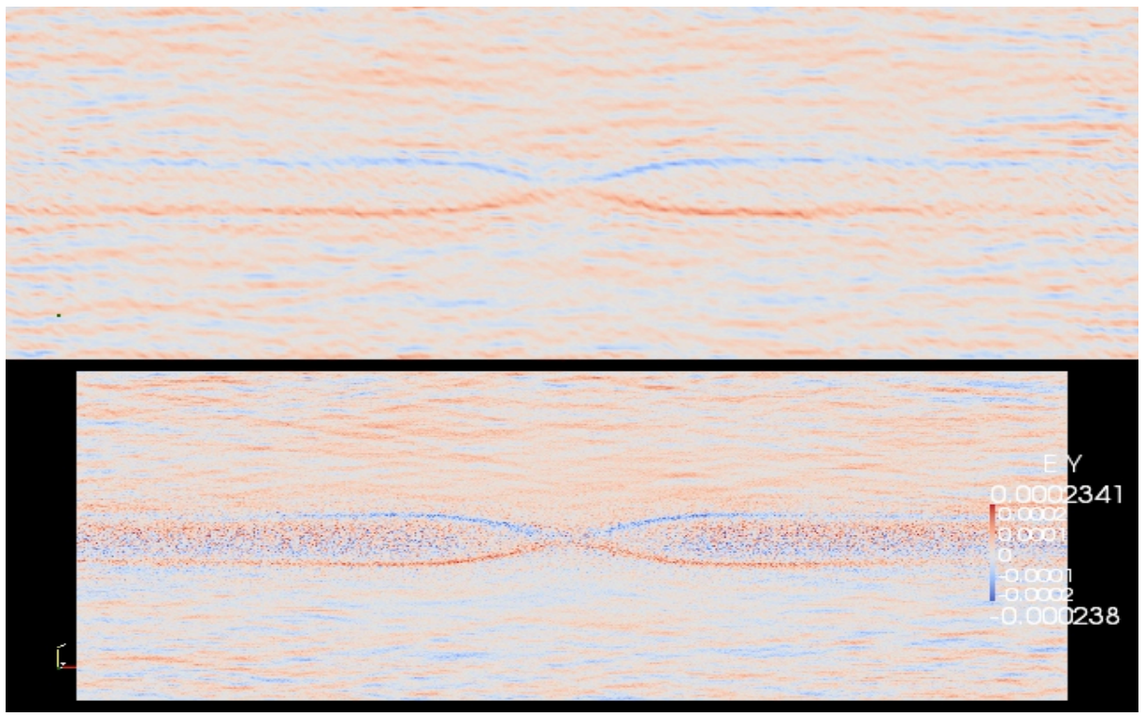
← J0x



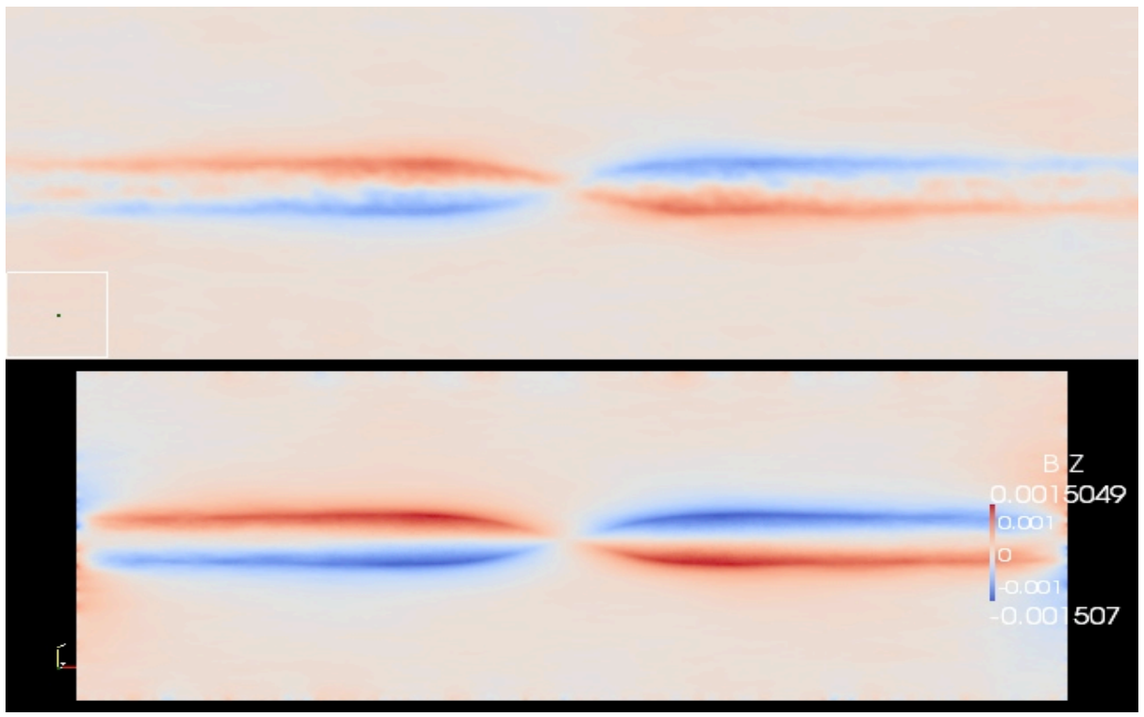
J0z →

MLMD test cases:
a big-domain high-resolution magnetic reconnection problem simulated with Parsek2D-MLMD

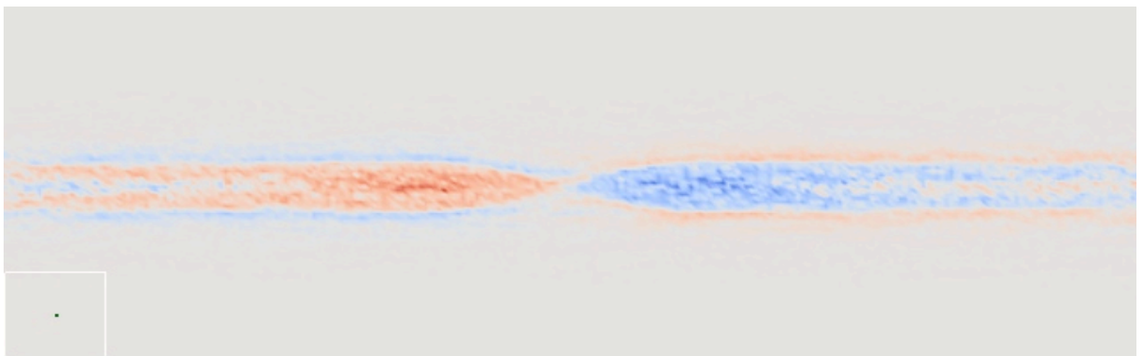
Time: 5.02 Ω_{ci}



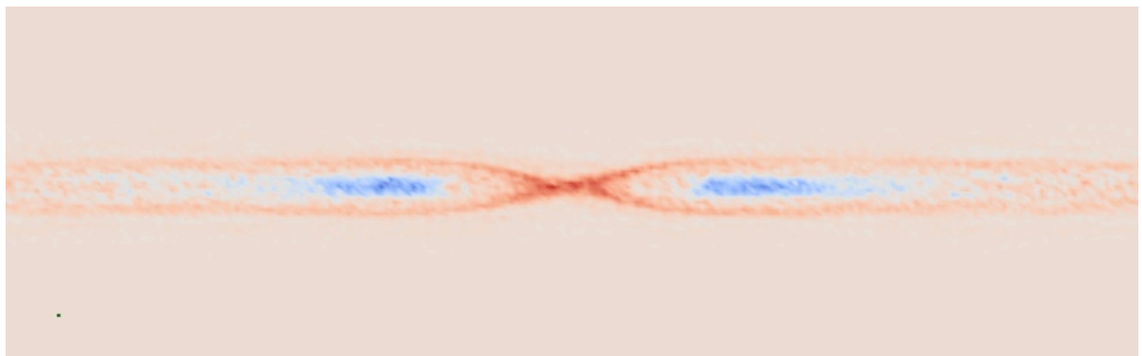
← Ey

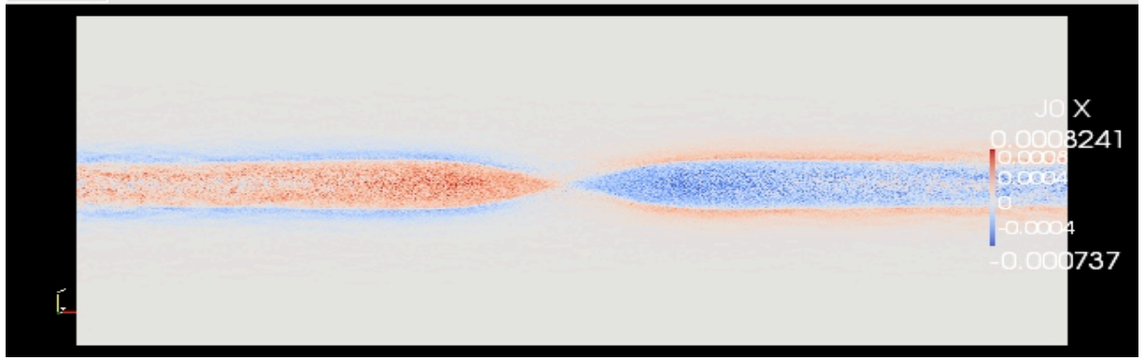


Bz →

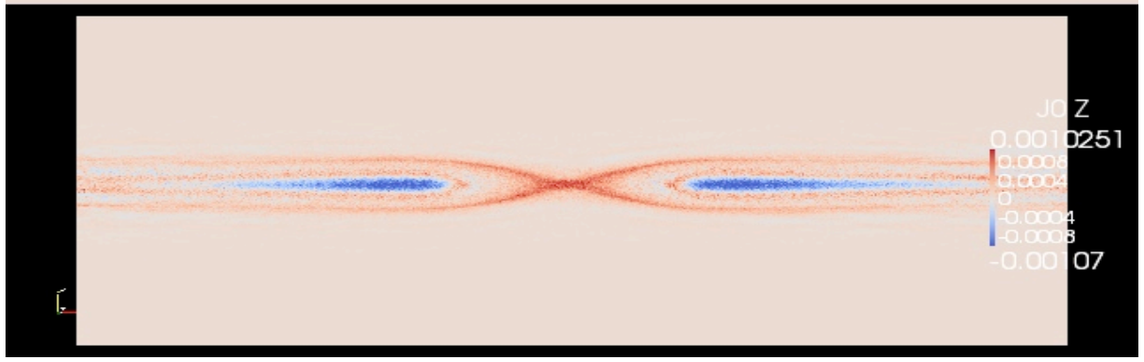


←J0x



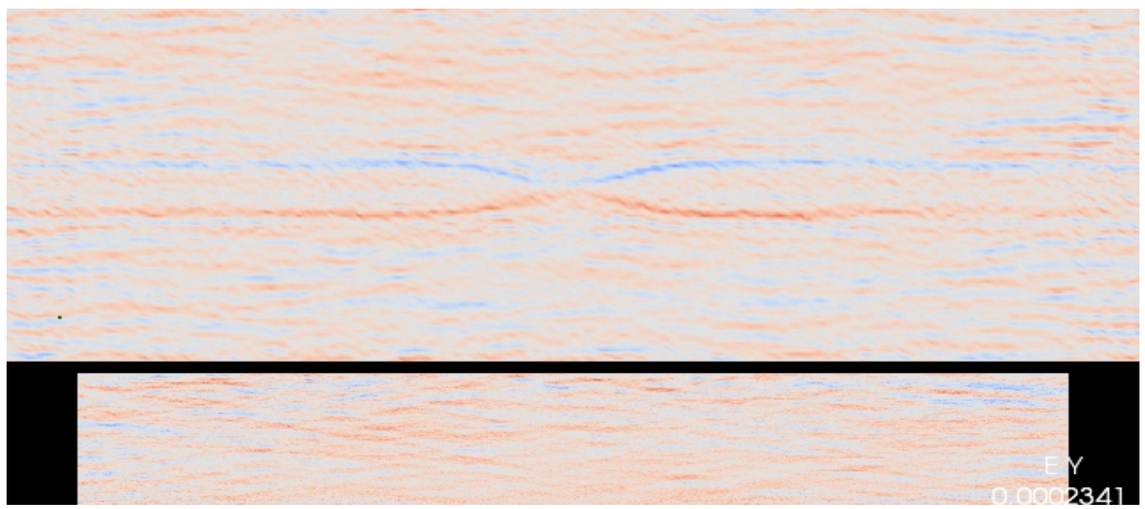


J0z →

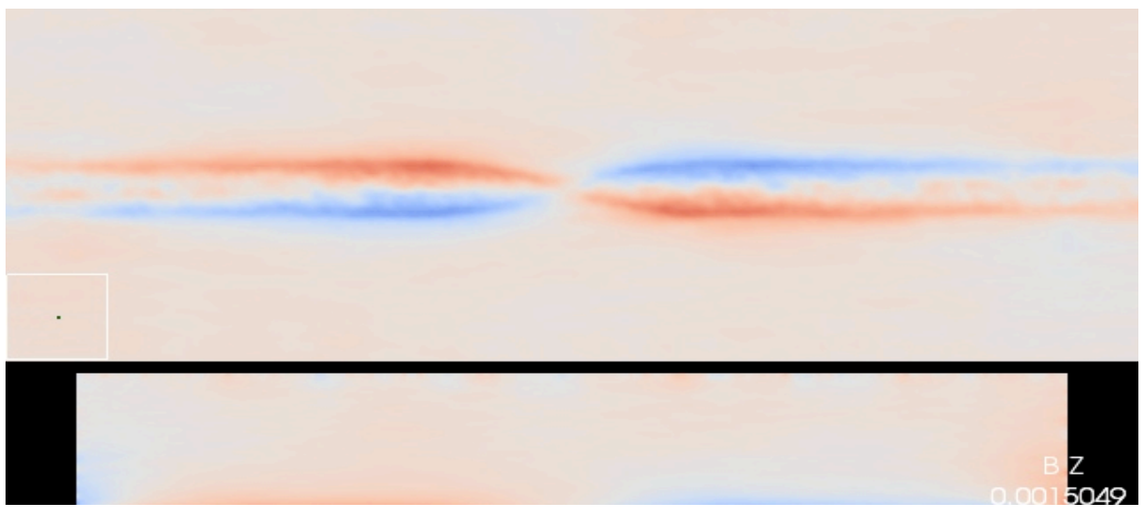


MLMD test cases:
a big-domain high-resolution magnetic reconnection problem simulated with Parsek2D-MLMD

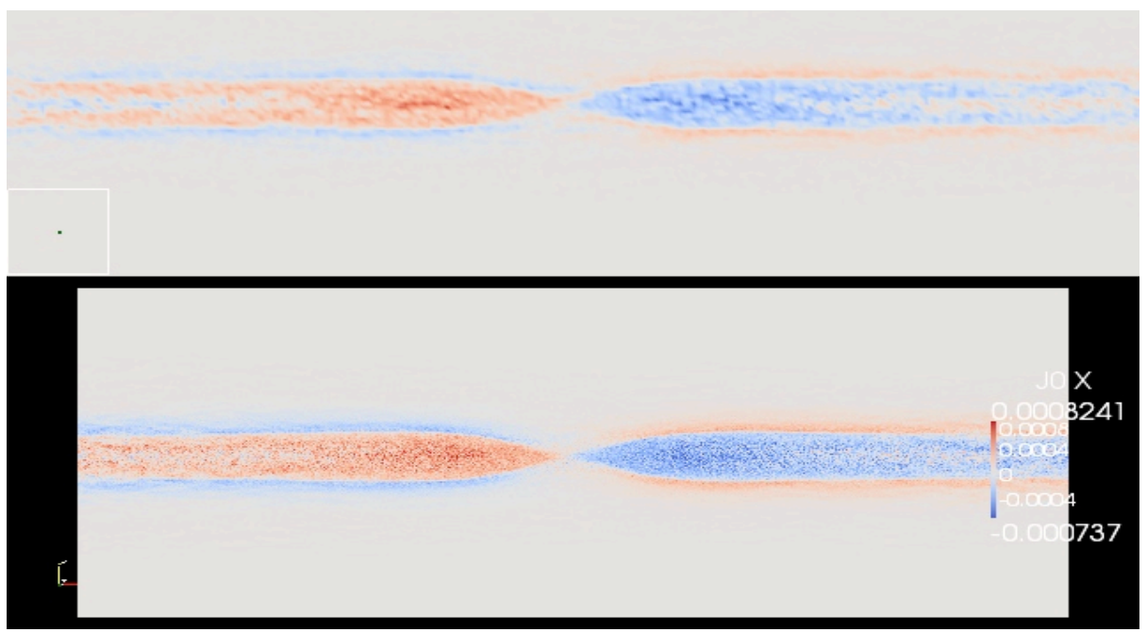
Time: 5.02 Ω_{ci}



← Ey

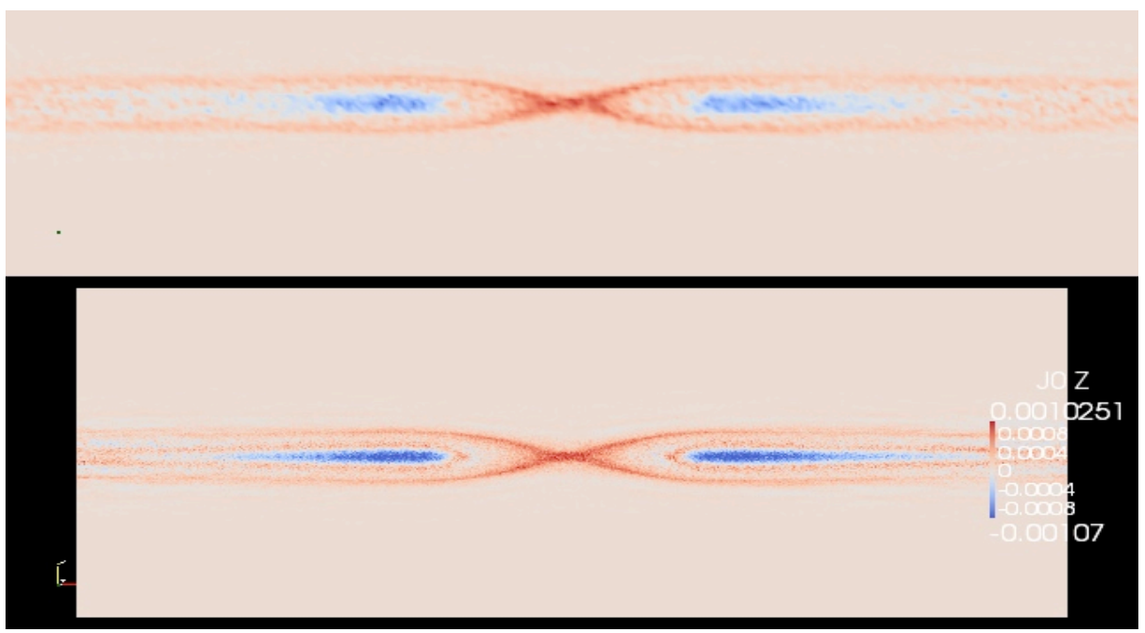


reaching this level of resolution on the entire domain
would require 30720 x 12288 cells and
21 10^9 particles
→absolutely challenging



←J0x

J0z →



MLMD test cases:
a big-domain high-resolution magnetic reconnection problem simulated with Parsek2D-MLMD

Why is this simulation so significative?

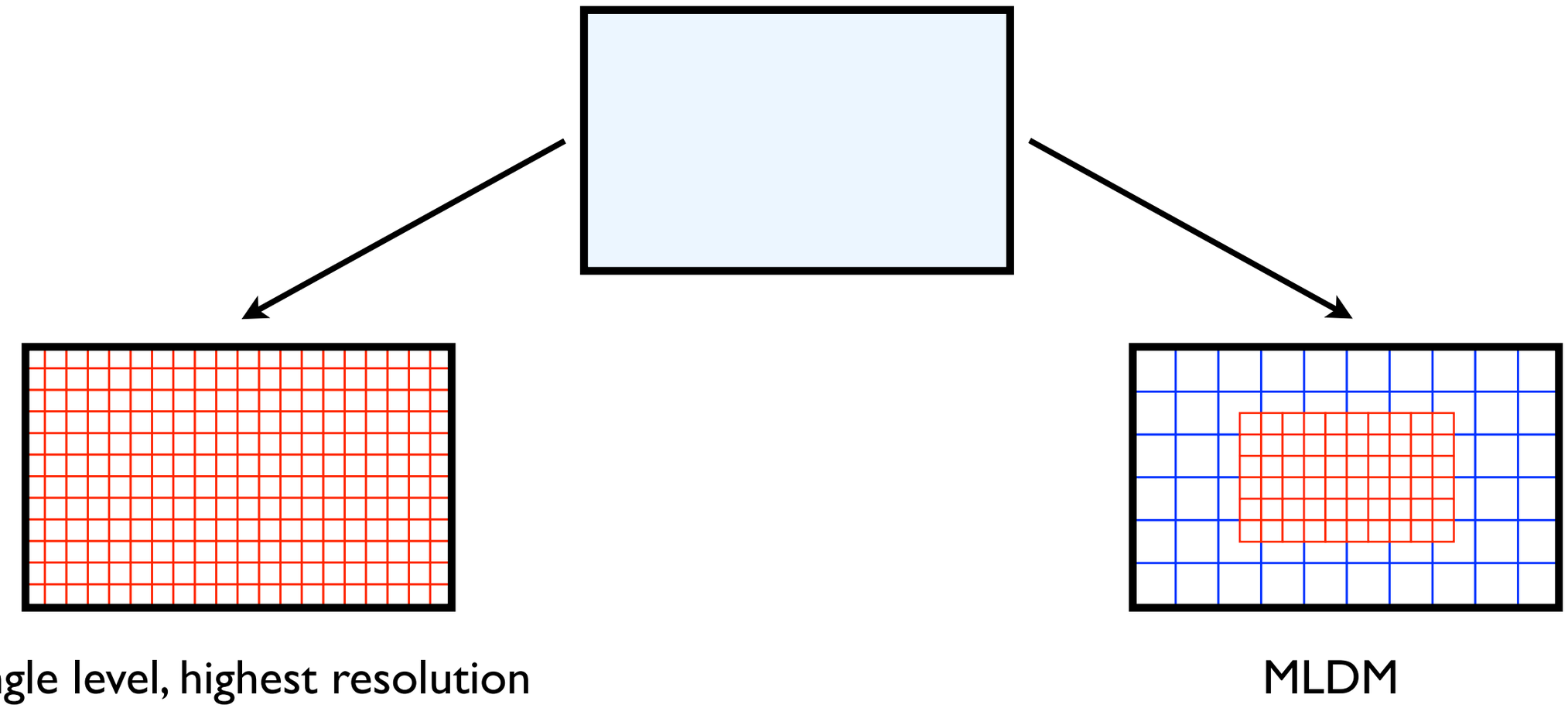
- very big domain: it is possible to follow Electron Distribution Region dynamics and the development of dipolarization fronts in the same simulation, with appropriate resolutions

- very high mass ratio (at least when comparing with explicit simulations)

- very high Refinement Ratio between the grids: $\Delta x \sim 1/10\ d_i$ on the coarse grid, $\Delta x \sim 1/10\ d_e$ on the refined grid

Performances

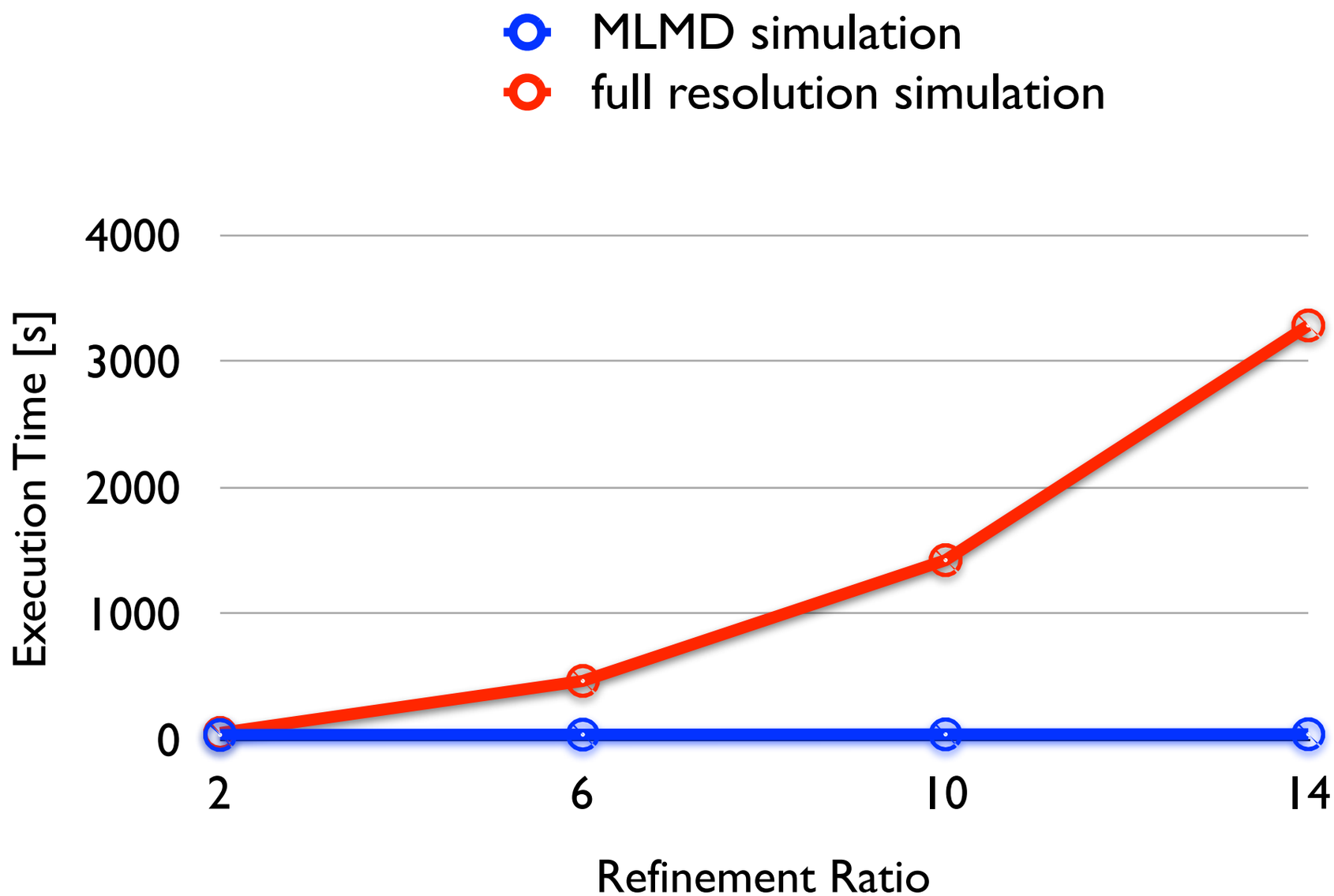
With fixed domain size,
compare the execution time of a single level, highest resolution simulation and that of a MLMD simulation



single level, highest resolution

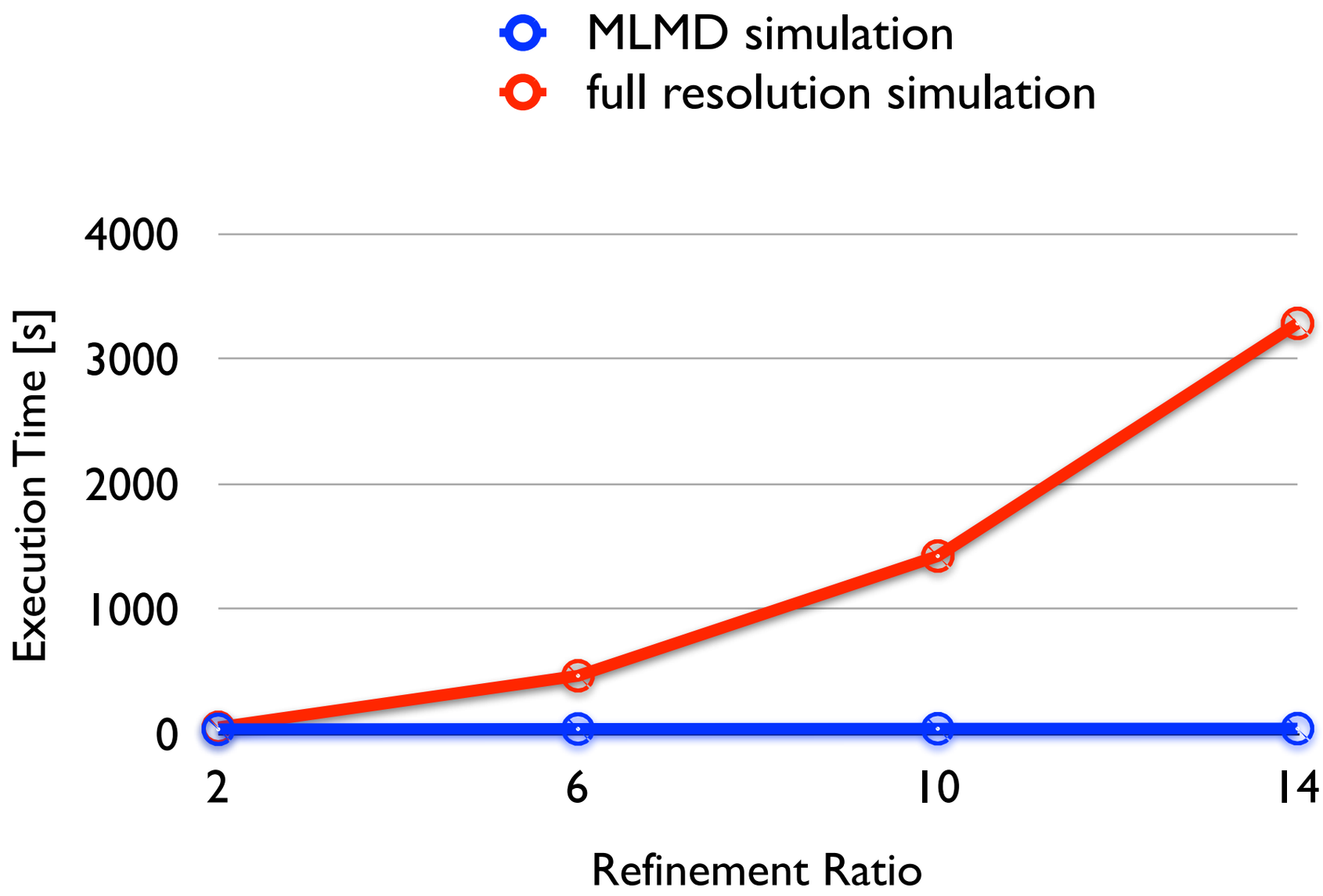
MLDM

Performances



...VERY GOOD !!!

Performances



the IMM allows very high Refinement Factors between the grids while still in the stability range of the method

...VERY GOOD !!!

Conclusions

- we have introduced the Multi Level Multi Domain method and Parsek2D-MLMD, a 2D parallel C++ implicit adaptive code

- the MLMD method combines two fundamental building blocks: the Implicit Moment Method and adaptivity

- communication operations between the levels allow optimal grid interlocking and excellent results with very high Refinement Ratios between the grids

- tests show that notable resources can be saved when comparing full resolution simulations and MLMD simulations

Future work

Three parallel lines of work:

- code optimization: improve code scalability for big domain simulations (PRACE Preparatory Access type C grant)

- scientific work: current sheet instabilities under realistic inflow conditions in reconnection problems, electron dynamics close to the shock front in collisionless shock simulations

- code development: 3D evolution, moving refined grids

References & contact info

Papers

M.E. Innocenti, G. Lapenta, S. Markidis, A. Beck, and A. Vapirev. *A Multi Level Multi Domain Method for Particle In Cell Plasma Simulations*. Journal of Computational Physics, 2013.

A. Beck, M.E. Innocenti, G. Lapenta, and S. Markidis. *Multi-level multi-domain algorithm implementation for two-dimensional multiscale particle in cell simulations*. submitted, 2013.

e-mail addresses

MariaElena.Innocenti@wis.kuleuven.be
beck@llr.in2p3.fr
markidis@pdc.kth.se
Giovanni.Lapenta@wis.kuleuven.be

References & contact info

Papers

M.E. Innocenti, G. Lapenta, S. Markidis, A. Beck, and A. Vapirev. *A Multi Level Multi Domain Method for Particle In Cell Plasma Simulations*. Journal of Computational Physics, 2013.

A. Beck, M.E. Innocenti, G. Lapenta, and S. Markidis. *Multi-level multi-domain algorithm implementation for two-dimensional multiscale particle in cell simulations*. submitted, 2013.

Thank you for your attention!

e-mail addresses

MariaElena.Innocenti@wis.kuleuven.be
beck@llr.in2p3.fr
markidis@pdc.kth.se
Giovanni.Lapenta@wis.kuleuven.be