# A semi-Lagrangian AMR scheme for 2D transport problems in conservation form

Pep Mulet Mestre, Francesco Vecil

Universitat de València

WONAPDE, Concepción (Chile), 16 January 2013

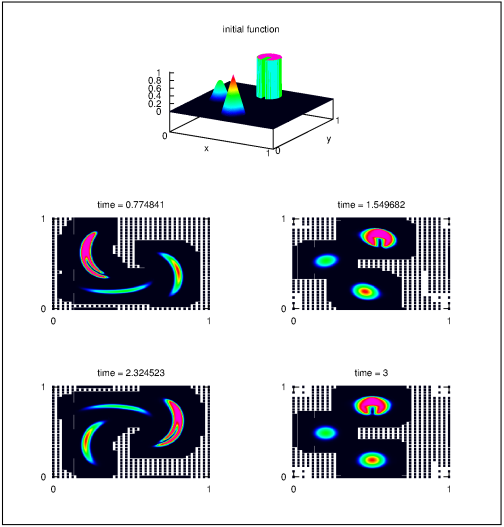
# Outline


1. Introduction

2. Numerical tools
   - Multiresolution framework
   - Time integration

3. Experiments
   - Introduction
   - 1D tests
   - 2D tests

# Motivation




No need for fine meshing everywhere in the domain.

$\Downarrow$

Refine only where the important information is.

# Framework

## Equations

In dimension $N$, transport equations written in conservtion form:

$$\frac{\partial u}{\partial t} + \mathrm{div}_{\boldsymbol{x}} \left[\boldsymbol{a}(t,\boldsymbol{x})u\right] = 0, \qquad u(0,\boldsymbol{x}) = u^0(\boldsymbol{x}), \qquad (t,\boldsymbol{x}) \in \mathbb{R}_{\geq 0} \times \Omega,$$

$\Omega = \prod_{n=1}^{N} [(x_n)_{\min}, (x_n)_{\max}]$ is the domain, $\boldsymbol{a} : \mathbb{R}_{\geq 0} \times \Omega \to \mathbb{R}^N$ is the advection field.

## Example

The three-dimensional Vlasov-Maxwell equation

$$\frac{\partial f}{\partial t} + \boldsymbol{v}(\boldsymbol{p}) \cdot \frac{\partial f}{\partial \boldsymbol{x}} + \boldsymbol{F} \cdot \frac{\partial f}{\partial \boldsymbol{p}} = 0, \quad \boldsymbol{v}(\boldsymbol{p}) := \frac{\boldsymbol{p}}{m\sqrt{1 + \frac{|\boldsymbol{p}|^2}{m^2c^2}}}, \quad \boldsymbol{F} := -e(\boldsymbol{E} + \boldsymbol{v}(\boldsymbol{p}) \wedge \boldsymbol{B}),$$

describes the evolution of $f(t,\boldsymbol{x},\boldsymbol{p})$, typically representing the concentration of electrons or holes at position $\boldsymbol{x}$ and momentum $\boldsymbol{p}$.

## Features

Shocks, large gradients, filamentation, microscopic structures.

# Framework

## Equations

In dimension $N$, transport equations written in conservtion form:

$$\frac{\partial u}{\partial t} + \mathrm{div}_{\boldsymbol{x}}\left[\boldsymbol{a}(t,\boldsymbol{x})u\right] = 0, \qquad u(0,\boldsymbol{x}) = u^0(\boldsymbol{x}), \qquad (t,\boldsymbol{x}) \in \mathbb{R}_{\geq 0} \times \Omega,$$

$\Omega = \prod_{n=1}^{N}[(x_n)_{\min}, (x_n)_{\max}]$ is the domain, $\boldsymbol{a} : \mathbb{R}_{\geq 0} \times \Omega \to \mathbb{R}^N$ is the advection field.

## Example

The three-dimensional Vlasov-Maxwell equation

$$\frac{\partial f}{\partial t} + \boldsymbol{v}(\boldsymbol{p}) \cdot \frac{\partial f}{\partial \boldsymbol{x}} + \boldsymbol{F} \cdot \frac{\partial f}{\partial \boldsymbol{p}} = 0, \quad \boldsymbol{v}(\boldsymbol{p}) := \frac{\boldsymbol{p}}{m\sqrt{1 + \frac{|\boldsymbol{p}|^2}{m^2 c^2}}}, \quad \boldsymbol{F} := -e(\boldsymbol{E} + \boldsymbol{v}(\boldsymbol{p}) \wedge \boldsymbol{B}),$$

describes the evolution of $f(t,\boldsymbol{x},\boldsymbol{p})$, typically representing the concentration of electrons or holes at position $\boldsymbol{x}$ and momentum $\boldsymbol{p}$.

## Features

Shocks, large gradients, filamentation, microscopic structures.

# Framework

### Equations

In dimension $N$, transport equations written in conservtion form:

$$\frac{\partial u}{\partial t} + \mathrm{div}_{\boldsymbol{x}} \left[\boldsymbol{a}(t,\boldsymbol{x})u\right] = 0, \qquad u(0,\boldsymbol{x}) = u^0(\boldsymbol{x}), \qquad (t,\boldsymbol{x}) \in \mathbb{R}_{\geq 0} \times \Omega,$$

$\Omega = \prod_{n=1}^{N} [(x_n)_{\min}, (x_n)_{\max}]$ is the domain, $\boldsymbol{a} : \mathbb{R}_{\geq 0} \times \Omega \to \mathbb{R}^N$ is the advection field.

### Example

The three-dimensional Vlasov-Maxwell equation

$$\frac{\partial f}{\partial t} + \boldsymbol{v}(\boldsymbol{p}) \cdot \frac{\partial f}{\partial \boldsymbol{x}} + \boldsymbol{F} \cdot \frac{\partial f}{\partial \boldsymbol{p}} = 0, \quad \boldsymbol{v}(\boldsymbol{p}) := \frac{\boldsymbol{p}}{m\sqrt{1 + \frac{|\boldsymbol{p}|^2}{m^2c^2}}}, \quad \boldsymbol{F} := -e(\boldsymbol{E} + \boldsymbol{v}(\boldsymbol{p}) \wedge \boldsymbol{B}),$$

describes the evolution of $f(t,\boldsymbol{x},\boldsymbol{p})$, typically representing the concentration of electrons or holes at position $\boldsymbol{x}$ and momentum $\boldsymbol{p}$.

### Features

Shocks, large gradients, filamentation, microscopic structures.

# Outline

1 Introduction

2 Numerical tools
- Multiresolution framework
- Time integration

3 Experiments
- Introduction
- 1D tests
- 2D tests

# Grid hierarchy

## Resolution levels

We define $L+1$ resolution levels: the coarsest is $\ell = 0$, the finest $\ell = L$. In 1D, the meshes are

$$x_{\ell,j} = x_{\min} + j\Delta x_\ell, \qquad \Delta x_\ell = \frac{x_{\max} - x_{\min}}{2^\ell N_0}.$$

## Grid

The $\ell$-grid at time $t^n$ is

$$G_\ell^n = \{x_{\ell,j}\}_{j \in \mathcal{G}_\ell^n}.$$

We are interested in

$$\mathcal{G}_\ell^n \subseteq \prod_{i=1}^{N} \{0, \dots, N_{i,\ell}\}.$$

# Grid hierarchy

### Resolution levels

We define $L+1$ resolution levels: the coarsest is $\ell = 0$, the finest $\ell = L$. In 1D, the meshes are

$$x_{\ell,j} = x_{\min} + j\Delta x_\ell, \qquad \Delta x_\ell = \frac{x_{\max} - x_{\min}}{2^\ell N_0}.$$

### Grid

The $\ell$-grid at time $t^n$ is

$$G_\ell^n = \{x_{\ell,j}\}_{j \in \mathcal{G}_\ell^n}.$$

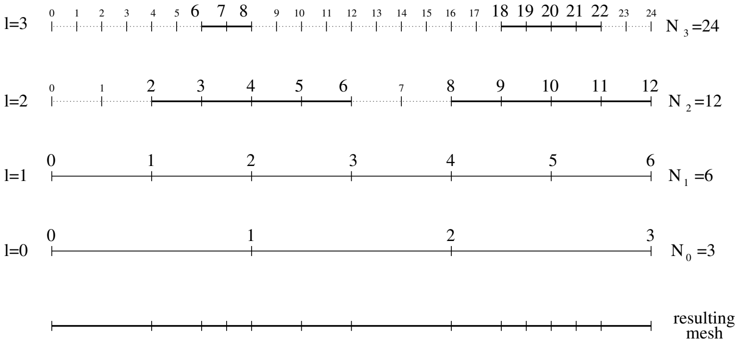
We are interested in

$$\mathcal{G}_\ell^n \subseteq \prod_{i=1}^{N} \{0, \dots, N_{i,\ell}\}.$$

# Grid hierarchy

## Nesting condition

We are interested in *nested* meshes:



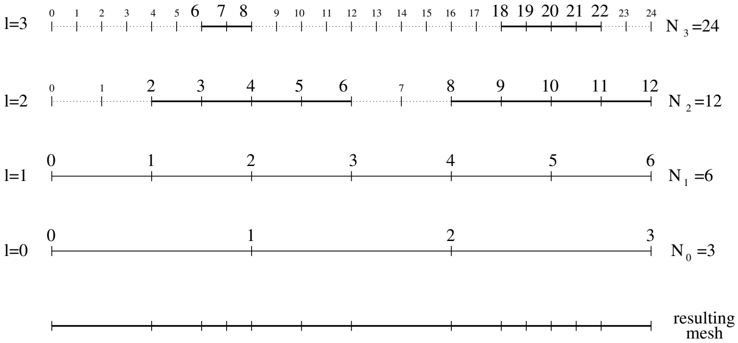

## Ghost points

Ghost points are added outside the $\ell$-grids to take into account the boundary conditions for the time integration.

# Grid hierarchy

## Nesting condition

We are interested in *nested* meshes:




## Ghost points

Ghost points are added outside the $\ell$-grids to take into account the boundary conditions for the time integration.

# Grid selection

The grid selection is based on two criteria: *prediction* and *gradient*.

### The *prediction* criterion

Let $\{u_{\ell,j}\}_{j\in\mathcal{G}_\ell^n}$ the point values at reslution level $\ell$. Let $p[u_\ell]$ an interpolator at resolution level $\ell$. We keep point $x_{\ell,j}$ for refinement if

$$|p[u_{\ell-1}](x_{\ell,j}) - u_{\ell,j}| > \tau_p.$$

### The *gradient* criterion

We estimate the gradient at point $x_{\ell,j}$ by means of the discrete gradient computed at resolution level $\ell - 1$: if it is over a certain tolerance parameter $\tau_{d,\ell-1}$, then point $x_{\ell,j}$ is selected for refinement.

### Reconstruction

Once the grid has been selected, reconstruct by means of an interpolator the point values that are not assigned yet.

# Grid selection

The grid selection is based on two criteria: *prediction* and *gradient*.

### The *prediction* criterion

Let $\{u_{\ell,j}\}_{j\in\mathcal{G}_\ell^n}$ the point values at reslution level $\ell$. Let $p[u_\ell]$ an interpolator at resolution level $\ell$. We keep point $x_{\ell,j}$ for refinement if

$$|p[u_{\ell-1}](x_{\ell,j}) - u_{\ell,j}| > \tau_p.$$

### The *gradient* criterion

We estimate the gradient at point $x_{\ell,j}$ by means of the discrete gradient computed at resolution level $\ell-1$: if it is over a certain tolerance parameter $\tau_{d,\ell-1}$, then point $x_{\ell,j}$ is selected for refinement.

### Reconstruction

Once the grid has been selected, reconstruct by means of an interpolator the point values that are not assigned yet.

# Grid selection

The grid selection is based on two criteria: *prediction* and *gradient*.

## The *prediction* criterion

Let $\{u_{\ell,j}\}_{j\in\mathcal{G}_\ell^n}$ the point values at reslution level $\ell$. Let $p[u_\ell]$ an interpolator at resolution level $\ell$. We keep point $x_{\ell,j}$ for refinement if

$$|p[u_{\ell-1}](x_{\ell,j}) - u_{\ell,j}| > \tau_p.$$

## The *gradient* criterion

We estimate the gradient at point $x_{\ell,j}$ by means of the discrete gradient computed at resolution level $\ell-1$: if it is over a certain tolerance parameter $\tau_{d,\ell-1}$, then point $x_{\ell,j}$ is selected for refinement.

## Reconstruction

Once the grid has been selected, reconstruct by means of an interpolator the point values that are not assigned yet.

# Grid selection

The grid selection is based on two criteria: *prediction* and *gradient*.

## The *prediction* criterion

Let $\{u_{\ell,j}\}_{j\in\mathcal{G}_\ell^n}$ the point values at reslution level $\ell$. Let $p[u_\ell]$ an interpolator at resolution level $\ell$. We keep point $x_{\ell,j}$ for refinement if

$$|p[u_{\ell-1}](x_{\ell,j}) - u_{\ell,j}| > \tau_p.$$

## The *gradient* criterion

We estimate the gradient at point $x_{\ell,j}$ by means of the discrete gradient computed at resolution level $\ell - 1$: if it is over a certain tolerance parameter $\tau_{d,\ell-1}$, then point $x_{\ell,j}$ is selected for refinement.

## Reconstruction

Once the grid has been selected, reconstruct by means of an interpolator the point values that are not assigned yet.

# Outline

1 Introduction

2 Numerical tools
- Multiresolution framework
- Time integration

3 Experiments
- Introduction
- 1D tests
- 2D tests

# 1D semi-Lagrangian strategy

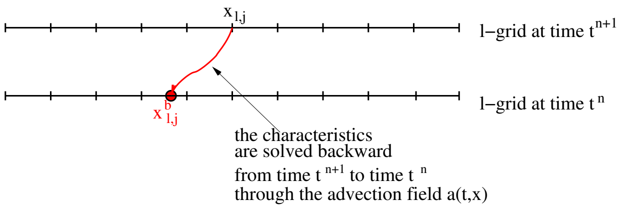
## Characteristic-based solution

The solution to the PDE $$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}(a(t,x)u) = 0, \qquad u(t=0,x) = u^0(x)$$

is given by $$u(t,x) = u\left(s, \mathcal{X}(s;t,x)\right) J(s;t,x),$$

with $\mathcal{X}(s;t,x)$ the characteristic at time $s$, starting from $x$ at time $t$:

$$\frac{\mathrm{d}\mathcal{X}(s;t,x)}{\mathrm{d}s} = a\left(s, \mathcal{X}(s;t,x)\right), \qquad \mathcal{X}(t;t,x) = x, J(s;t,x) := \frac{\partial \mathcal{X}(s;t,x)}{\partial x}.$$



# 1D semi-Lagrangian strategy

## Constant-coefficient advection

If $a$ is a real constant, then the solution of the characteristics is trivial

$$\mathcal{X}(s;t,x) = x + a\cdot(s-t)$$

and

$$J(s;t,x) := 1.$$

## Error estimate

The local truncation error can be estimated

$$E = \mathcal{O}\left(\Delta x_\ell^2\right) + \mathcal{O}\left(\Delta t^{s+1}\right),$$

where $s$ is the order of the integrator used to solve the characteristics (for example, Runge-Kutta). If the characteristics are solved exactly, then no order in time appears.

# 1D semi-Lagrangian strategy

## Constant-coefficient advection

If $a$ is a real constant, then the solution of the characteristics is trivial

$$\mathcal{X}(s;t,x) = x + a \cdot (s - t)$$

and

$$J(s;t,x) := 1.$$

## Error estimate

The local truncation error can be estimated

$$E = \mathcal{O}\left(\Delta x_\ell^2\right) + \mathcal{O}\left(\Delta t^{s+1}\right),$$

where $s$ is the order of the integrator used to solve the characteristics (for example, Runge-Kutta). If the characteristics are solved exactly, then no order in time appears.

# The 2D case

## Grid hierarchy and selection

We do not give details, but we apply strategies similar to the 1D case.

## The 2D PDE

We solve the 2D PDE $$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x_1}(a_1 u) + \frac{\partial}{\partial x_2}(a_2 u) = 0$$

by splitting the $(x_1, x_2)$-domain thanks to the second-order Strang scheme:

- Solve for a $\frac{\Delta t}{2}$ time step $\frac{\partial u}{\partial t} + \frac{\partial}{\partial x_1}(a_1 u) = 0$;
- Solve for a $\Delta t$ time step $\frac{\partial u}{\partial t} + \frac{\partial}{\partial x_2}(a_2 u) = 0$;
- Solve for a $\frac{\Delta t}{2}$ time step $\frac{\partial u}{\partial t} + \frac{\partial}{\partial x_1}(a_1 u) = 0$.

## Error estimate

The Strang splitting constrains the accuracy:

$$E = \mathcal{O}(\Delta x_{1,\ell}^2) + \mathcal{O}(\Delta x_{2,\ell}^2) + \mathcal{O}\left(\Delta t^{\min(s+1,3)}\right).$$

# The 2D case

## Grid hierarchy and selection

We do not give details, but we apply strategies similar to the 1D case.

## The 2D PDE

We solve the 2D PDE
$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x_1}(a_1 u) + \frac{\partial}{\partial x_2}(a_2 u) = 0$$

by splitting the $(x_1, x_2)$-domain thanks to the second-order Strang scheme:

- Solve for a $\frac{\Delta t}{2}$ time step $\frac{\partial u}{\partial t} + \frac{\partial}{\partial x_1}(a_1 u) = 0$;
- Solve for a $\Delta t$ time step $\frac{\partial u}{\partial t} + \frac{\partial}{\partial x_2}(a_2 u) = 0$;
- Solve for a $\frac{\Delta t}{2}$ time step $\frac{\partial u}{\partial t} + \frac{\partial}{\partial x_1}(a_1 u) = 0$.

## Error estimate

The Strang splitting constrains the accuracy:

$$E = \mathcal{O}(\Delta x_{1,\ell}^2) + \mathcal{O}(\Delta x_{2,\ell}^2) + \mathcal{O}\left(\Delta t^{\min(s+1,3)}\right).$$

# The 2D case

## Grid hierarchy and selection

We do not give details, but we apply strategies similar to the 1D case.

## The 2D PDE

We solve the 2D PDE
$$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x_1}(a_1 u) + \frac{\partial}{\partial x_2}(a_2 u) = 0$$

by splitting the $(x_1, x_2)$-domain thanks to the second-order Strang scheme:

- Solve for a $\frac{\Delta t}{2}$ time step $\frac{\partial u}{\partial t} + \frac{\partial}{\partial x_1}(a_1 u) = 0$;
- Solve for a $\Delta t$ time step $\frac{\partial u}{\partial t} + \frac{\partial}{\partial x_2}(a_2 u) = 0$;
- Solve for a $\frac{\Delta t}{2}$ time step $\frac{\partial u}{\partial t} + \frac{\partial}{\partial x_1}(a_1 u) = 0$.

## Error estimate

The Strang splitting constrains the accuracy:

$$E = \mathcal{O}(\Delta x_{1,\ell}^2) + \mathcal{O}(\Delta x_{2,\ell}^2) + \mathcal{O}\left(\Delta t^{\min(s+1,3)}\right).$$

# Outline

1 Introduction

2 Numerical tools
- Multiresolution framework
- Time integration

3 Experiments
- Introduction
- 1D tests
- 2D tests

# Goals

The Adaptive-Mesh-Refinement (AMR) framework is compared to the equivalent Fixed-Mesh (FM) results.

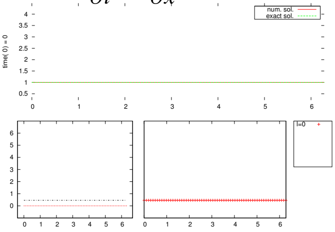
Of course, AMR cannot be more accurate than FM. Rather, it achieves faster computational times in exchange of a loss of precision.

# Outline

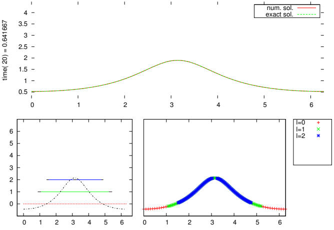
1. Introduction
2. Numerical tools
   - Multiresolution framework
   - Time integration
3. Experiments
   - Introduction
   - 1D tests
   - 2D tests

# Variable-coefficient advection

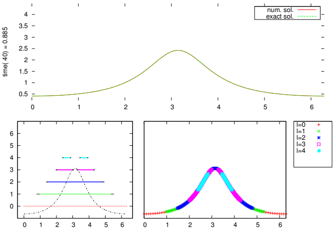
The test case $$\frac{\partial u}{\partial t} + \frac{\partial}{\partial x}\left(\sin(x)\, u\right) = 0$$ produces a blow-up.



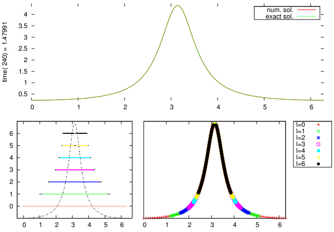

(a) at time 0




(b) at time 0.65




(c) at time 0.89




(d) at time 1.48

# Variable-coefficient advection

## Speedup

For parameters

$$
\begin{aligned}
N_0 &= 128 && \text{points at } \ell = 0 \\
L &= 4 && \text{number of resolution levels} \\
\Delta t_0 &= 0.125 && \text{maximum time step} \\
\tau_p &= 10^{-4} && \textit{prediction}\text{-criterion tolerance} \\
\tau_{d,0} &= 0.5 && \textit{gradient}\text{-criterion tolerance,}
\end{aligned}
$$

AMR reaches a speedup of 35 times with respect to the equivalently-resolved FM, with a loss of precision from $10^{-9}$ to roughly $10^{-6}$ (the $L^2$-error w.r.t. the analytical solution).

# Outline

1 Introduction

2 Numerical tools
- Multiresolution framework
- Time integration

3 Experiments
- Introduction
- 1D tests
- 2D tests

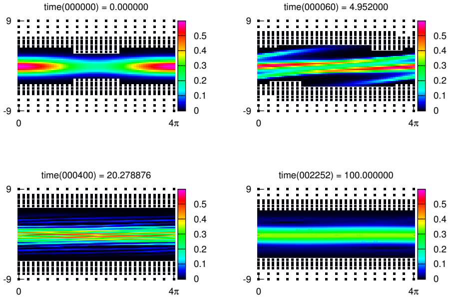
# Landau damping

## Vlasov-Poisson

The system reads $$\frac{\partial f}{\partial t} + v\frac{\partial f}{\partial x} + E\frac{\partial f}{\partial v} = 0, \quad \frac{\partial E}{\partial x} = 1 - \int_{\mathbb{R}} f(t,x,v)\,\mathrm{d}v$$

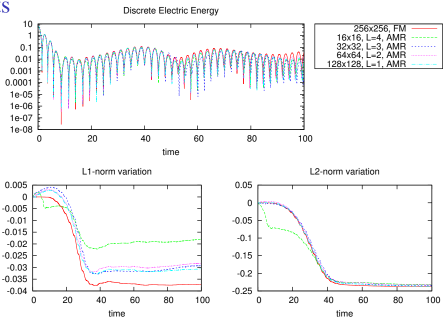
completed by periodic b.c. The Landau damping is

$$f^0(x,v) = \frac{e^{-\frac{v^2}{2}}}{\sqrt{2\pi}}\left(1 + 0.5\cdot\cos(0.5\cdot x)\right), \quad \Omega = \left[0, \frac{2\pi}{0.5}\right] \times [-9, 9].$$



# Landau damping

## Quality of the results



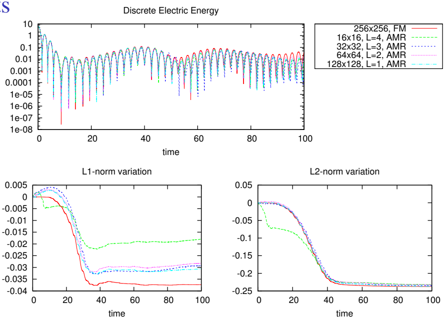

## Speedup

We fix parameters $N_{1,0} = N_{2,0} = 32,\ L = 3,\ \tau_{d,0} = 0.5.$ The speedups:

| | $\tau_p = 10^{-4}$ | $\tau_p = 10^{-8}$ | $\tau_p = 10^{-12}$ |
|---|---|---|---|
| speedup | 2.9 | 1.4 | 0.9 |

# Landau damping

## Quality of the results




## Speedup

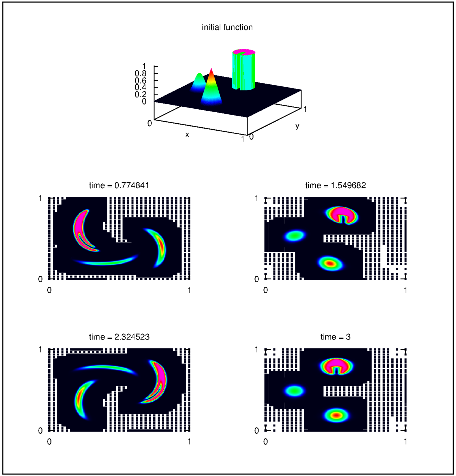
We fix parameters $N_{1,0} = N_{2,0} = 32,\ L = 3,\ \tau_{d,0} = 0.5.$ The speedups:

| | $\tau_p = 10^{-4}$ | $\tau_p = 10^{-8}$ | $\tau_p = 10^{-12}$ |
|---|---|---|---|
| speedup | 2.9 | 1.4 | 0.9 |

# Deformation flows

The system

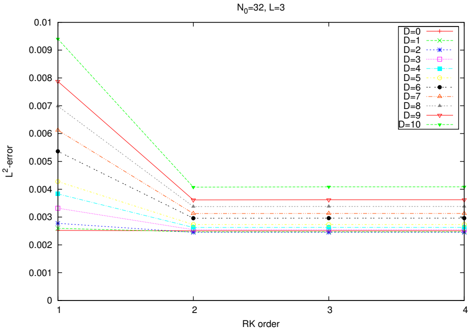
$$\frac{\partial f}{\partial t}+\frac{\partial}{\partial x}\left[\sin^2(\pi x)\sin(2\pi y)g(t)f\right]+\frac{\partial}{\partial y}\left[-\sin^2(\pi y)\sin(2\pi x)g(t)f\right]=0,\quad (x,y)\in[0,1]^2,$$




for $g(t)=\cos\left(\frac{\pi t}{T}\right)$, periodically recovers the initial datum after alternate clockwise and counterclockwise twistings.

# Deformation flows

## The ODE integrator for the characteristics




As announced by the error estimate

$$E = \mathcal{O}(\Delta x_{1,\ell}^2) + \mathcal{O}(\Delta x_{2,\ell}^2) + \mathcal{O}\left(\Delta t^{\min(s+1,3)}\right)$$

the Strang-splitting order constrains the accuracy.

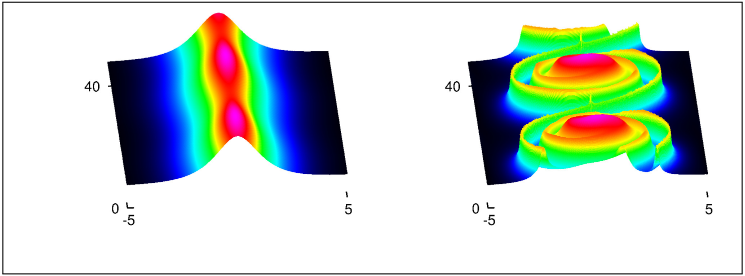
# Kelvin-Hemlholtz instabilities

## The model

The *guiding-center* model (omitting some details)

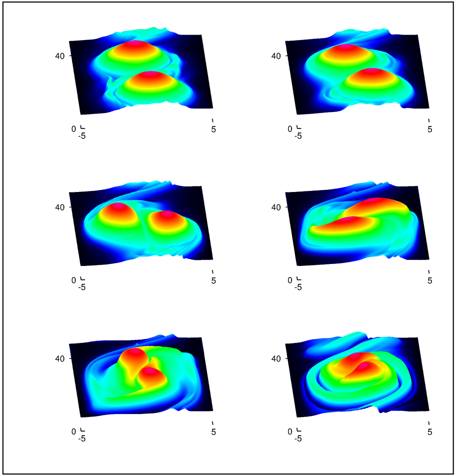
$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_1}\left[\frac{\partial \Phi}{\partial x_2}\rho\right] + \frac{\partial}{\partial x_2}\left[-\frac{\partial \Phi}{\partial x_1}\rho\right] = 0, \qquad \Delta_{x_1,x_2}\Phi = \rho,$$

for initial condition $\rho(0, x_1, x_2) = 1.5 \operatorname{sech}\left(\frac{x_2}{0.9}\right) \cdot (1 + 0.08 \sin(2kx_1))$,
periodic $x_1$- and Dirichlet $x_2$-boundaries, produces vortices and filamentation.

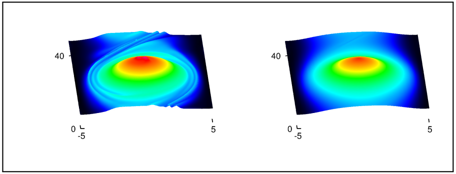



(i) first instability

# Kelvin-Hemlholtz instabilities




(j) second instability




(k) steady state

SPEEDUP

The AMR strategy achieves a speedup of about 2.8, for parameters $(N_{\star,0}, L) = (32, 4)$ (the maximum resolution is $512 \times 512$), $\tau_p = 10^{-3}$, $\tau_{d,0} = 1.5$.

# THANK YOU!
# GRACIAS!
# GRAZIE!
# MERCI!