



HAL
open science

Guiding center simulations on curvilinear grids

Adnane Hamiaz, Michel Mehrenberger

► **To cite this version:**

Adnane Hamiaz, Michel Mehrenberger. Guiding center simulations on curvilinear grids. 2013. hal-00908500v1

HAL Id: hal-00908500

<https://hal.science/hal-00908500v1>

Submitted on 23 Nov 2013 (v1), last revised 8 Oct 2015 (v3)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Guiding center simulations on curvilinear grids

A. Hamiaz¹, M. Mehrenberger^{1,2}

¹ INRIA-Nancy & IRMA, Strasbourg, France

² Max-Planck-Institut für Plasmaphysik, Garching, Germany

November 19, 2013

Abstract

We consider a guiding center simulation on curvilinear grids. The code uses a classical semi-Lagrangian scheme for the advection [5] and a curvilinear Poisson solver developed by A. Back [1]. We report here some numerical results.

1 Introduction

In this report, we give some description for dealing with semi-Lagrangian schemes on curvilinear grids and give some numerical results, by using a classical method [5] for the advective part and a new solver developed by A. Back for the Poisson equation [1]. Thanks to this solver, we need not to project the Poisson solver on a cartesian grid, as it is done in [2] for the specific case of an oblic transformation. This study is part of a general work on adding curvilinear capabilities for the simulation of drift kinetic and gyrokinetic equations in a semi-Lagrangian framework, and is in current development in a ongoing library [4]. The work is devoted to the case of the guiding center equation, which remains $2D$, and thus permits to perform more easily several tests, making vary the numerical parameters. Curvilinear grids are important in order to be able to deal with specific geometries and also for adapting the grid to save computational effort. To see the robustness of the numerical method, we test the method on some Colella grids [6],[7], which are sinusoidal perturbations of cartesian grids, and we consider also the case of a polar geometry; the same algorithm is used for both geometries.

2 2D Guiding-center equation on curvilinear mesh

2.1 Mapping

We denote $\Omega \in \mathbf{R}^2$ the computational domain in the phase space. The curvilinear coordinates system, that we will call mapped mesh in the discrete case, is defined by a mapping \mathcal{F} such that:

$$\mathcal{F} : Q = [\eta_{1_{min}}, \eta_{1_{max}}] \times [\eta_{2_{min}}, \eta_{2_{max}}] \rightarrow \Omega \quad (2.1)$$

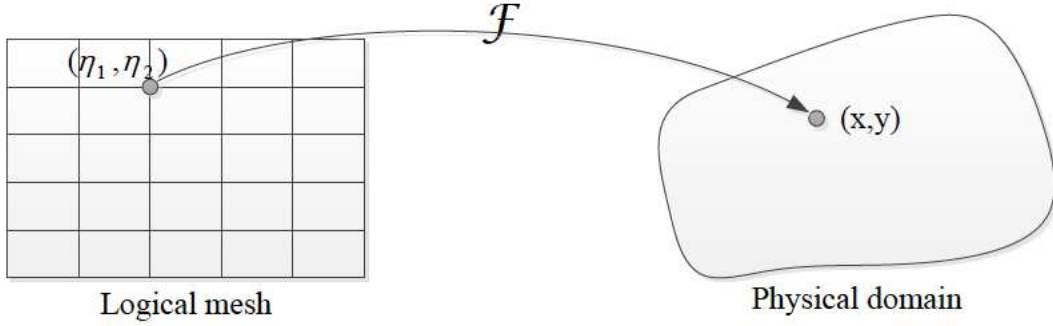


Figure 1: Representation of a mapped mesh in two dimensions

$$\begin{bmatrix} \eta_1 \\ \eta_2 \end{bmatrix} \rightarrow \begin{bmatrix} x(\eta_1, \eta_2) \\ y(\eta_1, \eta_2) \end{bmatrix}$$

This application could be defined piecewise. We consider in this paper only one mapping. When there is an analytical expression for the mapping, working on the mapped mesh is the same as working with the set of coordinates defined by the curvilinear coordinate system. The idea of the mapped mesh can be seen as a discrete representation of a curvilinear coordinate system, an interpolation method defining the mapping for points that are not on the grid.

We can define the Jacobian matrix of the transformation:

$$D\mathcal{F}(\eta_1, \eta_2) = \begin{pmatrix} \frac{\partial x}{\partial \eta_1} & \frac{\partial x}{\partial \eta_2} \\ \frac{\partial y}{\partial \eta_1} & \frac{\partial y}{\partial \eta_2} \end{pmatrix}$$

and we denote \sqrt{g} the jacobian of the matrix $D\mathcal{F}$:

$$\sqrt{g}(\eta_1, \eta_2) = \text{Det}(D\mathcal{F}) = \frac{\partial x}{\partial \eta_1} \frac{\partial y}{\partial \eta_2} - \frac{\partial x}{\partial \eta_2} \frac{\partial y}{\partial \eta_1}$$

also we can write the jacobian matrix:

$$D\mathcal{F} = (D\mathcal{F}^{-1})^{-1} = \begin{pmatrix} \frac{\partial \eta_1}{\partial x} & \frac{\partial \eta_1}{\partial y} \\ \frac{\partial \eta_2}{\partial x} & \frac{\partial \eta_2}{\partial y} \end{pmatrix}^{-1} = \sqrt{g} \begin{pmatrix} \frac{\partial \eta_2}{\partial y} & -\frac{\partial \eta_1}{\partial y} \\ -\frac{\partial \eta_2}{\partial x} & \frac{\partial \eta_1}{\partial x} \end{pmatrix} \quad (2.2)$$

Given the mapping \mathcal{F} , the gradient of a scalar ($\nabla_{(x,y)}$) and the divergence of a vector ($\nabla_{(x,y)} \cdot$) in physical space can be written in terms of derivatives in mapping space.

Lemma 2.1. *Let $u(x, y)$ be a scalar and we note $\tilde{u}(\eta_1, \eta_2) = u(x(\eta_1, \eta_2), y(\eta_1, \eta_2))$, then:*

$$\nabla_{(x,y)} u = D\mathcal{F}^{-t} \nabla_{(\eta_1, \eta_2)} \tilde{u} \quad (2.3)$$

that is equivalent to:

1. the non conservative form:

$$\nabla_{(x,y)} u = \sum_{i=1}^2 \nabla \eta_i \partial_{\eta_i} \tilde{u} \quad (2.4)$$

2. the conservative form:

$$\nabla_{(x,y)} u = \frac{1}{\sqrt{g}} \sum_{i=1}^2 \partial_{\eta_i} (\sqrt{g} \nabla \eta_i \tilde{u}) \quad (2.5)$$

Proof 2.1. We have

$$\partial_{\eta_1} \tilde{u} = \partial_{\eta_1} x \partial_x u + \partial_{\eta_1} y \partial_y u, \quad \partial_{\eta_2} \tilde{u} = \partial_{\eta_2} x \partial_x u + \partial_{\eta_2} y \partial_y u,$$

which leads to

$$\begin{bmatrix} \partial_{\eta_1} \tilde{u} \\ \partial_{\eta_2} \tilde{u} \end{bmatrix} = \begin{bmatrix} \partial_{\eta_1} x & \partial_{\eta_1} y \\ \partial_{\eta_2} x & \partial_{\eta_2} y \end{bmatrix} \begin{bmatrix} \partial_x u \\ \partial_y u \end{bmatrix}$$

This leads to

$$\nabla_{(\eta_1, \eta_2)} \tilde{u} = D\mathcal{F}^t \nabla_{(x,y)} u.$$

To prove the first point we used (2.2) and we have

$$\nabla_{(x,y)} u = \begin{pmatrix} \frac{\partial \eta_1}{\partial x} & \frac{\partial \eta_1}{\partial y} \\ \frac{\partial \eta_2}{\partial x} & \frac{\partial \eta_2}{\partial y} \end{pmatrix}^t \begin{bmatrix} \partial_{\eta_1} \tilde{u} \\ \partial_{\eta_2} \tilde{u} \end{bmatrix} = [\nabla \eta_1 \quad \nabla \eta_2] \begin{bmatrix} \partial_{\eta_1} \tilde{u} \\ \partial_{\eta_2} \tilde{u} \end{bmatrix} = \nabla \eta_1 \partial_{\eta_1} \tilde{u} + \nabla \eta_2 \partial_{\eta_2} \tilde{u}$$

For the second point, we have

$$\frac{1}{\sqrt{g}} \sum_{i=1}^2 \partial_{\eta_i} (\sqrt{g} \nabla \eta_i \tilde{u}) = \frac{1}{\sqrt{g}} \sum_{i=1}^2 \partial_{\eta_i} (\sqrt{g} \nabla \eta_i) \tilde{u} + \sum_{i=1}^2 \nabla \eta_i \partial_{\eta_i} \tilde{u} = \frac{\tilde{u}}{\sqrt{g}} \sum_{i=1}^2 \partial_{\eta_i} (\sqrt{g} \nabla \eta_i) + \nabla_{(x,y)} u$$

it suffices to show that

$$\sum_{i=1}^2 \partial_{\eta_i} (\sqrt{g} \nabla \eta_i) = 0$$

by expanding the left side

$$\begin{aligned} \sum_{i=1}^2 \partial_{\eta_i} (\sqrt{g} \nabla \eta_i) &= \partial_{\eta_1} (\sqrt{g}) \nabla \eta_1 + \partial_{\eta_2} (\sqrt{g}) \nabla \eta_2 + \sqrt{g} (\partial_{\eta_1} \nabla \eta_1 + \partial_{\eta_2} \nabla \eta_2) \\ &= \begin{bmatrix} \partial_{\eta_1} (\sqrt{g}) \partial_x \eta_1 + \partial_{\eta_2} (\sqrt{g}) \partial_x \eta_2 + \sqrt{g} (\partial_{\eta_1} \partial_x \eta_1 + \partial_{\eta_2} \partial_x \eta_2) \\ \partial_{\eta_1} (\sqrt{g}) \partial_y \eta_1 + \partial_{\eta_2} (\sqrt{g}) \partial_y \eta_2 + \sqrt{g} (\partial_{\eta_1} \partial_y \eta_1 + \partial_{\eta_2} \partial_y \eta_2) \end{bmatrix} \end{aligned}$$

using 2.2 and 2.1 we get

$$\begin{aligned} \sum_{i=1}^2 \partial_{\eta_i} (\sqrt{g} \nabla \eta_i) &= \begin{bmatrix} \partial_{\eta_1} (\sqrt{g}) \frac{\partial_{\eta_2} y}{\sqrt{g}} - \partial_{\eta_2} (\sqrt{g}) \frac{\partial_{\eta_1} y}{\sqrt{g}} + \sqrt{g} (\partial_{\eta_1} \frac{\partial_{\eta_2} y}{\sqrt{g}} + \partial_{\eta_2} \frac{-\partial_{\eta_1} y}{\sqrt{g}}) \\ \partial_{\eta_1} (\sqrt{g}) \frac{-\partial_{\eta_2} x}{\sqrt{g}} + \partial_{\eta_2} (\sqrt{g}) \frac{\partial_{\eta_1} x}{\sqrt{g}} + \sqrt{g} (\partial_{\eta_1} \frac{-\partial_{\eta_2} x}{\sqrt{g}} + \partial_{\eta_2} \frac{\partial_{\eta_1} x}{\sqrt{g}}) \end{bmatrix} \\ &= \begin{bmatrix} \partial_{\eta_1} (\sqrt{g}) \frac{\partial_{\eta_2} y}{\sqrt{g}} - \partial_{\eta_2} (\sqrt{g}) \frac{\partial_{\eta_1} y}{\sqrt{g}} + \partial_{\eta_1} \partial_{\eta_2} y - \partial_{\eta_1} (\sqrt{g}) \frac{\partial_{\eta_2} y}{\sqrt{g}} - \partial_{\eta_1} \partial_{\eta_2} y + \partial_{\eta_2} (\sqrt{g}) \frac{\partial_{\eta_1} y}{\sqrt{g}} \\ \partial_{\eta_1} (\sqrt{g}) \frac{-\partial_{\eta_2} x}{\sqrt{g}} + \partial_{\eta_2} (\sqrt{g}) \frac{\partial_{\eta_1} x}{\sqrt{g}} - \partial_{\eta_1} \partial_{\eta_2} x + \partial_{\eta_1} (\sqrt{g}) \frac{\partial_{\eta_2} x}{\sqrt{g}} + \partial_{\eta_1} \partial_{\eta_2} x - \partial_{\eta_2} (\sqrt{g}) \frac{\partial_{\eta_1} x}{\sqrt{g}} \end{bmatrix} = 0 \end{aligned}$$

thus we have demonstrated 2.5.

By using (2.5), the divergence of a vector v can be expressed as

$$\nabla_{(x,y)} \cdot v = \frac{1}{\sqrt{g}} \sum_{i=1}^2 \partial_{\eta_i} (\sqrt{g} \nabla \eta_i \cdot \tilde{v}). \quad (2.6)$$

2.2 2D Guiding-center model

The guiding-center model does not fit into the framework of linear transport. Unknown in the guiding-center satisfies the equation:

$$\frac{\partial f}{\partial t} + a_1(x, y)\partial_x f(t, x, y) + a_2(x, y)\partial_y f(t, x, y) = 0 \quad (2.7)$$

with

$$\partial_x a_1(t, x, y) + \partial_y a_2(t, x, y) = 0$$

Equation 2.7 can be written in the phase space (x, y) as standard advective:

$$\frac{\partial f}{\partial t} + \mathbf{A}(x, y) \cdot \nabla_{(x,y)} f = 0 \quad (2.8)$$

$$\text{div}_{(x,y)}(\mathbf{A}) = 0,$$

where $\mathbf{A} = (a_1, a_2)^t$ and since the divergence of \mathbf{A} is zero then there is as a scalar potential Φ such that:

$$\mathbf{A}(x, y) = \mathbf{rot} \Phi(x, y).$$

In other words, we have

$$\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \partial_y \Phi \\ -\partial_x \Phi \end{bmatrix}.$$

We can write this equation in curvilinear coordinates (η_1, η_2) using transformation \mathcal{F} , using $\nabla_{(x,y)} f = D\mathcal{F}^{-t} \nabla_{(\eta_1, \eta_2)} \tilde{f}$ we have:

$$\begin{aligned} \mathbf{A} \cdot \nabla_{(x,y)} f &= \mathbf{A} \cdot (D\mathcal{F}^{-t} \nabla_{(\eta_1, \eta_2)} \tilde{f}) \\ &= \mathbf{A}^t (D\mathcal{F}^{-t} \nabla_{(\eta_1, \eta_2)} \tilde{f}) \\ &= (D\mathcal{F}^{-1} \mathbf{A})^t \nabla_{(\eta_1, \eta_2)} \tilde{f} \\ &= \tilde{\mathbf{A}} \cdot \nabla_{(\eta_1, \eta_2)} \tilde{f} \end{aligned}$$

We denote :

$$\tilde{f}(t, \eta_1, \eta_2) = f(t, x(\eta_1, \eta_2), y(\eta_1, \eta_2))$$

$$\tilde{\mathbf{A}}(t, \eta_1, \eta_2) = (D\mathcal{F})^{-1} \mathbf{A}(t, x, y) = \begin{bmatrix} \nabla_{(x,y)} \eta_1 \cdot \mathbf{A} \\ \nabla_{(x,y)} \eta_2 \cdot \mathbf{A} \end{bmatrix} \quad (2.9)$$

Applying formulae (2.9, 2.2), we can calculate $\tilde{\mathbf{A}}$:

$$\tilde{\mathbf{A}}(t, \eta_1, \eta_2) = \frac{1}{\sqrt{g}} \begin{bmatrix} \partial_{\eta_2} \tilde{\Phi} \\ -\partial_{\eta_1} \tilde{\Phi} \end{bmatrix}$$

with

$$\tilde{\Phi}(\eta_1, \eta_2) = \Phi(x(\eta_1, \eta_2), y(\eta_1, \eta_2))$$

The equation of the guiding-center can be written in the curvilinear coordinates (η_1, η_2) as :

$$\frac{\partial \tilde{f}}{\partial t} + \frac{\partial_{\eta_2} \tilde{\Phi}}{\sqrt{g}} \partial_{\eta_1} \tilde{f} - \frac{\partial_{\eta_1} \tilde{\Phi}}{\sqrt{g}} \partial_{\eta_2} \tilde{f} = 0 \quad (2.10)$$

2.3 Poisson curvilinear coordinates

On the cartesian coordinates, the Poisson equation has the form:

$$-\nabla \cdot (b(x, y)\nabla\Phi(x, y)) + c(x, y)\Phi(x, y) = f(x, y) \quad (2.11)$$

where f is the distribution function given by solving the Vlasov equation. Φ is the electric potential. b and c are scalar functions.

We want to write the Poisson equation on curvilinear coordinates thanks to the transformation \mathcal{F} defined in (2.1).

Using the expression of the gradient (2.5) and the divergence of vector (2.6) in curvilinear coordinates we have:

$$\nabla \cdot b(x, y)\nabla\Phi(x, y) = \frac{1}{\sqrt{g}} \sum_{i=1}^2 \sum_{j=1}^2 \partial_{\eta_i} (\tilde{b}(\eta_1, \eta_2)\sqrt{g}g^{ij}\partial_{\eta_j}\tilde{\Phi}(\eta_1, \eta_2)), \quad (2.12)$$

where g^{ij} is the contravariant components of the metric tensor G witch are the dot products of the contravariant basis vectors:

$$g^{ij} = \nabla_{(x,y)}\eta_i \cdot \nabla_{(x,y)}\eta_j, \quad i, j \in \{1, 2\}$$

We note that G is symmetric and

$$\det(G) = \frac{1}{g}.$$

By (2.14), the new Poisson equation on curvilinear coordinates has the form:

$$-\sum_{i=1}^2 \sum_{j=1}^2 \partial_{\eta_i} (\tilde{b}(\eta_1, \eta_2)\sqrt{g}g^{ij}\partial_{\eta_j}\tilde{\Phi}(\eta_1, \eta_2)) + \sqrt{g}\tilde{c}(\eta_1, \eta_2)\tilde{\Phi}(\eta_1, \eta_2) = \sqrt{g}\tilde{f}(\eta_1, \eta_2) \quad (2.13)$$

By developing the left hand side of (2.13) we have :

$$-\partial_{\eta_1} [\tilde{b}(\eta_1, \eta_2)\sqrt{g}(g^{11}\partial_{\eta_1}\tilde{\Phi}(\eta_1, \eta_2)+g^{12}\partial_{\eta_2}\tilde{\Phi}(\eta_1, \eta_2))] - \partial_{\eta_2} [\tilde{b}(\eta_1, \eta_2)\sqrt{g}(g^{21}\partial_{\eta_1}\tilde{\Phi}(\eta_1, \eta_2)+g^{22}\partial_{\eta_2}\tilde{\Phi}(\eta_1, \eta_2))] + \sqrt{g}\tilde{c}(\eta_1, \eta_2)\tilde{\Phi}(\eta_1, \eta_2) = \sqrt{g}\tilde{f}(\eta_1, \eta_2)$$

and

$$-\partial_{\eta_1} [\tilde{b}(\eta_1, \eta_2)\sqrt{g} \begin{pmatrix} g^{11} \\ g^{12} \end{pmatrix}^t \cdot \nabla_{\eta_1, \eta_2} \tilde{\Phi}(\eta_1, \eta_2)] - \partial_{\eta_2} [\tilde{b}(\eta_1, \eta_2)\sqrt{g} \begin{pmatrix} g^{21} \\ g^{22} \end{pmatrix}^t \cdot \nabla_{\eta_1, \eta_2} \tilde{\Phi}(\eta_1, \eta_2)] + \sqrt{g}\tilde{c}(\eta_1, \eta_2)\tilde{\Phi}(\eta_1, \eta_2) = \sqrt{g}\tilde{f}(\eta_1, \eta_2)$$

Finally, (2.14) has the form:

$$-\nabla_{\eta_1, \eta_2} \cdot (\sqrt{g}\tilde{b}(\eta_1, \eta_2)G \nabla_{\eta_1, \eta_2} \tilde{\Phi}(\eta_1, \eta_2)) + \tilde{c}(\eta_1, \eta_2)\tilde{\Phi}(\eta_1, \eta_2) = \sqrt{g}\tilde{f}(\eta_1, \eta_2) \quad (2.14)$$

where

$$G = \begin{pmatrix} g^{11} & g^{12} \\ g^{21} & g^{22} \end{pmatrix} = \begin{pmatrix} \nabla_{(x,y)}\eta_1 \cdot \nabla_{(x,y)}\eta_1 & \nabla_{(x,y)}\eta_1 \cdot \nabla_{(x,y)}\eta_2 \\ \nabla_{(x,y)}\eta_2 \cdot \nabla_{(x,y)}\eta_1 & \nabla_{(x,y)}\eta_2 \cdot \nabla_{(x,y)}\eta_2 \end{pmatrix}.$$

3 Numerical method

3.1 The Semi-Lagrangian method for Guiding center equation

According to the study in the previous section, the equation of the guiding-center on a general mesh can be written in the form (2.10). Characteristics associated with this equation are given by the following system of differential equations:

$$\frac{\partial \eta_1(t)}{\partial t} = \frac{\partial_{\eta_2} \tilde{\Phi}(t, (\eta_1(t), \eta_2(t)))}{\sqrt{g}(\eta_1(t), \eta_2(t))} \quad (3.15)$$

$$\frac{\partial \eta_2(t)}{\partial t} = -\frac{\partial_{\eta_1} \tilde{\Phi}(t, (\eta_1(t), \eta_2(t)))}{\sqrt{g}(\eta_1(t), \eta_2(t))} \quad (3.16)$$

$$\eta_1(s) = \eta^* \quad (3.17)$$

$$\eta_2(s) = \eta^{**} \quad (3.18)$$

Let us denote the solution of this system by:

$$\Gamma(t; \eta, s) = \begin{bmatrix} \eta_1(t; \eta^*, s) \\ \eta_2(t; \eta^{**}, s) \end{bmatrix}$$

The distribution function \tilde{f} is constant along the characteristic curves $(\eta_1(t), \eta_2(t))$

$$\tilde{f}(t, \Gamma(t; \eta, s)) = \tilde{f}(s, \Gamma(s; \eta, s)) = \tilde{f}(s, \eta) \quad \forall t, s, \eta = (\eta^*, \eta^{**}) \quad (3.19)$$

This property will be used to solve a discrete problem, which is defined by introducing a finite set of mesh point $(\eta_{ij})_{0 < i, j < N}$. By introducing a finite set of mesh point $(\eta_{ij})_{0 < i, j < N}$ and using the property (3.19), the classical semi-Lagrangian method, or backward (BSL: Backward Semi Lagrangian)[5] is divided into two steps to calculate the distribution function $\tilde{f}_{ij}^{n+1} = \tilde{f}(t^{n+1}, \eta_{ij})$ at time t^{n+1} from the distribution function $\tilde{f}_{ij}^n = \tilde{f}(t^n, \eta_{ij})$ at time t^n :
For each mesh point η_{ij}

1. Calculating $\Gamma(t^n; \eta_{ij}, t^{n+1})$ the value of the characteristic at t^n which is η_{ij} at t^{n+1} .
2. As the distribution function solution of the equation of guiding-centering check (3.19), we have:

$$\tilde{f}_{ij}^{n+1} = \tilde{f}^n(\Gamma(t^n; \eta_{ij}, t^{n+1}))$$

usually the point $\Gamma(t^n; \eta_{ij}, t^{n+1})$ is not a point of the logic gride. The value of \tilde{f}_{ij}^{n+1} is obtained by cubic interpolation (spline, Hermite, Lagrange) of the fonction $\tilde{f}(t^n, \cdot)$ at mesh points $\Gamma(t^n; \eta_{ij}, t^{n+1})$ at time t^n .

Computation of the origin of the characteristics

In order to deal with step 1, we need to introduce a time discretization of (3.18). The starting point of the characteristic is obtained by:

$$\frac{\Gamma^{n+1} - \Gamma^n}{\Delta t} = \frac{\eta_{ij} - \Gamma^n}{\Delta t} = \mathbf{A}^n(\Gamma^{n+1/2}) \quad (3.20)$$

with

$$\Gamma^{n+1/2} = \frac{\Gamma^{n+1} + \Gamma^n}{2} = \frac{\eta_{ij} + \Gamma^n}{2}$$

We denote \mathbf{d}_{ij} such that $\Gamma^n = \eta_{ij} - \mathbf{d}_{ij}$. Then (3.20) becomes:

$$\mathbf{d}_{ij} = \Delta t \mathbf{A}^n \left(\eta_{ij} - \frac{\mathbf{d}_{ij}}{2} \right) \quad (3.21)$$

which can be solved iteratively for unknown \mathbf{d}_{ij} .

Denoting by $(\alpha_{ij}, \beta_{ij})$ the coordinates of the displacement \mathbf{d}_{ij} at the mesh point $\eta_{ij} = (\eta_{1_i}, \eta_{2_j})$. Thus, Eq. (3.21) becomes:

$$\alpha_{ij}^{k+1} = \Delta t \mathbf{A}^n \left(\eta_{1_i} - \frac{\alpha_{ij}^k}{2}, \eta_{2_j} - \frac{\beta_{ij}^k}{2} \right) \quad (3.22)$$

$$\beta_{ij}^{k+1} = \Delta t \mathbf{A}^n \left(\eta_{1_i} - \frac{\alpha_{ij}^k}{2}, \eta_{2_j} - \frac{\beta_{ij}^k}{2} \right) \quad (3.23)$$

starting from an initial data $(\alpha_{ij}^0, \beta_{ij}^0)$.

Notice that \mathbf{A}^n , known only on the mesh, so needs to be interpolated at the point $(\eta_{1_i} - \frac{\alpha_{ij}^k}{2}, \eta_{2_j} - \frac{\beta_{ij}^k}{2})$. We use a linear interpolation which is sufficient for our approximation:

$$\mathbf{A}^n \left(\eta_{1_i} - \frac{\alpha_{ij}^k}{2}, \eta_{2_j} - \frac{\beta_{ij}^k}{2} \right) = \begin{bmatrix} (1-\lambda) & \lambda \end{bmatrix} \begin{pmatrix} \mathbf{A}^n(\eta_{1_i}, \eta_{2_j}) & \mathbf{A}^n(\eta_{1_i}, \eta_{2_{j+1}}) \\ \mathbf{A}^n(\eta_{1_{i+1}}, \eta_{2_j}) & \mathbf{A}^n(\eta_{1_{i+1}}, \eta_{2_{j+1}}) \end{pmatrix} \begin{bmatrix} 1-\theta \\ \theta \end{bmatrix} \quad (3.24)$$

with $(\lambda, \theta) \in [0, 1]^2$.

Computational algorithm

We define a uniform mesh of phase space whose coordinates are given by:

$$\eta_i = \eta_{1_{min}} + i \frac{\eta_{1_{max}} - \eta_{1_{min}}}{N_{\eta_1}} \quad , \quad \eta_j = \eta_{2_{min}} + j \frac{\eta_{2_{max}} - \eta_{2_{min}}}{N_{\eta_2}}$$

where N_{η_1}, N_{η_2} are respectively the number of cells in each direction η_1 et η_2 . The algorithm for solving (2.10) with a predictor corrector-time scheme can be written in this case:

1. Initialization:

- $\eta_{1_{min}}, \eta_{2_{min}}, \eta_{1_{max}}, \eta_{2_{max}}$
- $N_{\eta_1}, N_{\eta_2}, \Delta\eta_1, \Delta\eta_2$
- initial distribution: $f(0, \eta_1, \eta_2) = f_0(\eta_1, \eta_2)$
- solving the Laplace equation with f_0 as a source term to get $\Phi(0, \eta_1, \eta_2) = \Phi_0(\eta_1, \eta_2)$
- coordinates transformation $\mathcal{F} : (\eta_1, \eta_2) \rightarrow (x, y)$
- computing the Jacobian matrix $D\mathcal{F}$

- computing the Jacobian using the formula:

$$\sqrt{g}(\eta_1, \eta_2) = \det(D\mathcal{F}) = \frac{\partial x}{\partial \eta_1} \frac{\partial y}{\partial \eta_2} - \frac{\partial x}{\partial \eta_2} \frac{\partial y}{\partial \eta_1}$$

2. Time loop:

- $t^n \rightarrow t^{n+1/2}$
 - calculation of $\Phi^{n+1/2}$ (solving the Poisson equation).
 - calculation of $\partial_{\eta_1} \Phi^{n+1/2}, \partial_{\eta_2} \Phi^{n+1/2}$ on grid points ¹:

$$\partial_{\eta_1} \Phi_{i,j}^{n+1/2} = \frac{\Phi_{i+1,j}^{n+1/2} - \Phi_{i,j}^{n+1/2}}{\Delta \eta_1} \quad \partial_{\eta_2} \Phi_{i,j}^{n+1/2} = \frac{\Phi_{i,j+1}^{n+1/2} - \Phi_{i,j}^{n+1/2}}{\Delta \eta_2}$$

- computing the coefficients of spline of f^n
- solve the equations of the characteristics:

$$\frac{\partial \eta_1(t)}{\partial t} = \frac{\partial_{\eta_2} \Phi^{n+1/2}((\eta_1(t), \eta_2(t)))}{\sqrt{g}} \quad (3.25)$$

$$\frac{\partial \eta_2(t)}{\partial t} = -\frac{\partial_{\eta_1} \Phi^{n+1/2}((\eta_1(t), \eta_2(t)))}{\sqrt{g}} \quad (3.26)$$

$$\eta_1(t^{n+1/2}) = \eta_{1_i} \quad (3.27)$$

$$\eta_2(t^{n+1/2}) = \eta_{2_j} \quad (3.28)$$

- calculation of $f^{n+1/2}$ by interpolation of f^n at the foot of the characteristics.

- $t^{n+1/2} \rightarrow t^{n+1}$
 - calculation of Φ^{n+1} (solving the Poisson equation)
 - calculation of $\partial_{\eta_1} \Phi^{n+1}, \partial_{\eta_2} \Phi^{n+1}$ on grid points :

$$\partial_{\eta_1} \Phi_{i,j}^{n+1} = \frac{\Phi_{i+1,j}^{n+1} - \Phi_{i,j}^{n+1}}{\Delta \eta_1} \quad \partial_{\eta_2} \Phi_{i,j}^{n+1} = \frac{\Phi_{i,j+1}^{n+1} - \Phi_{i,j}^{n+1}}{\Delta \eta_2}$$

- solve the equations of the characteristics:

$$\frac{\partial \eta_1(t)}{\partial t} = \frac{\partial_{\eta_2} \Phi^{n+1}((\eta_1(t), \eta_2(t)))}{\sqrt{g}} \quad (3.29)$$

$$\frac{\partial \eta_2(t)}{\partial t} = -\frac{\partial_{\eta_1} \Phi^{n+1}((\eta_1(t), \eta_2(t)))}{\sqrt{g}} \quad (3.30)$$

$$\eta_1(t^{n+1}) = \eta_{1_i} \quad (3.31)$$

$$\eta_2(t^{n+1}) = \eta_{2_j} \quad (3.32)$$

- calculation of f^{n+1} by interpolation of f^n at the foot of the characteristics.

3. End of time loop.

We can also use a leap frog scheme, as is done in [5].

¹Higher order formulae may be envisaged

3.2 Finite element method with B-spline for the Equation of Quasi neutrality on curvilinear coordinates

We refer to [1], for the computation of the Poisson solver; for sake of completeness, we give here some description of the system that has to be solved. In this section to alleviate the notation we will note the variables (x, y) instead of (η_1, η_2) . The equation 2.14 leads to

$$\begin{aligned} & -\partial_x(A_{1,1}(x, y)\partial_x\Phi(x, y) + A_{1,2}(x, y)\partial_y\Phi(x, y)) \\ & -\partial_y(A_{2,1}(x, y)\partial_x\Phi(x, y) + A_{2,2}(x, y)\partial_y\Phi(x, y)) + c(x, y)\Phi(x, y) = f(x, y)\sqrt{g}(x, y) \end{aligned} \quad (3.33)$$

We consider periodic boundary conditions on the domain $[x_{\min}, x_{\max}] \times [y_{\min}, y_{\max}]$.

Discretization and unknowns

We define $N_x, N_y \in \mathbb{N}^*$, and for $i \in \mathbb{Z}, j \in \mathbb{Z}$,

$$x_i = x_{\min} + i\Delta x, \quad y_j = y_{\min} + j\Delta y, \quad \Delta x = (x_{\max} - x_{\min})/N_x, \quad \Delta y = (y_{\max} - y_{\min})/N_y.$$

The indices i, j may even be extended to $i, j \in \mathbb{R}$.

The unknowns are $\Phi_{i,j} \simeq \Phi(x_i, y_j)$, for $i = 0, \dots, N_x - 1$ and $j = 0, \dots, N_y - 1$.

We suppose to know $A_{i,j}^{p,q} \simeq A_{p,q}(x_i, y_j)$, for $p, q = 1, 2$, $c_{i,j} \simeq c(x_i, y_j)$, $\sqrt{g}_{i,j} \simeq \sqrt{g}(x_i, y_j)$ and $f_{i,j} \simeq f(x_i, y_j)$ for $i = 0, \dots, N_x - 1$ and $j = 0, \dots, N_y - 1$.

We consider the B -spline of order $d \in \mathbb{N}$ B^d which has compact support $[-d/2, d/2]$.

Let $d_x, d_y \in \mathbb{N}$. We introduce a B -spline decomposition

$$\Phi_h(x, y) = \sum_{i \in \mathbb{Z}} \sum_{j \in \mathbb{Z}} \tilde{\Phi}_{i,j} B^{d_x}((x - x_i)/\Delta x) B^{d_y}((y - y_j)/\Delta y).$$

We look for computing the splines coefficients $\tilde{\Phi}_{i,j}$ for $i = 0, \dots, N_x - 1$ and $j = 0, \dots, N_y - 1$, which are then extend to $i, j \in \mathbb{Z}$ by periodicity.

Once they are computed, we get $\Phi_{i,j}$ for $i = 0, \dots, N_x - 1$ and $j = 0, \dots, N_y - 1$ by evaluation:

$$\Phi_{i,j} = \Phi_h(x_i, y_j) = \sum_{k \in \mathbb{Z}} \sum_{\ell \in \mathbb{Z}} \tilde{\Phi}_{k,\ell} B^{d_x}(i - k) B^{d_y}(j - \ell)$$

In order to compute the splines coefficients $\tilde{\Phi}_{i,j}$ for $i = 0, \dots, N_x - 1$ and $j = 0, \dots, N_y - 1$, we take (3.33), with $\Phi = \Phi_h$, multiply it by $B^{d_x}((x - x_k)/\Delta x) B^{d_y}((y - y_\ell)/\Delta y)$ and integrate

over \mathbb{R} , which leads, by integrating by part to the following variational equation:

$$\begin{aligned}
& \sum_{i \in \mathbb{Z}} \sum_{j \in \mathbb{Z}} \tilde{\Phi}_{i,j} \int_{\mathbb{R}^2} \left(\frac{A_h^{1,1}(x,y)}{\Delta x^2} (B^{d_x})'((x-x_i)/\Delta x) B^{d_y}((y-y_j)/\Delta y) (B^{d_x})'((x-x_k)/\Delta x) B^{d_y}((y-y_\ell)/\Delta y) \right. \\
& + \frac{A_h^{1,2}(x,y)}{\Delta x \Delta y} (B^{d_y})'((y-y_j)/\Delta y) B^{d_x}((x-x_i)/\Delta x) (B^{d_x})'((x-x_k)/\Delta x) B^{d_y}((y-y_\ell)/\Delta y) \\
& + \frac{A_h^{2,1}(x,y)}{\Delta x \Delta y} (B^{d_x})'((x-x_i)/\Delta x) B^{d_y}((y-y_j)/\Delta y) B^{d_x}((x-x_k)/\Delta x) (B^{d_y})'((y-y_\ell)/\Delta y) \\
& + \frac{A_h^{2,2}(x,y)}{\Delta y^2} (B^{d_y})'((y-y_j)/\Delta y) B^{d_x}((x-x_i)/\Delta x) B^{d_x}((x-x_k)/\Delta x) (B^{d_y})'((y-y_\ell)/\Delta y) \\
& \left. + c_h(x,y) B^{d_x}((x-x_i)/\Delta x) B^{d_y}((y-y_j)/\Delta y) B^{d_x}((x-x_k)/\Delta x) B^{d_y}((y-y_\ell)/\Delta y) \right) dx dy \\
& = \int_{\mathbb{R}^2} f_h(x,y) \sqrt{g_h}(x,y) B^{d_x}((x-x_k)/\Delta x) B^{d_y}((y-y_\ell)/\Delta y) dx dy
\end{aligned}$$

Here $A_h^{p,q}$, $p, q = 1, 2$, c_h , $\sqrt{g_h}$ and f_h are respectively approximations of $A_{p,q}$, c , \sqrt{g} and f . They may be obtained from a spline decomposition.

We can compute the $d+1$ values

$$L_\ell^d(x) = B^d\left(-\frac{d+1}{2} + \ell + x\right), \quad \ell = 0, \dots, d, \quad 0 \leq x \leq 1,$$

by using the initialization

$$L_0^0(x) = 1,$$

and the recurrence formula for $k = 1, \dots, d$,

$$L_\ell^k(x) = \frac{x+\ell}{k} L_\ell^{k-1}(x) + \frac{k+1-\ell-x}{k} L_{\ell-1}^{k-1}(x), \quad \ell = 1, \dots, k-1,$$

together with

$$L_0^k(x) = \frac{x}{k} L_0^{k-1}(x), \quad L_k^k(x) = \frac{1-x}{k} L_{k-1}^{k-1}(x).$$

4 Numerical results

In this section, we will discuss the results obtained on different geometries and meshes (cartesian, polar, Collela); we first give the expression of the mass and energy in this curvilinear context.

Proposition 4.1. *We define the electric energy in the cartesian case by:*

$$\mathcal{E}(t) = \int_{\Omega} |\partial_x \phi|^2 + |\partial_y \phi|^2 dx dy \quad (4.34)$$

and the mass

$$\mathcal{M}(t) = \int_{\Omega} f dx dy. \quad (4.35)$$

Then we have on the curvilinear geometry

$$\mathcal{E}(t) = \int_{\Omega} [(\partial_{\eta_2} y \partial_{\eta_1} \tilde{\phi} - \partial_{\eta_1} y \partial_{\eta_2} \tilde{\phi})^2 + (-\partial_{\eta_2} x \partial_{\eta_1} \tilde{\phi} + \partial_{\eta_1} x \partial_{\eta_2} \tilde{\phi})^2] \frac{d\eta_1 d\eta_2}{|\sqrt{g}|}, \quad (4.36)$$

with

$$\tilde{\phi}(\eta_1, \eta_2) = \phi(x(\eta_1, \eta_2), y(\eta_1, \eta_2)),$$

and

$$\sqrt{g} = \partial_{\eta_1} x \partial_{\eta_2} y - \partial_{\eta_1} y \partial_{\eta_2} x.$$

Proof 4.1. We have

$$\partial_{\eta_1} \tilde{\phi} = \partial_{\eta_1} x \partial_x \phi + \partial_{\eta_1} y \partial_y \phi, \quad \partial_{\eta_2} \tilde{\phi} = \partial_{\eta_2} x \partial_x \phi + \partial_{\eta_2} y \partial_y \phi,$$

which leads to

$$\begin{bmatrix} \partial_x \phi \\ \partial_y \phi \end{bmatrix} = \frac{1}{\sqrt{g}} \begin{bmatrix} \partial_{\eta_2} y \partial_{\eta_1} \tilde{\phi} - \partial_{\eta_1} y \partial_{\eta_2} \tilde{\phi} \\ -\partial_{\eta_2} x \partial_{\eta_1} \tilde{\phi} + \partial_{\eta_1} x \partial_{\eta_2} \tilde{\phi} \end{bmatrix}$$

This leads to (4.39).

Application to Colella mesh. We define the mapping

$$x(\eta_1, \eta_2) = \eta_1 + \alpha \sin(\eta_1) \sin(\eta_2), \quad y(\eta_1, \eta_2) = \eta_2 + \alpha \sin(\eta_1) \sin(\eta_2). \quad (4.37)$$

We then get

$$\partial_{\eta_1} x = 1 + \alpha \cos(\eta_1) \sin(\eta_2), \quad \partial_{\eta_1} y = \alpha \cos(\eta_1) \sin(\eta_2),$$

and

$$\partial_{\eta_2} x = \alpha \sin(\eta_1) \cos(\eta_2), \quad \partial_{\eta_2} y = 1 + \alpha \sin(\eta_1) \cos(\eta_2).$$

This leads to

$$\sqrt{g} = |(1 + \alpha \cos(\eta_1) \sin(\eta_2))(1 + \alpha \sin(\eta_1) \cos(\eta_2)) - \alpha \cos(\eta_1) \sin(\eta_2) \alpha \sin(\eta_1) \cos(\eta_2)|,$$

which reads

$$\sqrt{g} = 1 + \alpha \sin(\eta_1 + \eta_2),$$

for $|\alpha| < 1$. The energy reads

$$\mathcal{E}(t) = \int_{\Omega} \frac{1}{1 + \alpha \sin(\eta_1 + \eta_2)} [(\partial_{\eta_2} y \partial_{\eta_1} \tilde{\phi} - \partial_{\eta_1} y \partial_{\eta_2} \tilde{\phi})^2 + (-\partial_{\eta_2} x \partial_{\eta_1} \tilde{\phi} + \partial_{\eta_1} x \partial_{\eta_2} \tilde{\phi})^2] d\eta_1 d\eta_2.$$

We now write $\tilde{\phi}(\eta_1, \eta_2, \alpha) = \phi(x(\eta_1, \eta_2, \alpha), y(\eta_1, \eta_2, \alpha))$.

Proposition 4.2. On Colella mesh the electric energy and mass are independent of α .

$$\partial_{\alpha} \mathcal{M}(t) = 0 \quad (4.38)$$

$$\partial_{\alpha} \mathcal{E}(t) = 0 \quad (4.39)$$

Proof 4.2. *We consider*

$$I = \int_{\Omega} u(x, y) dx dy = \int_{\Omega} \sqrt{g} u(x(\eta_1, \eta_2, \alpha), y(\eta_1, \eta_2, \alpha)) d\eta_1 d\eta_2. \quad (4.40)$$

We have of course $\partial_{\alpha} I = 0$. We want to check this by taking the expression on the RHS of (4.40).

We compute

$$\partial_{\alpha} I = \int_{\Omega} \partial_{\alpha} \sqrt{g} u(x(\eta_1, \eta_2, \alpha), y(\eta_1, \eta_2, \alpha)) d\eta_1 d\eta_2 + \int_{\Omega} \sqrt{g} (\partial_{\alpha} x \partial_x u + \partial_{\alpha} y \partial_y u) d\eta_1 d\eta_2$$

Now, we have

$$\sqrt{g} \partial_x u = \partial_{\eta_2} y \partial_{\eta_1} \tilde{u} - \partial_{\eta_1} y \partial_{\eta_2} \tilde{u}, \quad \sqrt{g} \partial_y u = -\partial_{\eta_2} x \partial_{\eta_1} \tilde{u} + \partial_{\eta_1} x \partial_{\eta_2} \tilde{u}.$$

We can then check that $\partial_{\alpha} I = 0$, by integrating by parts and by checking the relation

$$\partial_{\alpha} \sqrt{g} - \partial_{\eta_1} (\partial_{\alpha} x \partial_{\eta_2} y - \partial_{\alpha} y \partial_{\eta_2} x) - \partial_{\eta_2} (-\partial_{\alpha} x \partial_{\eta_1} y + \partial_{\alpha} y \partial_{\eta_1} x) = 0.$$

Remark 4.3. *This property ensure that the mass and the energy are unaffected by the choice of the mapping.*

4.1 Cartesian domain

4.1.1 Cartesian mesh

We first test the code on a uniform, Cartesian mesh, which corresponds to the identity mapping

$$\begin{aligned} x(\eta_1, \eta_2) &= \eta_1 \\ y(\eta_1, \eta_2) &= \eta_2 \end{aligned}$$

The initial distribution \tilde{f}_0 is given by the formula :

$$\tilde{f}_0(x, y) = \sin(x) \sin(y) + \beta \sin(2\sigma x) \sin(2y)$$

where $\beta = 0.015$ and $\sigma = 0.5$.

We have considered a predictor-corrector scheme (PC) and also a leap-frog scheme (LF) for comparisons, on a 128×128 or 256×256 grid, and $dt = 0.05$. Results are given on Figures 2,3,4 and 5.

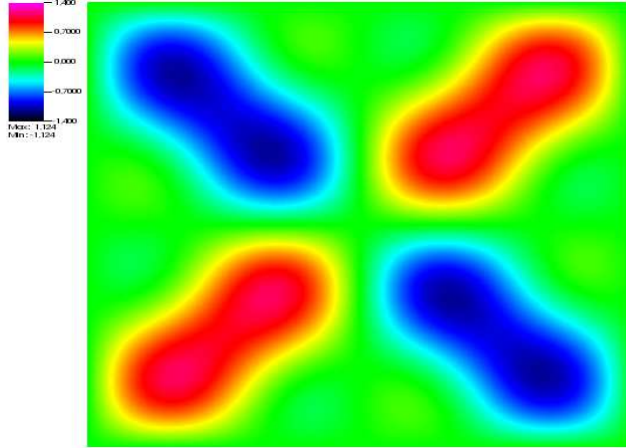
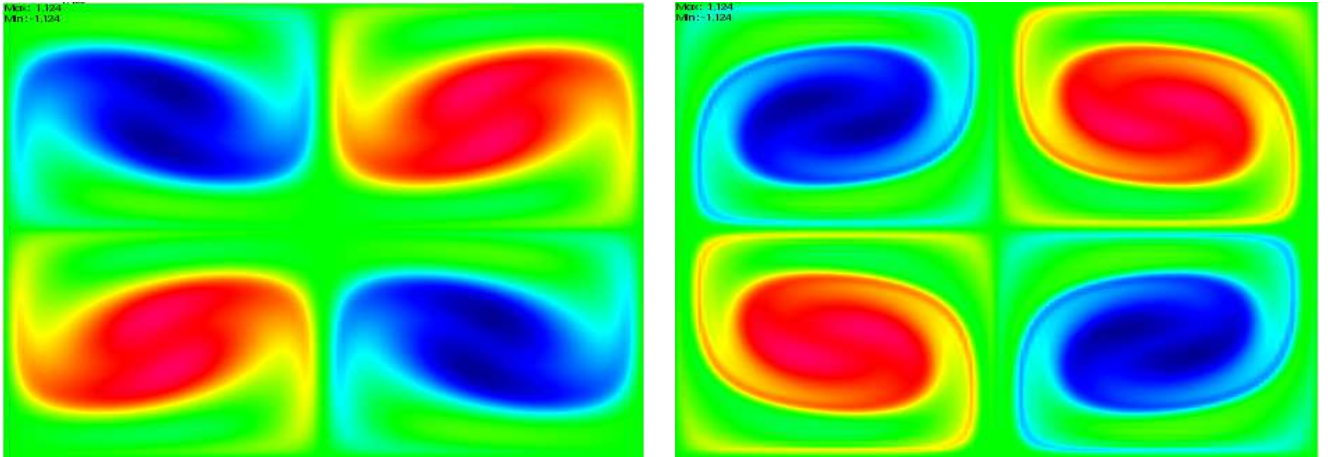


Figure 2: Plot of the initial distribution.



4.1.2 Colella mesh

We solve the equations on a regular rectangular mesh. The domain is mapped according to the nonlinear mapping we considered which is a specialization of the mapping used in [6][7]. This mapping is generated by perturbing a uniform Cartesian mesh by a Cartesian sinusoidal product, specifically,

$$x(\eta_1, \eta_2) = \eta_1 + \alpha \sin(\eta_1) \sin(\eta_2)$$

$$y(\eta_1, \eta_2) = \eta_2 + \alpha \sin(\eta_1) \sin(\eta_2)$$

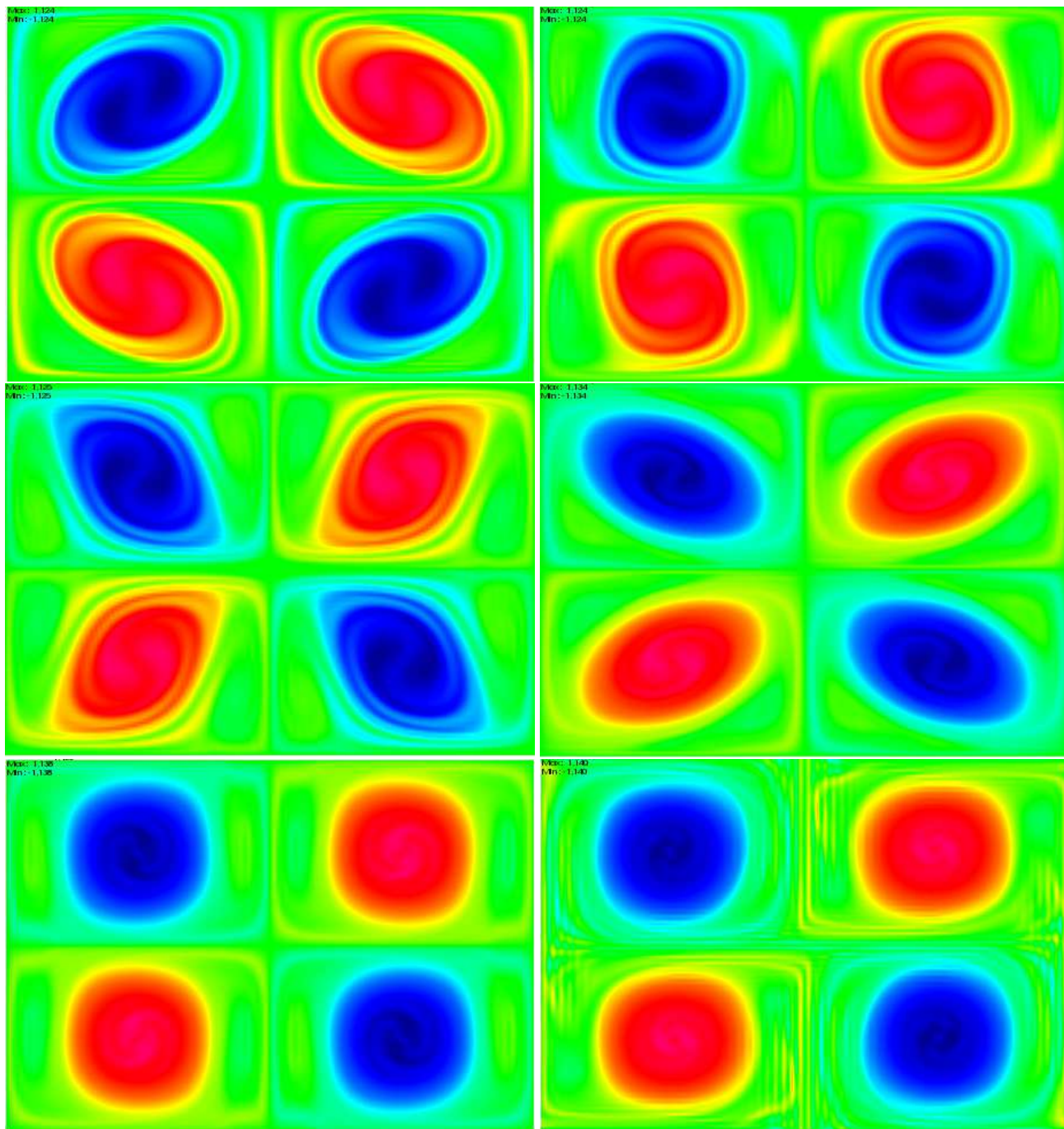
where α is the mesh parameter and $\eta_1, \eta_2 \in (0, 2\pi)$. To ensure that the perturbed mesh does not tangle, it is sufficient to take $0 < \alpha < 1$.

The initial distribution \tilde{f}_0 is given by the formula :

$$\tilde{f}_0(x, y) = \sin(y) + \beta \cos(\sigma x)$$

where $\beta = 0.015$ and $\sigma = 0.5$.

For reference, we compute on a uniform, Cartesian mesh, which corresponds to the identity mapping ($\alpha = 0$). We made several simulations by varying the value of parameter α . We have taken the time step $dt = 0.1$ (or $dt = 0.05$, when specified) and the space step $d\eta_1 = d\eta_2 = 1/256$. Results are given on Figures 6 to 20.



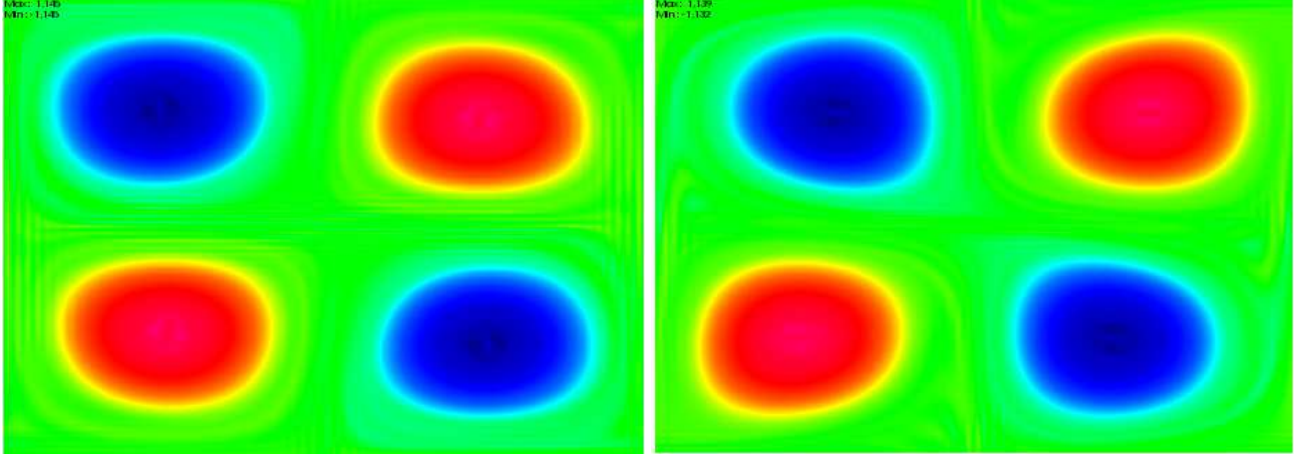


Figure 3: Plot of the solution on 128×128 grid at several times $t = 10, 20, 30, 40, 50, 100, 150, 200, 350, 400$.

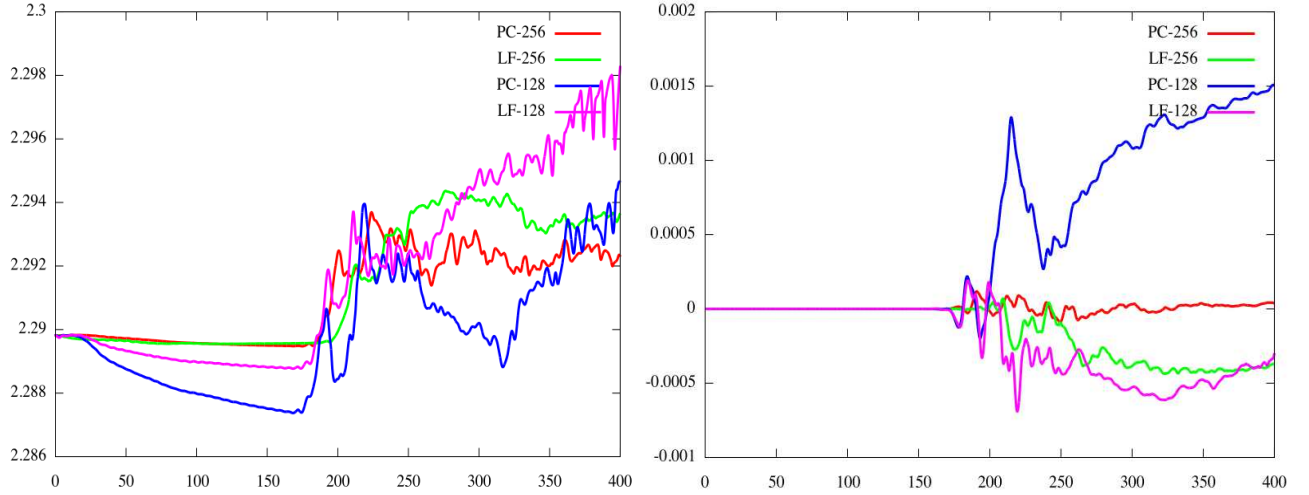


Figure 4: Plot of energy (left) and mass (right)

4.2 Polar domain

We are interested in a polar geometry which corresponds to the mapping

$$\begin{aligned} x(\eta_1, \eta_2) &= \eta_1 \cos(\eta_2) \\ y(\eta_1, \eta_2) &= \eta_2 \sin(\eta_2) \end{aligned}$$

to simulate the diocotron instability for an annular electron layer, see [3]. We consider the following initial data:

$$f_0(\eta_1, \eta_2) = \begin{cases} 0, & r_{min} \leq \eta_1 \leq r^- \\ 1 + \varepsilon \cos(m\eta_2) & r^- \leq \eta_1 \leq r^+ \\ 0 & r^+ \leq \eta_1 \leq r_{max} \end{cases}$$

where $\varepsilon = 10^{-6}$ is a small parameter, and m is the mode. In our simulations, we take $m = 3, r^- = 4, r^+ = 5$ for the first one, and $m = 7, r^- = 6, r^+ = 7$ for second one. We

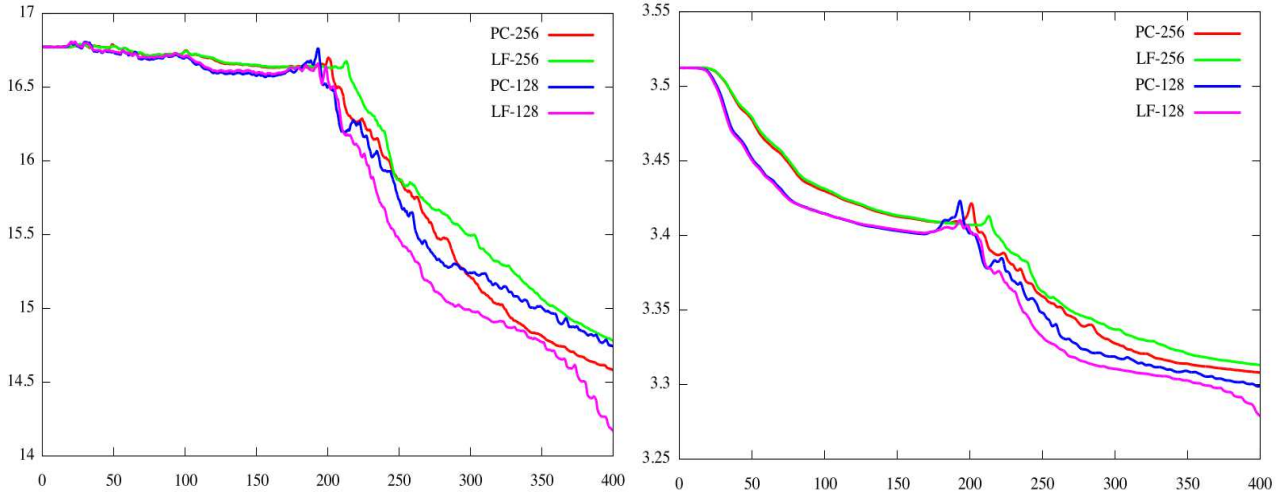


Figure 5: Plot of norm L1 (left) and norm L2 (right)

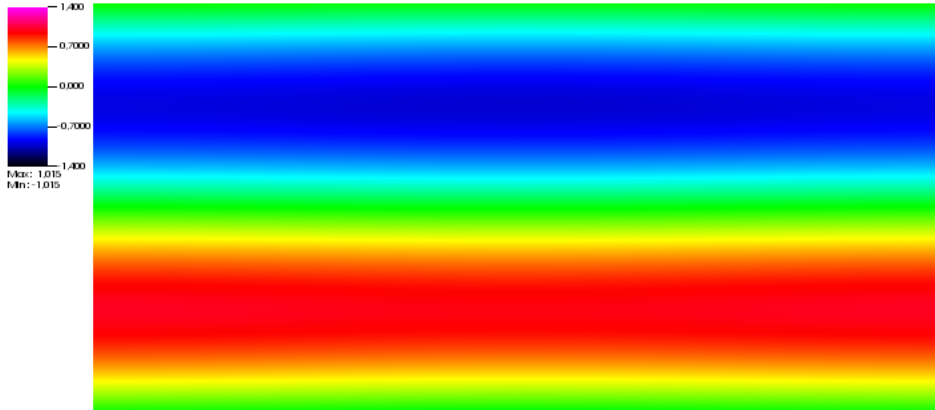


Figure 6: Plot of f_0

have considered Dirichlet boundary condition on both sides r^- and r^+ for the solving of the Poisson equation. For f , we take the periodic condition on η_2 and Dirichlet condition on η_1 . Numerical results are given on Figures 21 to 24. We compare here with a simulation using FFT for the Poisson solver [3]. We obtain the right slopes, as in [3].

5 Conclusion

Some first guiding-center semi-Lagrangian simulations have been performed on general curvilinear grids. The method has been validated on two different geometries and on different grids. One key point has been to have the possibility of using a Poisson solver [1] on curvilinear grids. In future work, we must deepen our investigations and make other tests case in more complex geometry to better establish the robustness of the numerical method for the simulation of 2D guiding-center and Vlasov equations. Further numerical methods should be tried to see if we can gain on the conservation of quantities like mass and energy, which is generally more complicated to have in this setting. We plan to do comparisons with other numerical methods like CSL 1D [8] and CSL 2D [9] recently developed in the project Selalib.

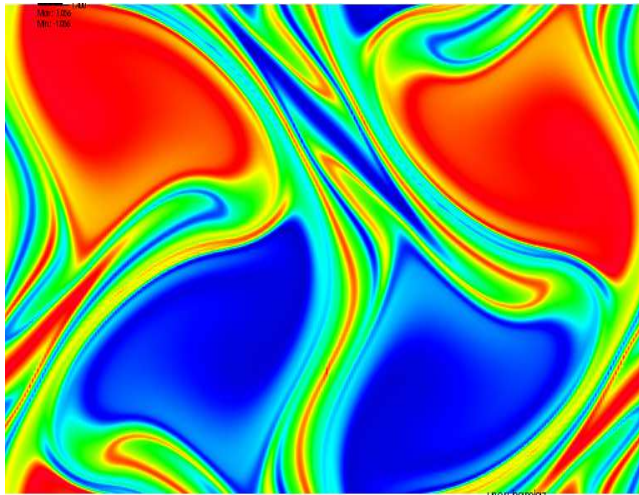


Figure 7: Plot of the reference solution at time $t = 30$.

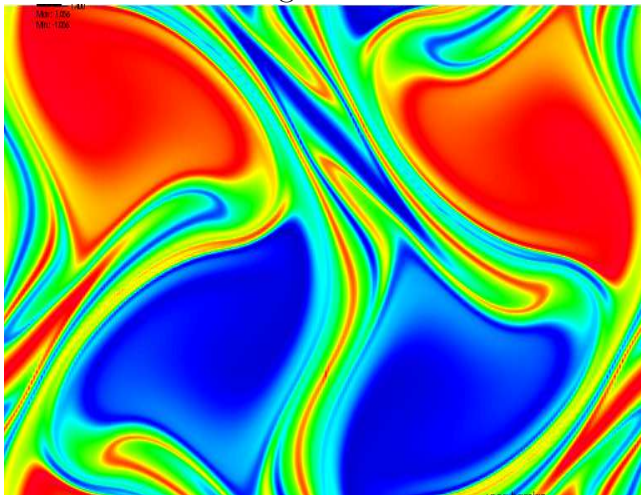


Figure 8: Plot of the solution at time $t = 30$ for $\alpha = 1e - 6$.

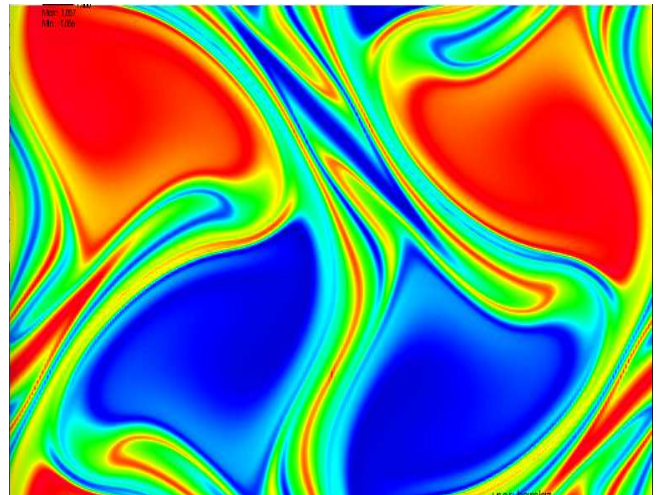


Figure 9: Plot of the solution at time $t = 30$ for $\alpha = 1e - 3$.

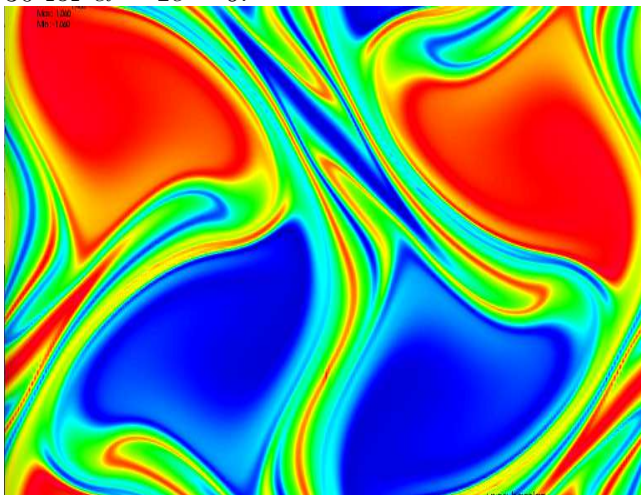


Figure 10: Plot of the solution at time $t = 30$ for $\alpha = 0.1$.

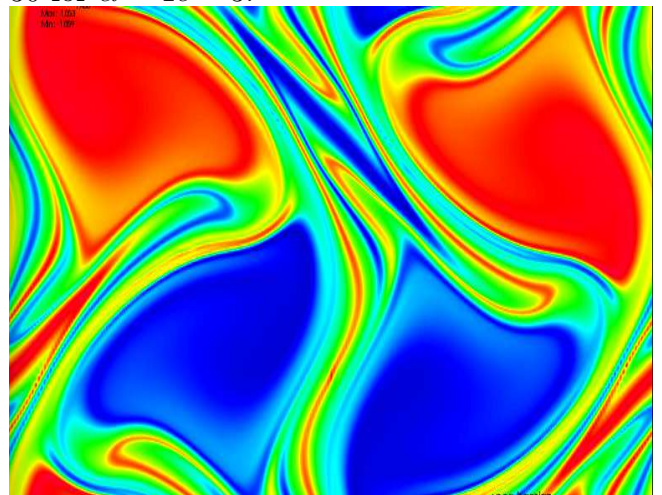


Figure 11: Plot of the solution at time $t = 30$ for $\alpha = 0.2$.

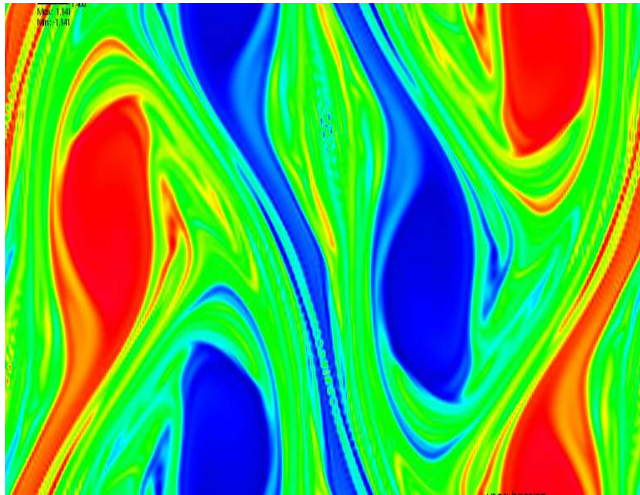


Figure 12: Plot of the reference solution at time $t = 60$.

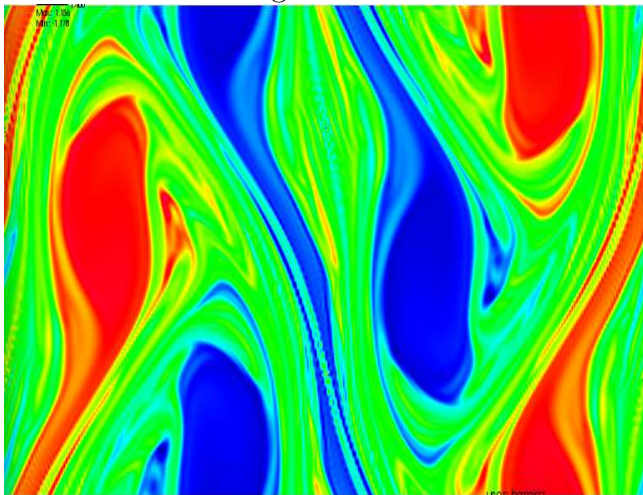


Figure 13: Plot of the solution at time $t = 60$ for $\alpha = 1e - 6$.

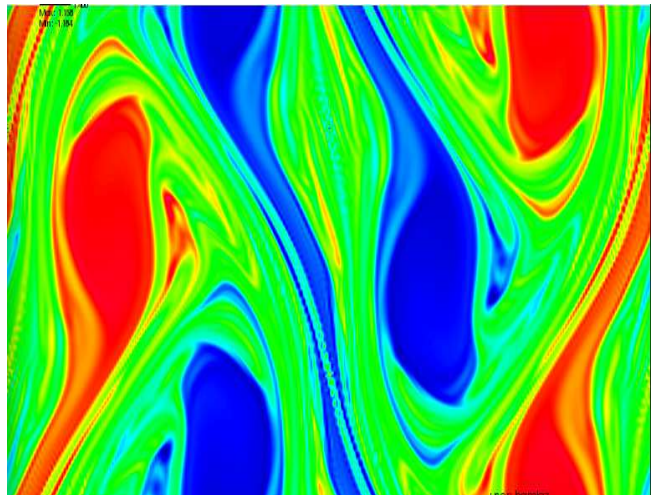


Figure 14: Plot of the solution at time $t = 60$ for $\alpha = 1e - 3$.

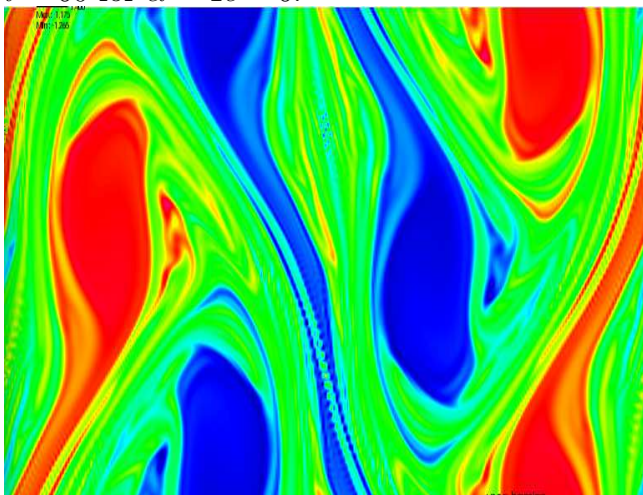


Figure 15: Plot of the solution at time $t = 60$ for $\alpha = 0.1$.

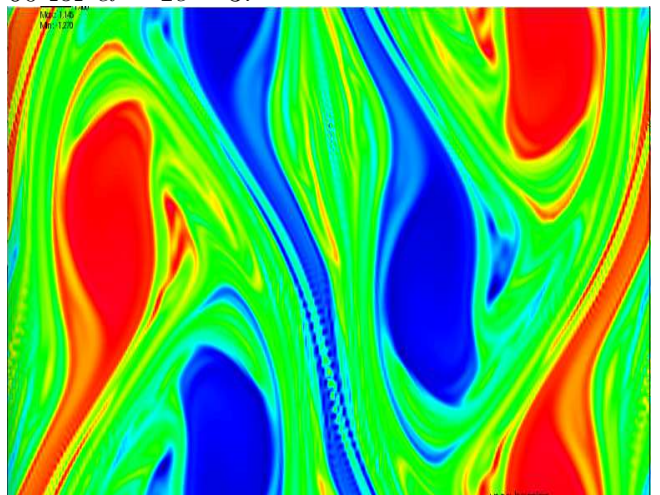


Figure 16: Plot of the solution at time $t = 60$ for $\alpha = 0.2$.

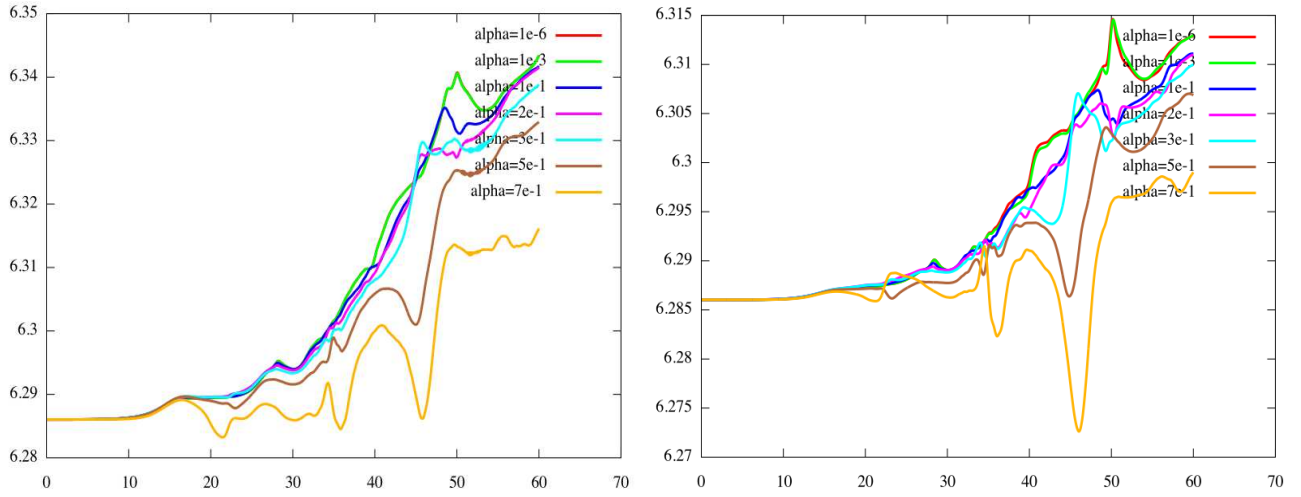


Figure 17: Plot of energy (left with $dt=0.1$ and right with $dt=0.05$)

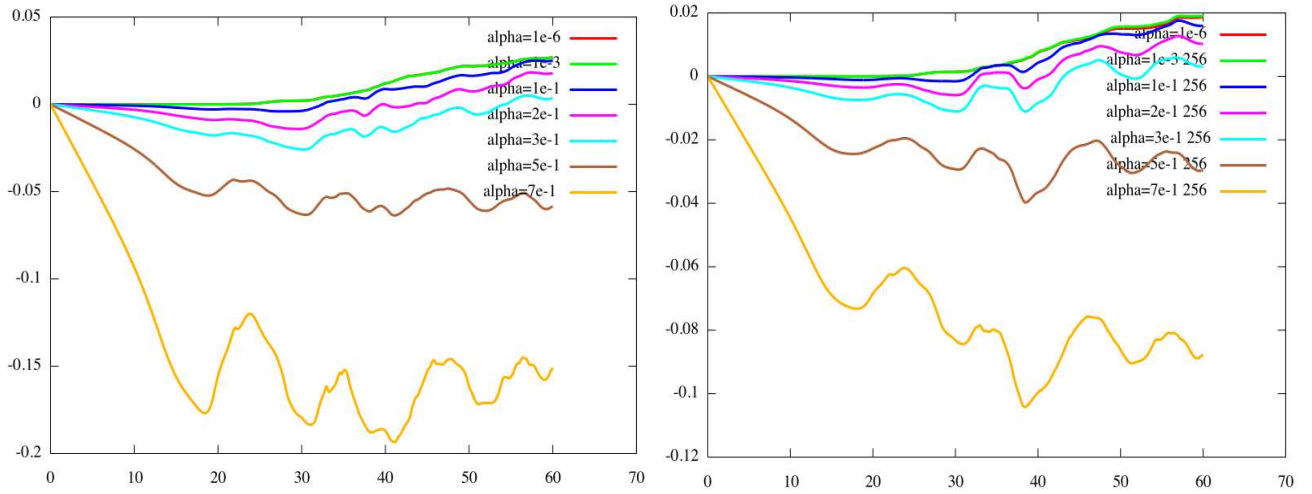


Figure 18: Plot of mass (left with $dt=0.1$ and right with $dt=0.05$)

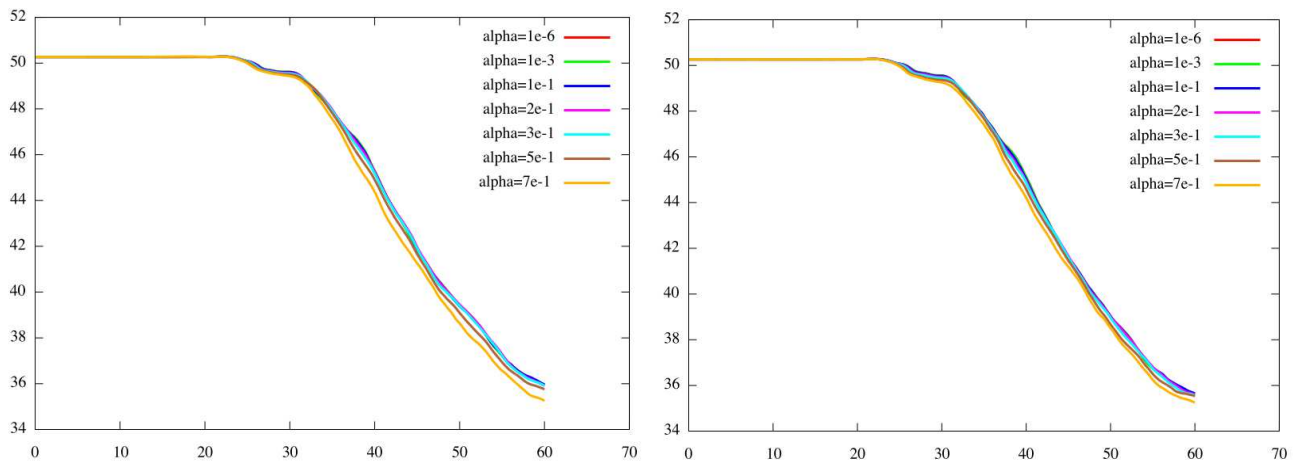


Figure 19: Plot of norm L1 (left with $dt=0.1$ and right with $dt=0.05$)

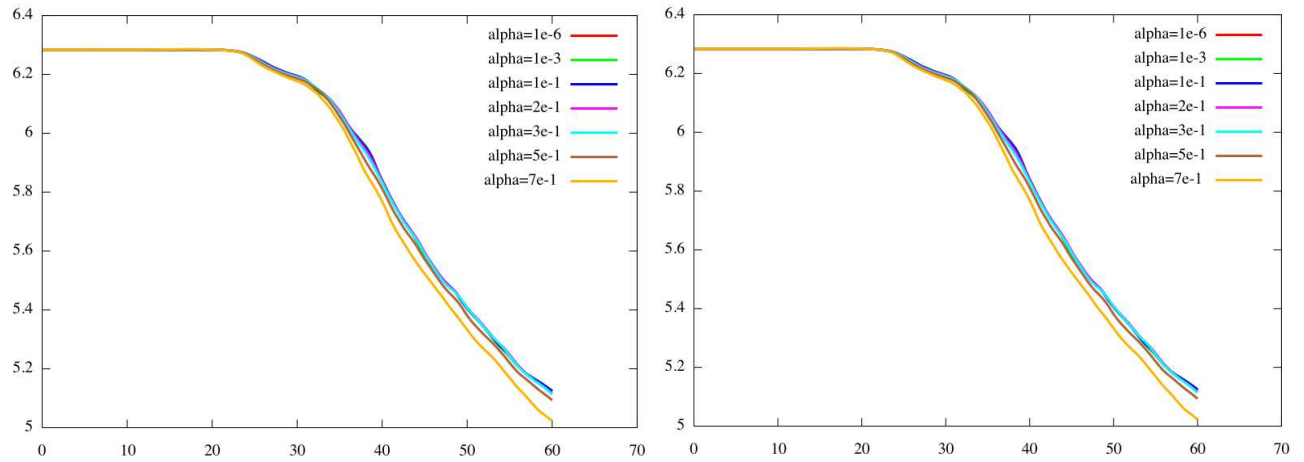


Figure 20: Plot of norm L2 (left with $dt=0.1$ and right with $dt=0.05$)



Figure 21: Plot of the initial density f_0

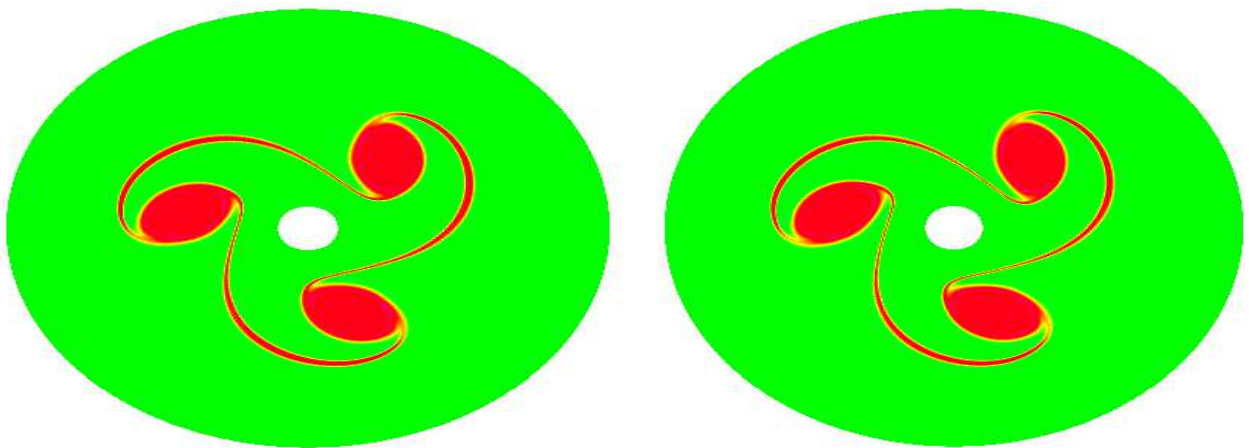


Figure 22: Cartography of the density f obtained with $m=3$. (Left) curvilinear Poisson solver (Right) Fourier Poisson solver.

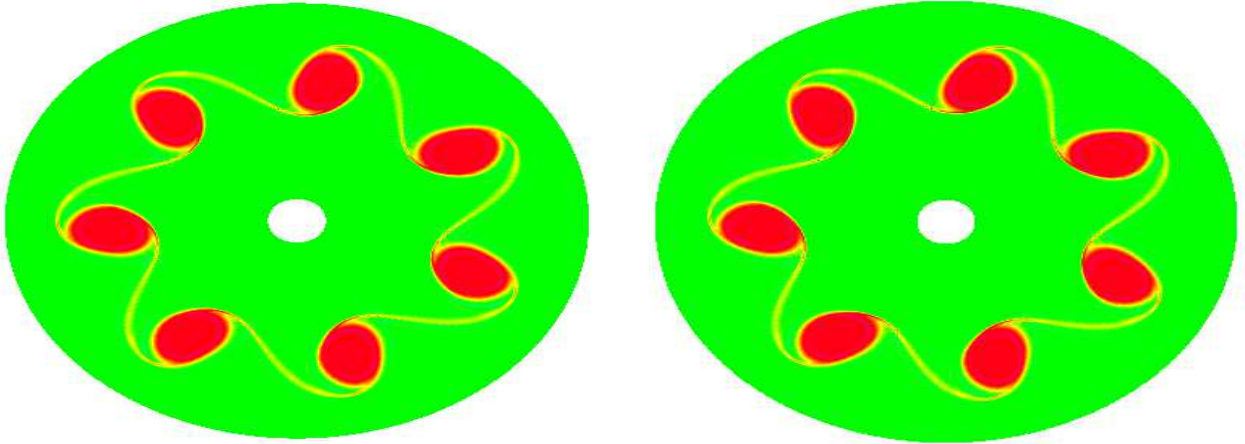


Figure 23: (Left) Cartography of the density f obtained with $m = 7$. (Left) curvilinear Poisson solver (Right) Fourier Poisson solver

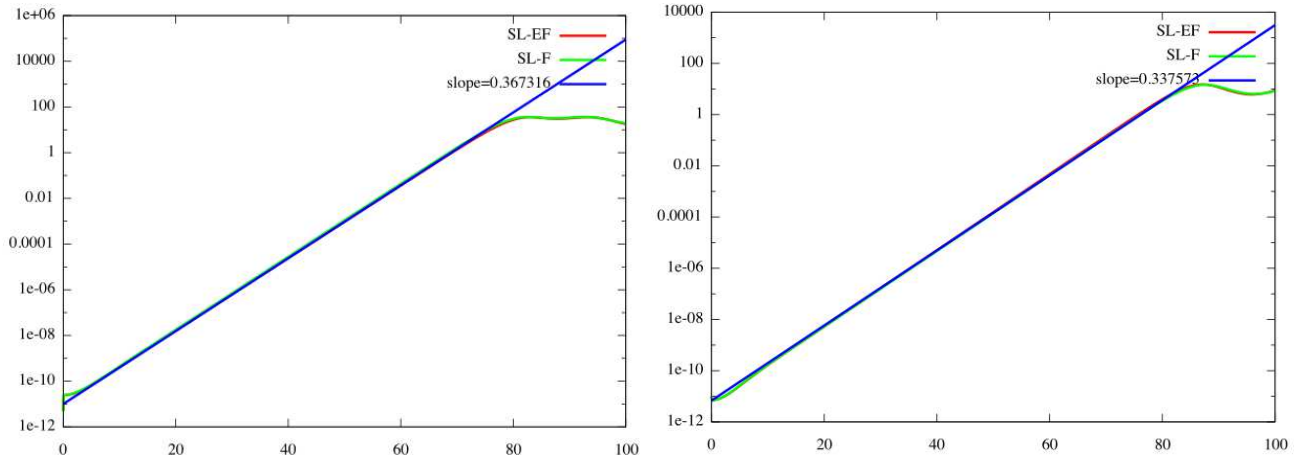


Figure 24: Square modulus of the 3rd(Left) and 7th (Right) Fourier mode of $\int_{r_{min}}^{r_{max}} \eta_1 \phi d\eta_1$ SL-EF is for curvilinear Poisson solver and SL-F is for Fourier Poisson solver.

References

- [1] A. Back, Work in progress.
- [2] J. P. Braeunig, N. Crouseilles, M. Mehrenberger, E. Sonnendrucker, *Guiding-center simulations on curvilinear meshes*, Discrete and Continuous Dynamical Systems Series S, Volume 5, Number 3, June 2012
- [3] S. Hirstoaga, E. Madaule, M. Mehrenberger, J. Pétri, *Semi-Lagrangian simulations of the diocotron instability*, hal-00841504.
- [4] SELALIB, <http://selalib.gforge.inria.fr/>
- [5] E. SONNENDRÜCKER, J. ROCHE, P. BERTRAND AND A. GHIZZO, *The Semi-Lagrangian Method for the Numerical Resolution of the Vlasov Equation*, J. Comput. Phys. **149**, 201-220 (1999).

- [6] P.O. PERSSON, J. BONET AND J. PERAIRE, *Discontinuous Galerkin solution of the Navier–Stokes equations on deformable domains*, Computer Methods in Applied Mechanics and Engineering **198**(17), 1585-1595 (2009).
- [7] P. COLELLA, M. R. DORRAND, J.A. HITTINGER AND D.F. MARTIN, *High-order, finite-volume methods in mapped coordinates*, J. Comput. Phys. **230**(8), 2952–2976 (2011).
- [8] N. CROUSEILLES, M. MEHRENBERGER, E. SONNENDRUCKER, *Conservative semi-Lagrangian schemes for Vlasov equations*, Journal of Computational Physics **229**(2010), 1927–1953.
- [9] P. GLANC N. CROUSEILLES, M. MEHRENBERGER , Work in progress.