A Discontinuous Galerkin semi-Lagrangian solver for the guiding-center problem

Nicolas Crouseilles

INRIA-Rennes Bretagne Atlantique & IRMAR, 263 Avenue du General Leclerc CS 74205 35042 RENNES CEDEX (France)

Michel Mehrenberger

UDS-IRMA, 7 rue Rene Descartes 67084 Strasbourg (France)

Francesco Vecil

Universitat de València, Departament de Matemàtica Aplicada, calle del Doctor Moliner 50, Burjassot 46100 (Spain)

Abstract

In this paper, we test an innovative numerical scheme for the simulation of the guiding-center model, of interest in the domain of plasma physics, namely for fusion devices. We propose a 1D Discontinuous Galerkin (DG) discretization, whose basis are the Lagrange polynomials interpolating the Gauss points inside each cell, coupled to a conservative semi-Lagrangian (SL) strategy. Then, we pass to the 2D setting by means of a second-order Strang-splitting strategy. In order to solve the 2D Poisson equation on the DG discretization, we adapt the spectral strategy used for equally-spaced meshes to our Gauss-point-based basis. The 1D solver is validated on a standard benchmark for the nonlinear advection; then, the 2D solver is tested against the swirling deformation flow test case; finally, we pass to the simulation of the guiding-center model, and compare our numerical results to those given by the Backward Semi-Lagrangian method.

Keywords: Discontinuous Galerkin, semi-Lagrangian method, Strang

Email addresses: nicolas.crouseilles@inria.fr (Nicolas Crouseilles), mehrenbe@math.unistra.fr (Michel Mehrenberger), francesco.vecil@gmail.com (Francesco Vecil)

splitting, guiding-center, plasma physics, conservative
2000 MSC: 35L04, 35L65, 65M25, 65M60

1. Introduction

The numerical simulation of the dynamics of charged particles in a plasma is becoming increasingly important with reference to applications such as propagating beams or fusion devices used, for example, in the ITER project. The proper evolution equation to describe those phenomena would be a full 3D collisionless Vlasov kinetic equation, which is 6D in the phase space and, therefore, numerically too costly to simulate: reduced models must thus be taken into account. Inside the tokamak reactor, intense magnetic fields are generated: they are the superposition of a poloidal field, which makes the ions circulate around the reactor chamber, and a toroidal field, which makes the ions move following a fast cyclotronic motion (helices): the goal of the toroidal part is to stabilize the orbits of the charged particles and thus prevent them from touching the metallic walls. The *gyrokinetic* model is a reduced model that describes, instead of the trajectories of the helices, those of the centers of the helices: in the parallel direction and if one considers a constant magnetic field directed along this direction, it is nothing else but a 1D Vlasov equation; in the perpendicular direction, the so-called 2D *guiding-center* model is obtained (see for example [1] for a mathematical derivation). Denoting by $\rho(t, x_1, x_2)$ the density of particles at time $t \in [0, +\infty[$ and position $(x_1, x_2) \in \left[0, \frac{2\pi}{k}\right] \times [0, 2\pi]$, and $\Phi(t, x_1, x_2)$ the self-consistent electric potential given by the Poisson equation, the guiding-center model writes

$$\frac{\partial \rho}{\partial t} + \frac{\partial \Phi}{\partial x_2}\frac{\partial \rho}{\partial x_1} - \frac{\partial \Phi}{\partial x_1}\frac{\partial \rho}{\partial x_2} = 0, \qquad -\Delta_{x_1,x_2}\Phi = \rho(t, x_1, x_2) \tag{1}$$

completed by the initial condition

$$\rho(t = 0, x_1, x_2) = \sin(x_2) + \varepsilon \cos(k x_1)$$

and endowed by periodic boundaries for both the Vlasov and the Poisson equations. A compatibility condition $\iint \Phi(x_1, x_2) \mathrm{d}x_1 \mathrm{d}x_2 = 0$ ensures the well posedness of the model, which is nothing else but the 2D Euler system.

The guiding-center model is more challenging to simulate than the Vlasov system, because the advection fields

$$a_1(t, x_1, x_2) := \frac{\partial \Phi}{\partial x_2}(t, x_1, x_2) \qquad \text{and} \qquad a_2(t, x_1, x_2) := -\frac{\partial \Phi}{\partial x_1}(t, x_1, x_2)$$

are nonlinear and non-constant with respect to the advected variables x_1 and x_2. Note that, due to the fact that the advection field

$$\boldsymbol{a}(t, x_1, x_2) := \left(\frac{\partial \Phi}{\partial x_2}(t, x_1, x_2), -\frac{\partial \Phi}{\partial x_1}(t, x_1, x_2) \right)$$

is divergence-free, the evolution equation can be written in an equivalent conservative form

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_1}\left[a_1 \rho\right] + \frac{\partial}{\partial x_2}\left[a_2 \rho\right] = 0.$$

In this work, we propose for problem (1) an innovative way to couple a Discontinuous Galerkin (DG) discretization to a semi-Lagrangian (SL) strategy and a spectrally-accurate solver for the 2D Poisson equation in a periodic setting, adapted for the computation on our non-equally-spaced discretization points. First of all, in order to reduce the dimensionality of the problem from 2D to 1D, we use a splitting procedure [2, 3]. Then, the computational domain Ω is meshed using the DG philosophy, which consists in dividing Ω into "big" cells, inside each of which there is a subdivision given by a certain number of points, called *degrees of freedom*: for the scope of this work, we shall use $d + 1$ Gauss points for an order-d method. Finally, we solve the series of 1D problems through a conservative method based on calculating the characteristics, following the SL approach, in order to ensure the conservation of the number of particles (see [4]). Using this DG SL strategy, the resolution of the density function is thus improved by local refinement, which provides several advantages (see [5, 6, 7]):

- In this SL framework, the constraints on the time step are weaker and are made dependent on the cell size rather than the degrees of freedom, therefore we can improve the resolution without making $\Delta t \to 0$.
- Parallelization can be made easier because the communication concerns only one cell at each of the boundaries.
- Improved flexibility due to the discontinuous representation.
- They open the doors to an Adaptive Mesh Refinement (AMR) strategy, possibly based on increasing the degrees of freedom in the zones where the distribution function presents microscopic structures or violent gradients, like filamentation, vortices or sharp edges.

DG methods have been successfully used and analyzed in several contexts: advection-reaction-diffusion problems [8, 9], convection-diffusion problems [10, 11, 12, 13, 14], elliptic equations [15, 16, 17, 18, 19], kinetic equations [20, 21], Euler and aeroacoustic equations [22, 23]. DG methods coupled with SL schemes have also been used for meteorological applications [24] and kinetic equations, see [25, 26, 27] and references therein.

In this paper we extend the previous work [27] to general variable coefficients: there, 1D and 2D linear advections were implemented and tested through Vlasov-Poisson test cases, with encouraging results. In this work, our starting point is a scheme for the 1D case of non-constant advection in order to, at the end, be able to simulate the guiding-center model (1). As in [27] we discretize the 1D computational domain Ω by dividing it into cells $\{I_i\}$; for a discretization of order d, we use as basis of the DG space the Lagrange polynomials $\{\varphi_{i,j}(x)\}$ interpolating the Gauss points inside the cells: by doing so, we get an orthogonal basis in the sense that

$$\int_\Omega \varphi_{i,j}(x)\varphi_{i',j'}(x)\mathrm{d}x = \delta_{i,i'}\delta_{j,j'} \qquad \text{up to a constant.}$$

This property is important because we can thus avoid solving costlier linear systems inside our scheme. By using the volume conservation property, the weak formulation enables to advance in time and conserve, by construction, the total number of particles. With respect to the linear case treated in [27], additional work has to be done in order to deal with advection fields non-constant with respect to the advected variable, as in (1), thus the SL solver has to be coupled to an ODE integrator to solve the characteristics, either backward of forward: we show that the resulting schemes, which we call *backward scheme* and *forward scheme*, are the same, provided that the characteristics are solved in a reversible-in-time way. To do that, we impose that the relative distances between points

$$x_{i,j} < x < x_{i,j+1}$$

be kept fixed at the level of their backward images

$$x^b_{i,j} := \mathcal{X}\left(t^n; t^{n+1}, x_{i,j}\right) < x^b < x^b_{i,j+1} := \mathcal{X}\left(t^n; t^{n+1}, x_{i,j+1}\right),$$

being $\mathcal{X}\left(t^n; t^{n+1}, x\right)$ the characteristic starting at time t^{n+1} from point x

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

by means of an affine approximation. Moreover, we stress that it is very important that the advection field $a(t,x)$ be reconstructed in a continuous way, otherwise the characteristics might cross and mass could be lost. We then extend our scheme to the 2D case by means of the Strang splitting [2, 3]. Inspired by the standard schemes that solve the Poisson equation in a periodic setting for an equally-spaced mesh, we propose a spectrally-accurate solver based on mixing the representation of $\rho(t,x_1,x_2)$, $\frac{\partial\Phi}{\partial x_1}$ and $\frac{\partial\Phi}{\partial x_2}$ both on the Fourier basis and on the DG basis: this way we can compute with arbitrarily high precision the advection field on the Gauss points; notwithstanding, we do not represent it on the discontinuous basis, rather we linearly interpolate between two consecutive points. We want to stress that the spectral strategy is efficient from the computational point of view, because much of the calculation can be precomputed, and the Discrete Fourier Transform and AntiTransform involved can be performed using an optimized FFT routine.

The paper is organized as follows: in Section 2 we explain in detail how the 1D and 2D nonlinear advections are solved; then, in Section 3 we validate the method against some standard benchmark tests; in Section 4 we show our numerical results for the guiding-center model and the comparison with the Backward Semi-Lagrangian method; finally, in AppendixA we present the spectrally-accurate solver for the solution of the 2D Poisson equation in a periodic setting on a DG discretization.

2. Numerical methods

In this section, we expose our numerical methods. First, we give the basis for our DG discretization. Then, we explain in detail how we solve the 1D nonlinear advection, with particular emphasis on the affine approximation for the characteristics; we show, moreover, that the forward and the backward formulations give the same scheme. Finally, we explain how we pass to the 2D case thanks to a Strang-splitting strategy.

2.1. DG discretization

The domain $\Omega = [x_{\min}, x_{\max}] \subseteq \mathbb{R}$ is partitioned into N homogeneous intervals

$$\left\{ I_i := [x_{i-1/2}, x_{i+1/2}] = [x_{i-1/2}, x_{i-1/2} + \Delta x] \right\}_{i=0,\dots,N-1}$$

so that $\Omega = \bigcup_{i=0}^{N-1} I_i$. We denote by $V_{\Delta x}^d(\Omega)$ the discontinuous finite element space

$$V_{\Delta x}^d(\Omega) = \left\{ \psi \in L^2(\Omega) : \psi \in \mathbb{P}^d(I_i), \qquad \text{for } i = 0, \ldots, N-1 \right\},$$

where $\mathbb{P}^d(I_i)$ is the space of the one-variable polynomials of degree at most d on the interval I_i. As a basis for $V_{\Delta x}^d(\Omega)$ we use the $N \times (d+1)$ Lagrange polynomials that interpolate the $d+1$ Gauss points, noted

$$\{x_{i,j} \in I_i\}_{(i,j) \in \{0,\ldots,N-1\} \times \{0,\ldots,d\}},$$

inside the cell I_i:

$$\varphi_{i,j}(x) \quad = \quad \begin{cases} \varphi^j[x_{i,0}, x_{i,1}, \ldots, x_{i,d}](x) := \displaystyle\prod_{0 \le \ell \le d, \ell \neq j} \frac{x - x_{i,\ell}}{x_{i,j} - x_{i,\ell}} & x \in I_i \\ \\ 0 & x \in \Omega \setminus I_i \end{cases}.$$

For the sake of compactness of notation in the future formulae, we shall use

$$x_{i,-1} := x_{i-1/2}, \qquad x_{i,d+1} := x_{i+1/2}, \qquad i = 0, \ldots, N-1.$$

If we now let Π^d the standard projection of $L^2(\Omega)$ onto $V_{\Delta x}^d(\Omega)$:

$$\begin{array}{rcl} L^2 & \to & V_{\Delta x}^d \\ \omega & \mapsto & \Pi^d(\omega) = \omega_{i,j} := \langle \omega, \varphi_{i,j} \rangle := \displaystyle\int_\Omega \omega(x) \varphi_{i,j}(x) \mathrm{d}x, \end{array}$$

we can represent $f(t^n, x)$ at time t^n on the DG basis as

$$f^n(x) = \sum_{i'=0}^{N-1} \sum_{j'=0}^{d} f_{i',j'}^n \varphi_{i',j'}(x). \tag{2}$$

2.2. 1D nonlinear advection

We shall suppose that the 1D nonlinear advection equation

$$\frac{\partial f}{\partial t} + \frac{\partial}{\partial x}[af] = 0, \qquad f(t^0, x) = f^0(x), \tag{3}$$

where $f : [0,+\infty[\times\Omega \longrightarrow \mathbb{R}$ and $a : [0,+\infty[\times\Omega \longrightarrow \mathbb{R}$, admits solutions, whose existence and uniqueness are discussed, for instance, in [28]. Problem (3) is solved through the characteristics as

$$f(t,x) = f(s,\mathcal{X}(s;t,x))J(s;t,x) \quad \forall (s,t,x) \in [0,+\infty[\times[0,+\infty[\times\Omega, \tag{4}$$

where

$$\frac{\mathrm{d}\mathcal{X}}{\mathrm{d}s} = a(s,\mathcal{X}(s;t,x)), \qquad \mathcal{X}(t;t,x) = x, \qquad J(s;t,x) := \frac{\partial \mathcal{X}(s;t,x)}{\partial x} \tag{5}$$

are the characteristics and their Jacobian. Integrating (4) against the test function φ on an arbitrary set $K \subseteq \Omega$, we obtain

$$\int_K f(t,x)\varphi(x)\mathrm{d}x = \int_K f\left(s,\mathcal{X}(s;t,x)\right) J(s;t,x)\varphi(x)\mathrm{d}x, \tag{6}$$

which, by changing variables

$$y = \mathcal{X}(s;t,x), \qquad x = \mathcal{X}(t;s,y),$$

rewrites as

$$\int_K f(t,x)\varphi(x)\mathrm{d}x = \int_{\mathcal{X}(s;t,K)} f(s,y)\varphi(\mathcal{X}(t;s,y))\mathrm{d}y. \tag{7}$$

If we now let in (5), (6) and (7)

$$K = I_i, \qquad t = t^{n+1}, \qquad s = t^n, \qquad \varphi(x) = \varphi_{i,j}(x)$$

and introduce notations

$$x^b := \mathcal{X}(t^n;t^{n+1},x), \qquad x^f := \mathcal{X}(t^{n+1};t^n,x), \qquad I_i^b := \mathcal{X}(t^n;t^{n+1},I_i),$$

where b stands for "back" and f for "forth", then (6) and (7) rewrite as the backward and the forward formulae:

$$\begin{aligned} \int_{I_i} f^{n+1}(x)\varphi_{i,j}(x)\mathrm{d}x &= \int_{I_i} f^n(x^b)J(x^b)\varphi_{i,j}(x)\mathrm{d}x & (8)\\ &= \int_{I_i^b} f^n(x)\varphi_{i,j}(x^f)\mathrm{d}x. & (9) \end{aligned}$$

In the following, we describe in detail the affine approximation for the solution of the characteristics, then we develop the forward formula (9) on our DG setting; finally, we show that the forward formula (9) and the backward formula (8) give rise to the same numerical scheme.

2.2.1. Computation of the characteristics

We propose an affine approximation for the solution of the characteristics, in order to fulfill the the reversibility constraint

$$(x^b)^f = x, \ \forall x \in \Omega.$$

Compute the feet of the characteristics from the Gauss points $\{x^b_{i,j}\}$ through any ODE solver: for the scope of this work, we use the Runge-Kutta multi-stage integrator. Compute the feet from the cell extrema $\{x_{i,-1}\}$ and $\{x_{i,d+1}\}$ by means of the two neighboring Gauss points

$$x^b_{i,-1} = \frac{x^b_{i-1,d} + x^b_{i,0}}{2}, \qquad x^b_{i,d+1} = \frac{x^b_{i,d} + x^b_{i+1,0}}{2}, \qquad \text{for } i = 0, \dots, N-1,$$

where, using periodicity, by convention we have set $x^b_{-1,d+1} := x^b_{N-1,d+1} - \lambda(\Omega)$ and $x^b_{N,0} := x^b_{0,0} + \lambda(\Omega)$, with $\lambda(\Omega) := x_{\max} - x_{\min}$ the length of the domain Ω. The affine approximation we propose consists of a linear interpolation between the two neighboring feet of the characteristics from the Gauss points: if

$$x \in]x_{i,j}, x_{i,j+1}[\qquad \exists (i,j) \in \{0, \dots, N\} \times \{-1, \dots, d+1\}$$

we can express it as

$$x = (1-u)x_{i,j} + u x_{i,j+1} \qquad \exists u \in (0,1).$$

We define the characteristic backward in time from x as

$$x^b := (1-u)x^b_{i,j} + u x^b_{i,j+1}.$$

Moreover, we can explicitly computed the backward Jacobian:

$$J(x^b) := \frac{\partial x^b}{\partial x} = \frac{x^b_{i,j+1} - x^b_{i,j}}{x_{i,j+1} - x_{i,j}}. \tag{10}$$

In the other direction, if

$$x \in]x^b_{i,j}, x^b_{i,j+1}[\qquad \exists (i,j) \in \{0, \dots, N\} \times \{-1, \dots, d+1\},$$

then

$$x = (1-u)x^b_{i,j} + u x^b_{i,j+1} \qquad \exists u \in (0,1)$$

and we define the characteristic forward in time from x as

$$x^f := (1-u)x_{i,j} + u x_{i,j+1}. \tag{11}$$

It is easy to check that the reversibility constraint $(x^b)^f = x$ is fulfilled.

2.2.2. Forward scheme

We now describe in detail the scheme to which formula (9) gives rise. Inject the representation of f (2) into the forward formula (9):

$$\begin{aligned} f_{i,j}^{n+1}\omega_j\frac{\Delta x}{2} &= \int_{I_i^b} f^n(x)\varphi_{i,j}(x^f)\mathrm{d}x, \\ &= \sum_{i',j'} f_{i',j'}^n \int_{I_i^b} \varphi_{i',j'}(x)\varphi_{i,j}(x^f)\mathrm{d}x, \\ &= \sum_{i',j'} f_{i',j'}^n \sum_{m=-1}^{d} \int_{x_{i,m}^b}^{x_{i,m+1}^b} \varphi_{i',j'}(x)\varphi_{i,j}(x^f)\mathrm{d}x, \end{aligned}$$

change variables $x = (1-u)x_{i,m}^b + ux_{i,m+1}^b$,

$$\begin{aligned} f_{i,j}^{n+1}\omega_j\frac{\Delta x}{2} &= \sum_{i',j'} f_{i',j'}^n \sum_{m=-1}^{d} \int_0^1 \varphi_{i',j'}((1-u)x_{i,m}^b + ux_{i,m+1}^b) \\ &\times \quad \varphi_{i,j}((1-u)x_{i,m} + ux_{i,m+1})\mathrm{d}u(x_{i,m+1}^b - x_{i,m}^b), \end{aligned} \tag{12}$$

where we have used the affine expression (11) for x^f. In order to evaluate this expression, let us focus on computing

$$\mathcal{A} := \sum_{i'} f_{i',j'}^n \int_0^1 \varphi_{i',j'}((1-u)x_{i,m}^b + ux_{i,m+1}^b)\varphi_{i,j}((1-u)x_{i,m} + ux_{i,m+1})\mathrm{d}u,$$

which, in the maximum generality, decouples into three sums, because the feet of the characteristics $x_{i,m}^b$ and $x_{i,m+1}^b$ might be separated by a whole cell, as sketched in Figure 1(i):

$$\mathcal{A} = f_{s_{i,m},j'}^n \int_0^1 \varphi_{s_{i,m},j'}((1-u)x_{i,m}^b + ux_{i,m+1}^b)\varphi_{i,j}((1-u)x_{i,m} + ux_{i,m+1})\mathrm{d}u \tag{13}$$

$$+ \sum_{i'=s_{i,m}+1}^{s_{i,m+1}-1} f_{i',j'}^n \int_0^1 \varphi_{i',j'}((1-u)x_{i,m}^b + ux_{i,m+1}^b)\varphi_{i,j}((1-u)x_{i,m} + ux_{i,m+1})\mathrm{d}u \tag{14}$$

$$+ f_{s_{i,m+1},j'}^n \int_0^1 \varphi_{i',j'}((1-u)x_{i,m}^b + ux_{i,m+1}^b)\varphi_{i,j}((1-u)x_{i,m} + ux_{i,m+1})\mathrm{d}u \tag{15}$$

where we have set $s_{i,m}$ the index for which $x_{i,m}^b \in I_{s_{i,m}}$. We now change variables in the three pieces of the integral: for (13)

$$(1-u)x_{i,m}^b + ux_{i,m+1}^b = (1-v)x_{i,m}^b + vx_{s_{i,m}+1/2},$$

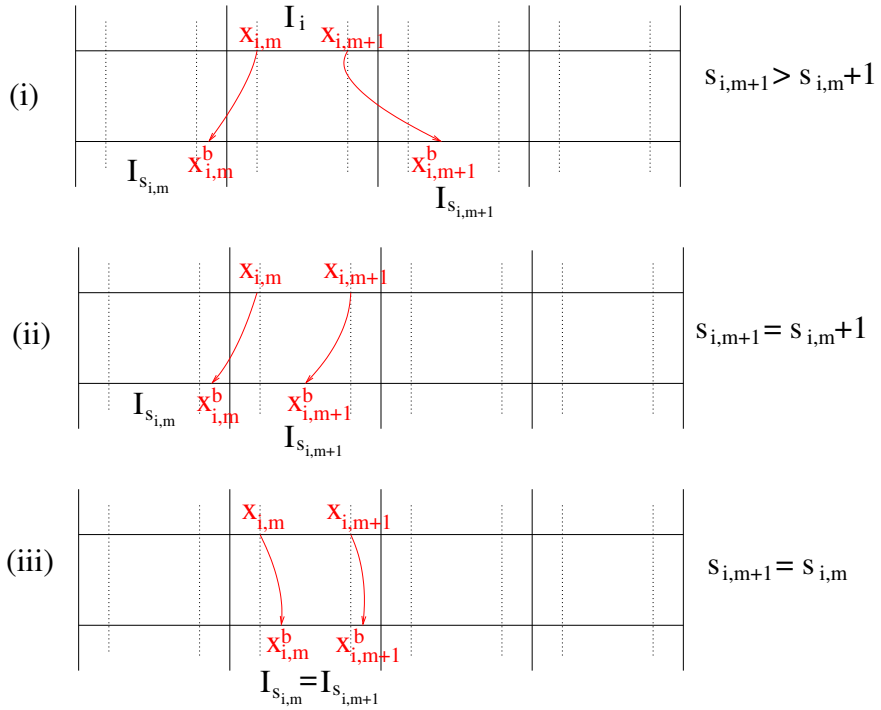


Figure 1: **Backward characteristics.** The feet of the characteristics $x^b_{i,m}$ and $x^b_{i,m+1}$ from points $x_{i,m}$ and $x_{i,m+1}$ may give rise to three different configurations: (i) they are separated by a whole cell; (ii) they belong to contiguous cells; (iii) they belong to the same cell.

for (14),

$$(1-u)x^b_{i,m} + ux^b_{i,m+1} = (1-v)x_{i'-1/2} + vx_{i'+1/2},$$

whereas for (15)

$$(1-u)x^b_{i,m} + ux^b_{i,m+1} = (1-v)x_{s_{i,m+1}-1/2} + vx^b_{i,m+1}.$$

This gives

$$\begin{aligned}
\mathcal{A} &= \frac{x_{s_{i,m}+1/2}-x^b_{i,m}}{x^b_{i,m+1}-x^b_{i,m}} f^n_{s_{i,m},j'} \int_0^1 \mathrm{d}v \varphi_{s_{i,m},j'}((1-v)x^b_{s_{i,m},m} + v x_{s_{i,m}+1/2}) \\
&\times \varphi_{i,j}\left(\left(1-\frac{x_{s_{i,m}+1/2}-x^b_{i,m}}{x^b_{i,m+1}-x^b_{i,m}}v\right)x_{i,m} + \frac{x_{s_{i,m}+1/2}-x^b_{i,m}}{x^b_{i,m+1}-x^b_{i,m}} v x_{i,m+1}\right) \\
&+ \frac{\Delta x}{x^b_{i,m+1}-x^b_{i,m}} \sum_{i'=s_{i,m}+1}^{s_{i,m+1}-1} f^n_{i',j'} \int_0^1 \mathrm{d}v \varphi_{i',j'}((1-v)x_{i'-1/2} + v x_{i'+1/2}) \\
&\times \varphi_{i,j}\left(\left(1-\frac{\Delta x}{x^b_{i,m+1}-x^b_{i,m}}v\right)x_{i,m} + \frac{\Delta x}{x^b_{i,m+1}-x^b_{i,m}} v x_{i,m+1}\right) \qquad (16) \\
&+ \frac{x^b_{i,m+1}-x_{s_{i,m+1}-1/2}}{x^b_{i,m+1}-x^b_{i,m}} f^n_{s_{i,m+1},j'} \int_0^1 \mathrm{d}v \varphi_{s_{i,m+1},j'}((1-v)x_{s_{i,m+1}-1/2} + v x^b_{i,m+1}) \\
&\times \varphi_{i,j}\left(\left(1-\frac{x^b_{i,m+1}-x_{s_{i,m+1}-1/2}}{x^b_{i,m+1}-x^b_{i,m}}v\right)x_{i,m} + \frac{x^b_{i,m+1}-x_{s_{i,m+1}-1/2}}{x^b_{i,m+1}-x^b_{i,m}} v x_{i,m+1}\right).
\end{aligned}$$

Remark that the situation in Figure 1(ii) implies that (14) is zero. Moreover, if $s_{i,m+1} = s_{i,m}$ as in Figure 1(iii), formula (16) is not valid anymore, and the following must be used instead:

$$\mathcal{A} = f^n_{s_{i,m},j'} \int_0^1 \varphi_{s_{i,m},j'}((1-u)x^b_{i,m} + u x^b_{i,m+1}) \varphi_{i,j}\left((1-u)\,x_{i,m} + u x_{i,m+1}\right) \mathrm{d}u. \qquad (17)$$

Formulae (16) and (17) are computed exactly through a Gauss quadrature.

2.2.3. Backward scheme

If we inject the representation of f (2) into the backward formula (8)

$$\begin{aligned}
f^{n+1}_{i,j}\omega_j \frac{\Delta x}{2} &= \int_{I_i} J(x^b) f^n(x^b) \varphi_{i,j}(x) \mathrm{d}x \\
&= \sum_{i',j'} f^n_{i',j'} \int_{I_i} J(x^b) \varphi_{i',j'}(x^b) \varphi_{i,j}(x) \mathrm{d}x \\
&= \sum_{i',j'} f^n_{i',j'} \sum_{m=-1}^{d} \int_0^1 \varphi_{i',j'}((1-u)x^b_{i,m+1} + u x^b_{i,m}) \\
&\times \varphi_{i,j}((1-u)x_{i,m+1} + u x_{i,m}) \mathrm{d}u (x^b_{i,m+1} - x^b_{i,m}),
\end{aligned}$$

we recover formula (12) and the resulting scheme coincides with the forward scheme.

2.2.4. L^p-norms and conservation properties

As $L^p(\Omega)$-norms and mass for the distribution function f^n we shall use

$$\|f^n\|_{L^p(\Omega)} = \left(\frac{\Delta x}{2}\sum_{i,j}\omega_j|f^n_{i,j}|^p\right)^{1/p}, \quad M(f^n) = \frac{\Delta x}{2}\sum_{i,j}\omega_j f^n_{i,j}.$$

It can be checked that the scheme is by construction conservative:

$$M(f^n) = M(f^0).$$

2.3. 2D nonlinear advection

The 2D nonlinear advection problem

$$\frac{\partial f}{\partial t} + \frac{\partial}{\partial x}\left[a_1 f\right] + \frac{\partial}{\partial y}\left[a_2 f\right] = 0, \qquad f(t^0, x, y) = f^0(x, y), \tag{18}$$

is discretized as a tensor product of the 1D problem. The computational domain $\Omega = [x_{\min}, x_{\max}] \times [y_{\min}, y_{\max}] \subseteq \mathbb{R}^2$ is partitioned into rectangles

$$\Omega = \bigcup_{i=0}^{N_x-1}\bigcup_{k=0}^{N_y-1}\mathcal{E}_{i,k},$$

where

$$\begin{aligned}\mathcal{E}_{i,k} := I_i \times J_k &= [x_{i-1/2}, x_{i+1/2}] \times [y_{k-1/2}, y_{k+1/2}] \\ &= [x_{i-1/2}, x_{i-1/2} + \Delta x] \times [y_{k-1/2}, y_{k-1/2} + \Delta y].\end{aligned}$$

The discontinuous finite element space is given by

$$\begin{aligned}V^d_{\Delta x,\Delta y} = \big\{&\omega \in L^2(\Omega) \text{ s.t. } \omega(x,y) = \varphi(x)\psi(y), \\ &\varphi \in \mathbb{P}^d(I_{i_1}),\ i_1 = 0, \ldots, N_x - 1, \\ &\psi \in \mathbb{P}^d(J_{i_2}),\ i_2 = 0, \ldots, N_y - 1\big\}.\end{aligned}$$

The basis for this space is the tensor product of the bases for $V^d_{\Delta x}$ and $V^d_{\Delta y}$:

$$\mathcal{B} = \left\{\varphi_{i_1,j_1}(x)\psi_{i_2,j_2}(y)\right\}_{(i_1,j_1,i_2,j_2)\in\{0,\ldots,N_x-1\}\times\{0,\ldots,d\}\times\{0,\ldots,N_y-1\}\times\{0,\ldots,d\}}.$$

As for the time discretization, instead of solving a real 2D problem, we reduce to the solution of 1D problems through the second-order Strang-splitting

scheme [2, 3], together with a *predictor/corrector* strategy in order to keep the good order-in-time; to advance from time t^n to time t^{n+1}, we need an approximation of the advection field $\boldsymbol{a}(t,x,y) := (a_1(t,x,y), a_2(t,x,y))$ at time $t^{n+1/2}$:

$$\left.\begin{array}{l} 1.\ \text{Get } \boldsymbol{a}(t^n) \\ 2.\ \text{Perform a } \Delta t/2\text{-time step along } x \\ 3.\ \text{Perform a } \Delta t/2\text{-time step along } y \\ 4.\ \text{Get } \boldsymbol{a}(t^{n+1/2}) \end{array}\right\} \text{prediction}$$

$$\left.\begin{array}{l} 5.\ \text{Perform a } \Delta t/2\text{-time step along } x \\ 6.\ \text{Perform a } \Delta t\text{-time step along } y \\ 7.\ \text{Perform a } \Delta t/2\text{-time step along } x \end{array}\right\} \text{correction}$$

As we use a predictor/corrector strategy, the time step can be adapted during the simulation: we have chosen to constrain it in such a way that

$$x^b_{i,m} \in \bigcup_{i'=i-1}^{i+1} I_{i'}, \quad \forall (i,m) \in \{0,\dots,N_x-1\} \times \{0,\dots,d\}$$

and

$$y^b_{k,m} \in \bigcup_{k'=k-1}^{k+1} J_{k'}, \quad \forall (k,m) \in \{0,\dots,N_y-1\} \times \{0,\dots,d\}$$

during all the advection steps. In other words, this means that we limit the displacements of the characteristics to at most one cell. We want to stress that an advantage in using the DG strategy is the fact that this constraint on the time step is not so heavy because it depends only on the cell size and not on the local refinement due to the Gauss points; moreover, it might be useful in view of a parallelization of the code, and it is also meant to improve the accuracy of the computation of the characteristics. We emphasize that this constraint is reasonable and might sometimes even be needed to achieve an acceptable precision of the solution.

3. Validation of the method

In this section, we validate our 1D solver against the model problem

$$\frac{\partial f}{\partial t} + \frac{\partial}{\partial x}\left(\sin(x) f\right) = 0$$

by showing the order-in-space and the order-in-time, and the influence that the ODE integrator for the characteristics has. Then, we pass to the 2D case with the swirling deformation flow benchmark

$$\frac{\partial f}{\partial t} + \frac{\partial}{\partial x}\left(\sin^2(\pi x)\sin(2\pi y)g(t)f\right) + \frac{\partial}{\partial y}\left(-\sin^2(\pi y)\sin(2\pi x)g(t)f\right) = 0$$

and visually show how important the order d of the method is for the resolution of the distribution function.

3.1. 1D nonlinear advection

We test the scheme detailed in Section 2.2 on the benchmark equation

$$\frac{\partial f}{\partial t} + \frac{\partial}{\partial x}\left(\sin(x)f\right) = 0, \qquad f_0(x) = 1, \qquad x \in [0, 2\pi],$$

which admits as analytical solution

$$f(t,x) = \frac{e^{-t}}{\left(1 + \tan^2\left(\frac{x}{2}\right)e^{-2t}\right)\cos^2\left(\frac{x}{2}\right)}.$$

In Figure 2 we plot the error we commit with respect to the the number of points, in log-log scale; we observe that the convergence degenerates before we reach the machine-error (which is about 10^{-12}) because we are constrained by the numerical solution of the characteristics. If, instead, we use the analytic solution, we see in Figure 3 that we can achieve much smaller error magnitudes. With more detail, Figure 4 shows how important the accuracy is when solving the characteristics: the higher the order, the later the error saturates. Up to $d = 3$, using RK3 seems to provide a reasonable accuracy; for higher orders, using RK4 seems more appropriate. Finally, in Figure 3 we estimate the order-in-space of the method by measuring the slope of the lines for the exact solution of the characteristics.

In Figure 5 we plot the error committed by the method, with fixed $N = 128$ and $d = 4$, with respect to the time step Δt. The convergence rate is given by the accuracy of the solution of the characteristics. When we are exact in time the error keeps the same order and does essentially not depend on Δt, which means that errors do not accumulate, because the increasing number of iterations due to Δt is compensated by the improving accuracy in the solution of the transport due to values of $x^b_{i,m}$ and $x^b_{i,m+1}$ converging to $x_{i,m}$ and $x_{i,m+1}$ as $\Delta t \to 0$.

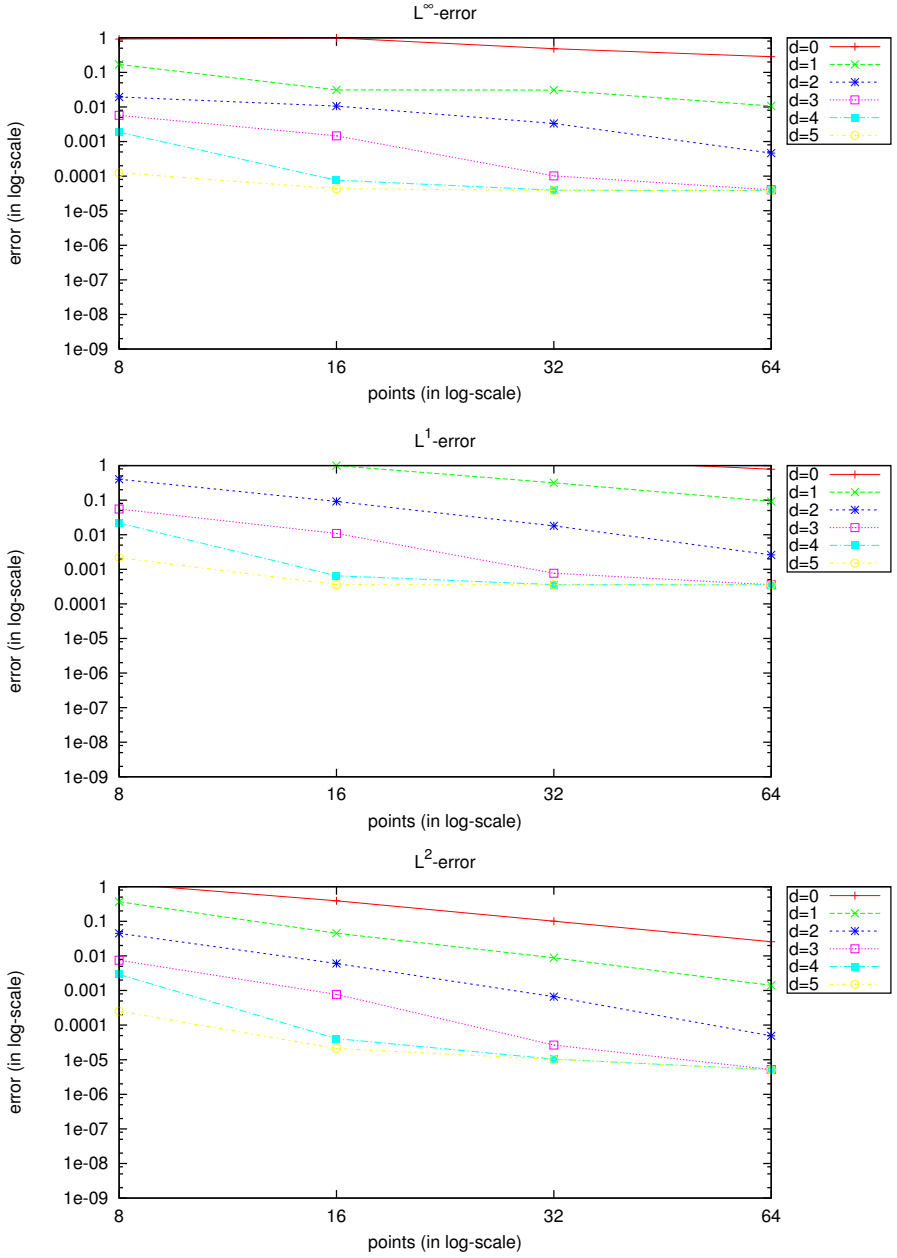


Figure 2: **1D nonlinear advection.** The numerically evaluated order-in-space of the method for problem $\frac{\partial f}{\partial t} + \frac{\partial}{\partial x}(\sin(x)f) = 0$, solved in the interval $x \in [0, 2\pi]$ with periodic boundary conditions. Here the characteristics are evaluated numerically through the second order Runge-Kutta scheme.

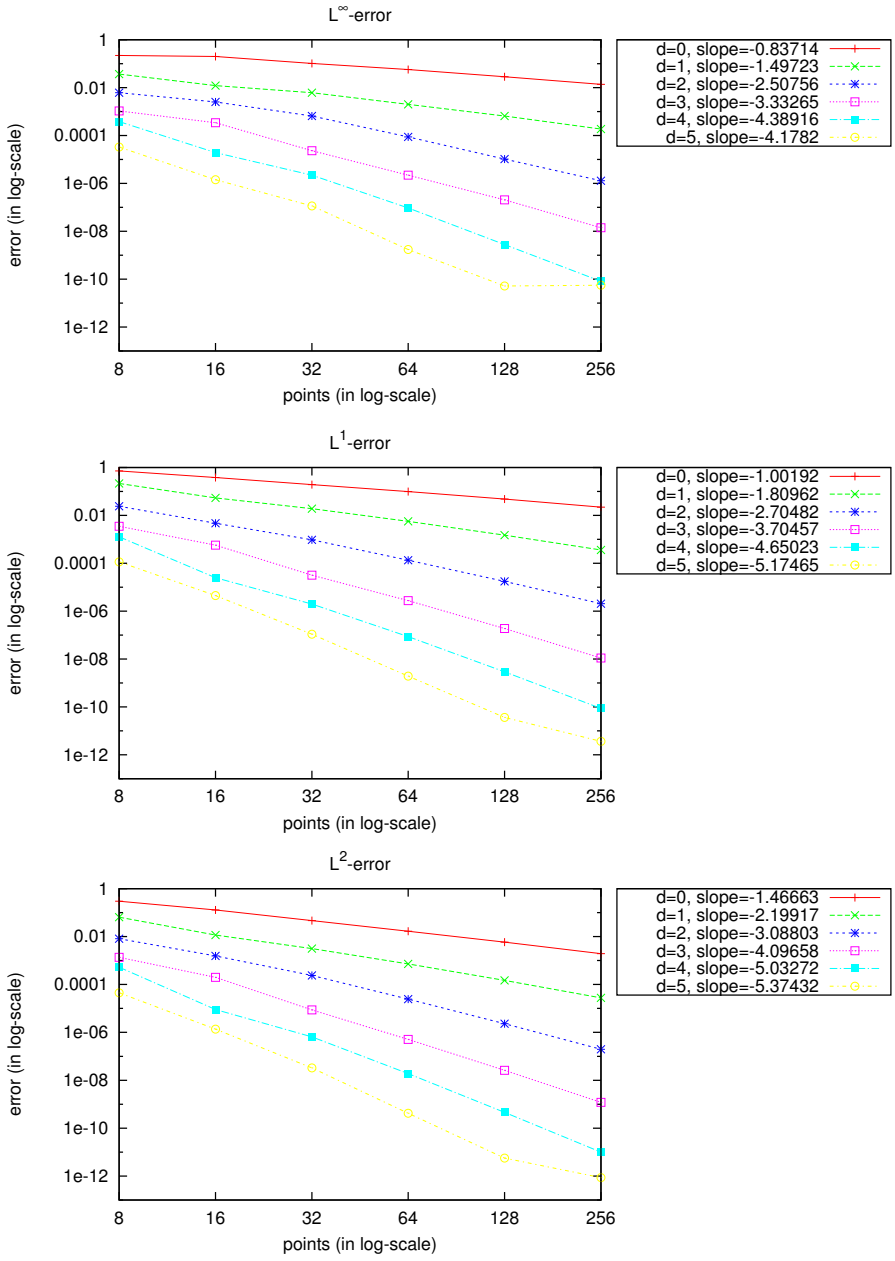


Figure 3: **1D nonlinear advection.** The numerically evaluated order-in-space of the method for problem $\frac{\partial f}{\partial t} + \frac{\partial}{\partial x}(\sin(x) f) = 0$, solved in the interval $x \in [0, 2\pi]$ with periodic boundary conditions. Here the analytic solution of the characteristics is used.

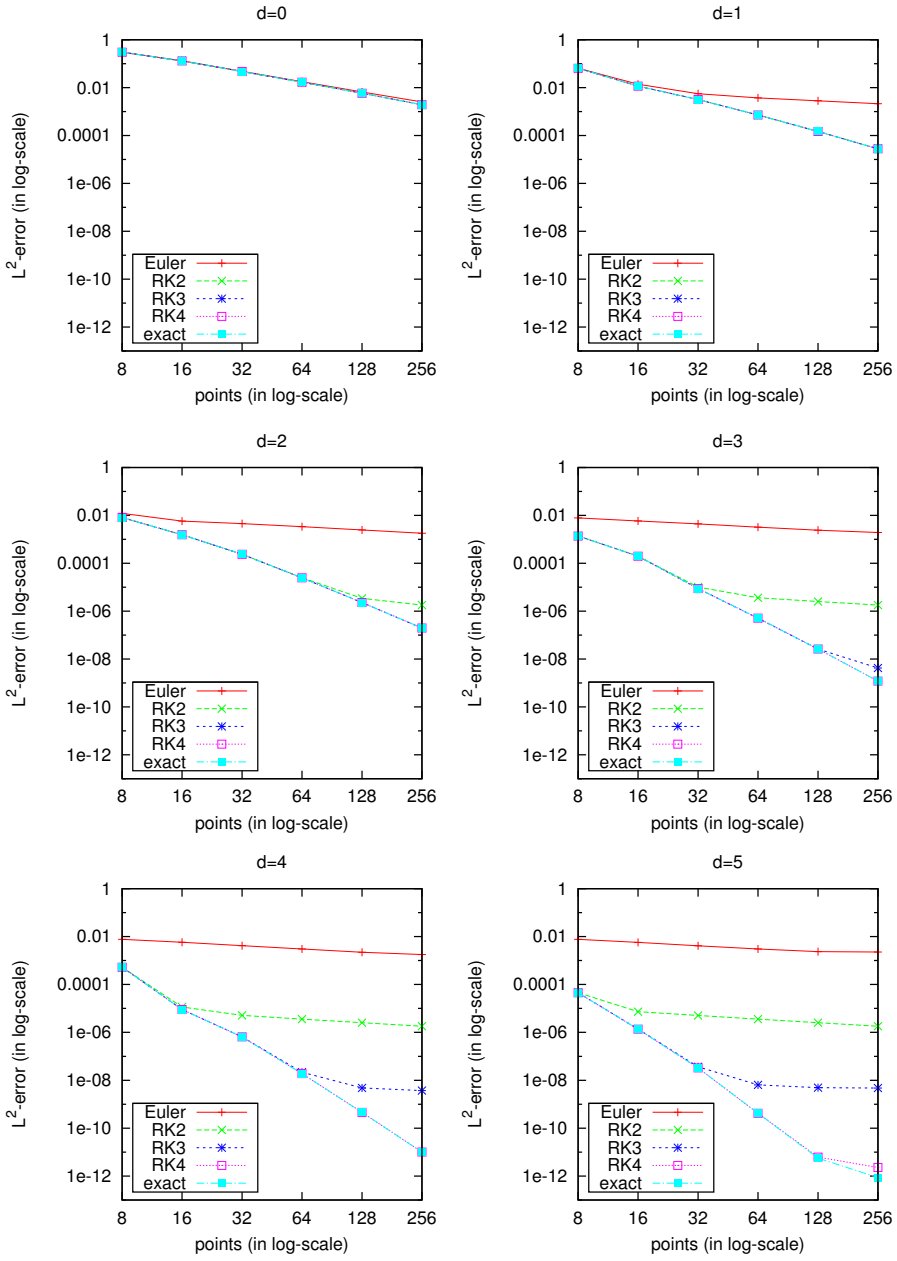


Figure 4: **1D nonlinear advection.** The numerically evaluated order-in-space of the method for problem $\frac{\partial f}{\partial t} + \frac{\partial}{\partial x}(\sin(x) f) = 0$, solved in the interval $x \in [0, 2\pi]$ with periodic boundary conditions, for different accuracy in the solution of the characteristics.

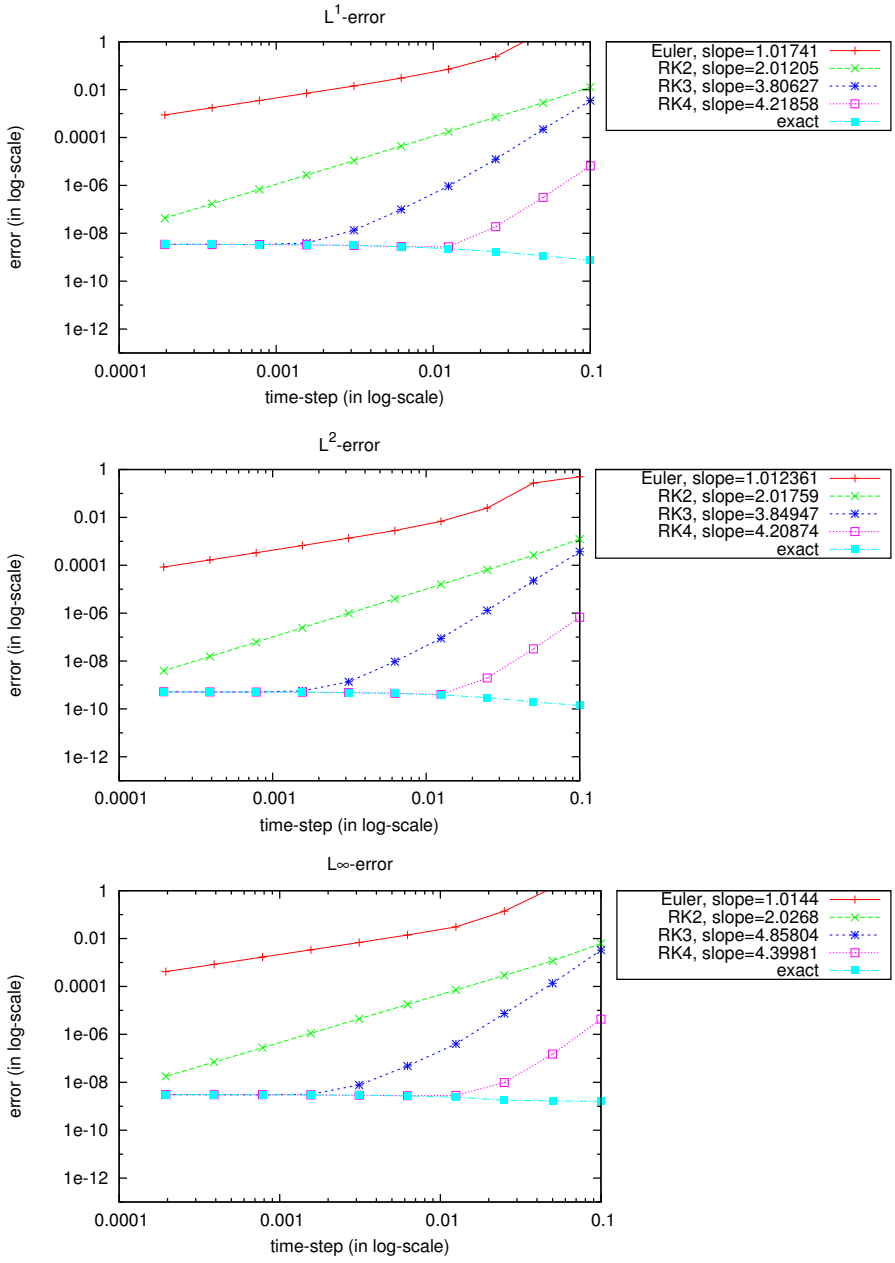


Figure 5: **1D nonlinear advection.** The numerically evaluated order-in-time of the method for problem $\frac{\partial f}{\partial t} + \frac{\partial}{\partial x}(\sin(x) f) = 0$, solved in the interval $x \in [0, 2\pi]$ with periodic boundary conditions, for different time-steps Δt.

3.2. 2D nonlinear advection

We test the scheme for the 2D nonlinear advection detailed in Section 2.3, against the swirling deformation flow benchmark test [29]

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

in the interval $(x, y) \in [0, 1] \times [0, 1]$. In Figure 6 we plot the results for $g(t) = \cos\left(\frac{\pi t}{T}\right)$: we know that the shape of the initial function is recovered, after deformations by alternate clockwise and counterclockwise twisting, at times $\mathbb{Z}T$; in our tests, $T = 1.5$ and the results shown in Figure 6 are for $t = 2T = 3$, so that the initial shapes have been twisted and then restored. We compare the evolutions of the system for different degrees d of the method: the higher the order of the method, the more accurate the results. In Figure 7 we plot the results for $g(t) = 1$ and initial function

$$f^0(x, y) = \begin{cases} 1 & \text{if } (x-1)^2 + (y-1)^2 < 0.8 \\ 0 & \text{else.} \end{cases} \tag{19}$$

We see that the higher the order, the better the resolution of the function, plotted at time $t = 2.5$. Nevertheless, the method does not prove well designed as for the preservation of maxima and minima, as can be seen in Figure 7 (b): values appear over 1.1 and below -0.1, even for a quite high order and exactly solved characteristics. The shocks are not well resolved, and some oscillations appear: a further step to improve our scheme is its modification in order to better deal with sharp edges by means of limiter procedures [15, 25, 24, 30].

4. Simulations of the guiding-center model

In order to validate our code, we compare it with the classical Backward Semi-Lagrangian (BSL) scheme [31]. We use for the DG method $\Delta t = 0.01$ in order to prevent the code itself from reducing the step; this way we are sure we get the density at multiples of $\Delta t = 0.01$, as for the BSL code. Should we have let the DG solver free to adapt the step, it would have reduced it up to $\Delta t \approx 0.02$. The BSL solver allows us to use larger time steps, e.g. $\Delta t = 0.1$, while the empirical experiments on the DG solver without the constraint of one-cell displacement have shown inaccurate results. In Figure 8 and Figure 9 we plot the density $\rho(t, x_1, x_2)$ at times $t = 30$ and $t = 60$ and the evolution of some reference scalar magnitudes for both methods. The DG solver recovers, qualitatively and quantitatively, comparable results.

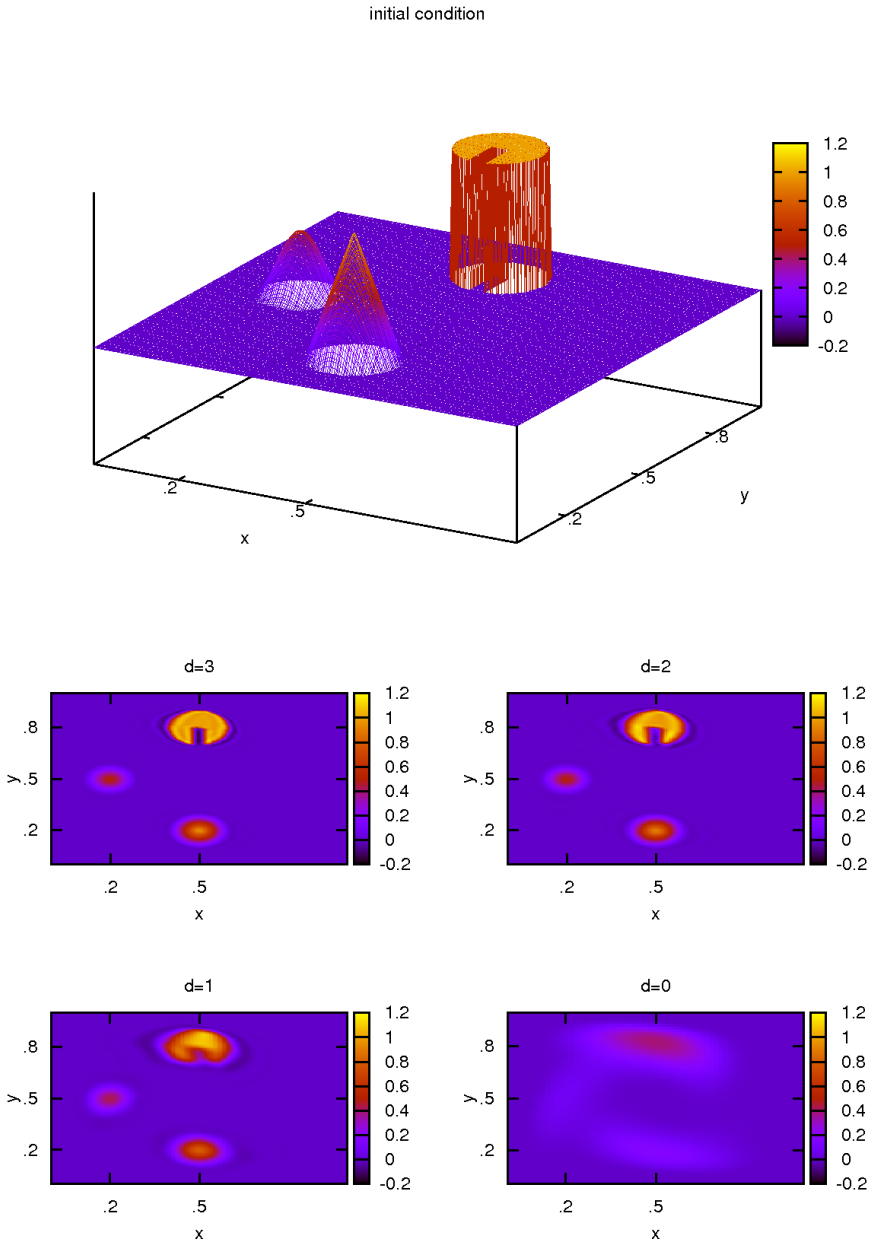
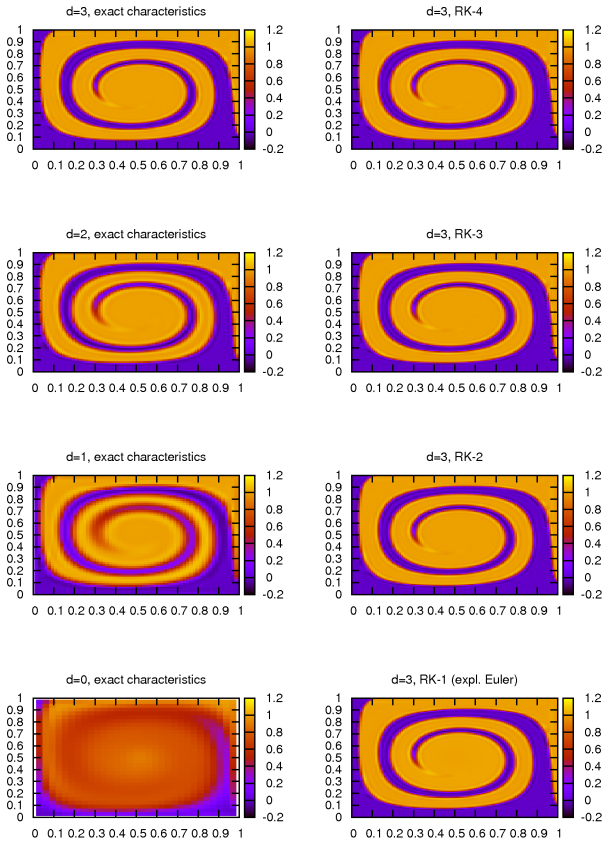
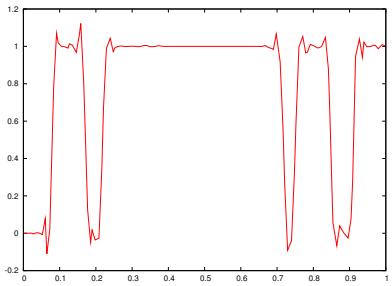


Figure 6: **2D nonlinear advection.** The swirling flow with $g(t) = \cos\left(\frac{\pi t}{T}\right)$, $T = 1.5$. The grid is 32×32, $\Delta t = 0.001$. Top: the initial function. Bottom: the evolution at time T, for different degrees d. Here $N = 64$, $\Delta t = 0.001$, the characteristics are solved by RK-4.



(a)



(b)

Figure 7: **2D nonlinear advection.** The swirling flow with $g(t) = 1$. The grid is 32×32, $\Delta t = 0.001$. The initial function is (19) (a): The solution at time $t = 2.5$ in the (x, y)-space. (b): A cut of the solution for $y = 0.5$ at time $t = 2.5$, for $d = 3$ and exactly solved characteristics.

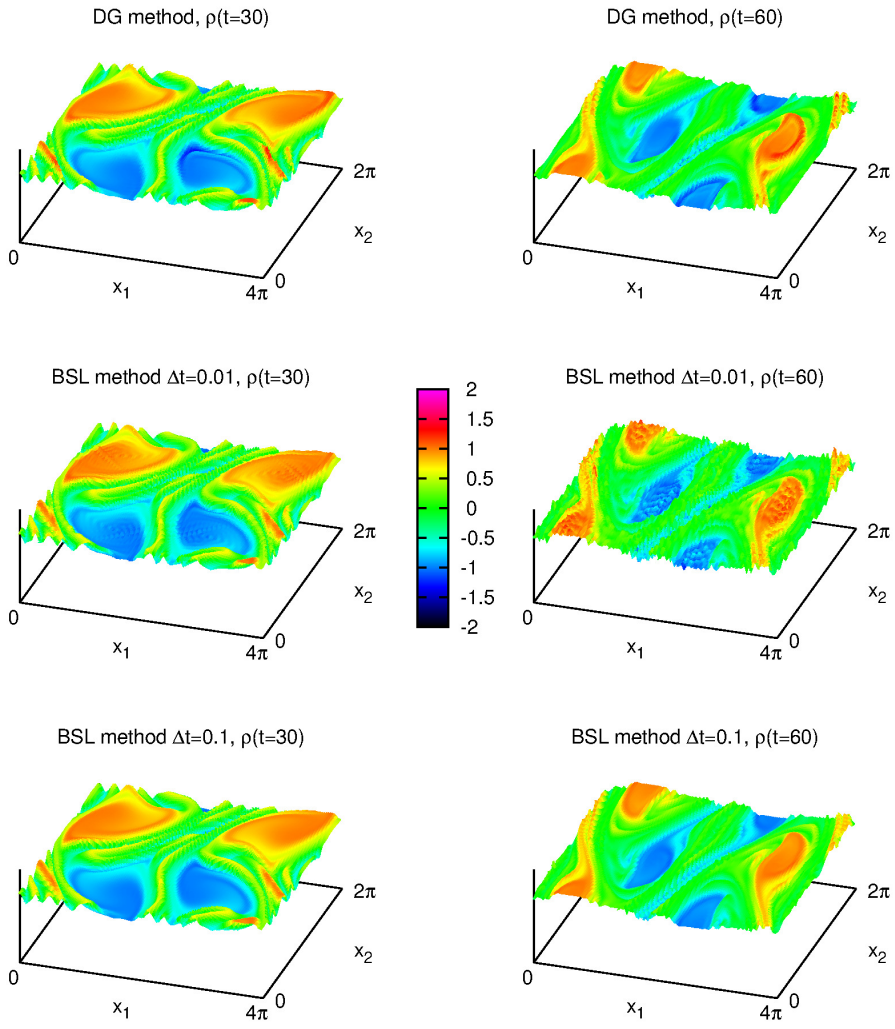


Figure 8: **Guiding-center.** Evolution of the density $\rho(x_1, x_2)$, drawn at time $t = 30$ and $t = 60$. The time step is $\Delta t = 0.01$ for DG. The meshes are $(N_x, N_y, d) = (32, 32, 3)$ for DG, $(N_x, N_y) = (128, 128)$ for BSL.

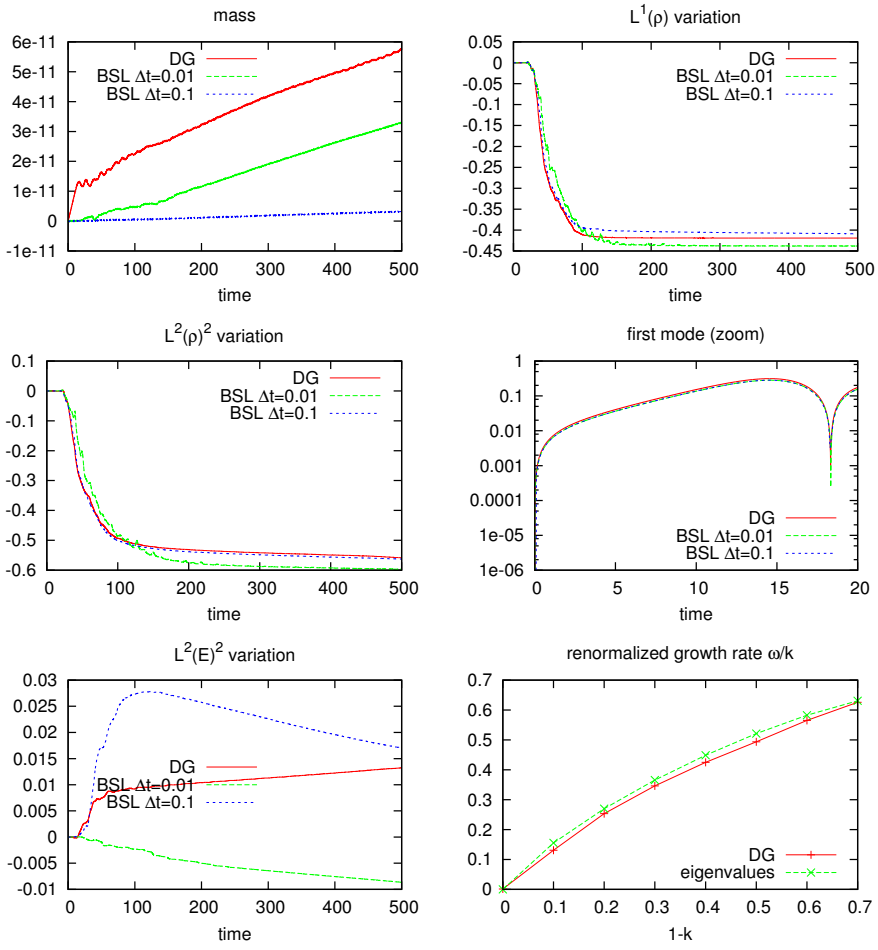


Figure 9: **Guiding-center.** Behavior of some scalar magnitudes. Top left: time-evolution of the total mass. Top right: time-evolution of the $\|\rho(t)\|_{L^1}$ variation. Center left: time-evolution of the $\|\rho(t)\|_{L^2}$ (enstrophy) variation. Center right: time-evolution of the first Fourier mode $\|\phi_{k=0.5}(t)\|_{L^2}$ Bottom left: time-evolution of the $\|E(t)\|_{L^2}$ variation. Bottom right: normalized growth rate ω/k against $k-1$. The time step is $\Delta t = 0.01$ for DG. The meshes are $(N_x, N_y, d) = (32, 32, 3)$ for DG, $(N_x, N_y) = (128, 128)$ for BSL.

5. Conclusions and future plans

Our SLDG scheme has been successfully applied to the simulation of a guiding-center model. We plan to improve our work in mainly three directions:

- Parallelization of the scheme. We want to exploit the favorable features of local refinement provided by the DG discretization to implement a parallel solver.

- Divergence-free version. By now, we have constructed a variant of the scheme that has the weak property

  $$f^n_{i,j,k,\ell} = 1 \ \forall (i,j,k,\ell) \quad \Rightarrow \quad \sum_{j,\ell} \omega_j \omega_\ell f^{n+1}_{i,j,k,\ell} = 1 + \mathcal{O}(\Delta t^2) \ \forall (i,k) :$$

  if the distribution function is constant at any Gauss point at time t^n, then, at first order, at time t^{n+1} that constant will be preserved as cell averages but not at any Gauss point (see AppendixB). Numerical evidence shows that

  $$\sum_{j,\ell} \omega_j \omega_\ell f^n_{i,j,k,\ell} = 1 \ \forall (i,k) \quad \not\Rightarrow \quad \sum_{j,\ell} \omega_j \omega_\ell f^{n+1}_{i,j,k,\ell} = 1 + \mathcal{O}(\Delta t^2) \ \forall (i,k),$$

  therefore after one time step the conservation property is completely lost. We want to improve the numerical scheme in order to conserve the constants as cell averages, i.e.

  $$\sum_{j,\ell} \omega_j \omega_\ell f^n_{i,j,k,\ell} = 1 \ \forall (i,k) \quad \Rightarrow \quad \sum_{j,\ell} \omega_j \omega_\ell f^{n+1}_{i,j,k,\ell} = 1 + \mathcal{O}(\Delta t^2) \ \forall (i,k).$$

- Coupling to an Adaptive Mesh Refinement strategy. A priori, there is no reason why the degree d should be the same for all the cells I_i. This opens to the possibility to a coupling with an AMR strategy: depending on the features of the solution $f(t,x,y)$, different zones of the domain Ω can be given more or less resolution, thus reducing the computational times.

- Use of limiters. In the philosophy of [15, 25, 24, 30] we plan to add limiters in order to avoid the problems pointed out by Figure 7: maxima and minima are not preserved because the shocks are not properly resolved thus possibly making some oscillations appear.

AppendixA. Spectral solver for the Poisson equation

We now explain in detail how to compute, with spectral accuracy, the electric potential Φ and the electric field $E = -\nabla_{x_1,x_2}\Phi$ obtained by solving the Poisson equation

$$-\Delta_{x_1,x_2}\Phi = f.$$

We have a representation of f in the Fourier space as

$$f(x,y) = \sum_{(p,q)\in\mathbb{Z}\times\mathbb{Z}} a_{p,q} e^{ipx\frac{2\pi}{L_x}} e^{iqy\frac{2\pi}{L_y}}, \tag{A.1}$$

where

$$a_{p,q} = \frac{1}{L_x L_y}\int_{x_{\min}}^{x_{\max}}\int_{y_{\min}}^{y_{\max}} f(x,y) e^{-ipx\frac{2\pi}{L_x}} e^{-iqy\frac{2\pi}{L_y}}\mathrm{d}x\mathrm{d}y,$$

$L_x = x_{\max} - x_{\min}$ is the length of the x-domain, and $L_y = y_{\max} - y_{\min}$ is the length of the y-domain. After injecting the representation of $f(x,y)$ on the DG space (2)

$$a_{p,q} = \sum_{k_1,j_1,k_2,j_2} f_{k_1,j_1,k_2,j_2}\frac{1}{L_x L_y}\int_{x_{\min}}^{x_{\max}}\varphi_{k_1,j_1}(x)e^{-ipx\frac{2\pi}{L_x}}\mathrm{d}x\int_{y_{\min}}^{y_{\max}}\psi_{k_2,j_2}(y)e^{-iqy\frac{2\pi}{L_y}}\mathrm{d}y,$$

we change variables

$$(x,y) = (x_{k_1-1/2} + s_1\Delta x, y_{k_2-1/2} + s_2\Delta y), \tag{A.2}$$

to obtain

$$a_{p,q} = \frac{\Delta x\Delta y}{L_x L_y}\sum_{k_1',j_1',k_2',j_2'} f_{k_1',j_1',k_2',j_2'} e^{-ipx_{k_1'-1/2}\frac{2\pi}{L_x}} e^{-iqy_{k_2'-1/2}\frac{2\pi}{L_y}} I_{j_1',-p\frac{2\pi}{N_x}} I_{j_2',-q\frac{2\pi}{N_y}} \tag{A.3}$$

where we have set

$$I_{j,\alpha} := \int_0^1 \tilde{\varphi}^j(s)e^{i\alpha s}\mathrm{d}s.$$

As $E^{\{x,y\}} \in L^2(\Omega)$, they have a representation in the Fourier space:

$$E^\star(x,y) = \sum_{(p,q)\in\mathbb{Z}\times\mathbb{Z}} E^\star_{p,q} e^{ipx\frac{2\pi}{L_x}} e^{iqy\frac{2\pi}{L_y}}, \tag{A.4}$$

with

$$E^x_{p,q} = \begin{cases} 0 & \text{for } (p,q)=(0,0), \\ -\dfrac{ip\frac{2\pi}{L_x}a_{p,q}}{\left(p\frac{2\pi}{L_x}\right)^2+\left(q\frac{2\pi}{L_y}\right)^2} & \text{else,} \end{cases} \tag{A.5}$$

$$E^y_{p,q} = \begin{cases} 0 & \text{for } (p,q)=(0,0), \\ -\dfrac{iq\frac{2\pi}{L_y}a_{p,q}}{\left(p\frac{2\pi}{L_x}\right)^2+\left(q\frac{2\pi}{L_y}\right)^2} & \text{else,} \end{cases} \tag{A.6}$$

since we impose a zero average condition $\int_\Omega E^\star(x,y)\mathrm{d}x\mathrm{d}y = 0$. Its representation on the DG space will be denoted by

$$\tilde{E}^\star(x,y) = \sum_{k'_1,j'_1,k'_2,j'_2} E^\star_{k'_1,j'_1,k'_2,j'_2}\varphi_{k'_1,j'_1}(x)\psi_{k'_2,j'_2}(y),$$

with

$$E^\star_{k_1,j_1,k_2,j_2} = \frac{4}{\Delta x\Delta y\omega_{j_1}\omega_{j_2}}\int_{I_{k_1}\times J_{k_2}} E^\star(x,y)\varphi_{k_1,j_1}(x)\psi_{k_2,j_2}(y)\mathrm{d}x\mathrm{d}y. \tag{A.7}$$

In order to express $\{E^\star_{k_1,j_1,k_2,j_2}\}_{k_1,j_1,k_2,j_2}$ in terms of $\{f_{k_1,j_1,k_2,j_2}\}_{k_1,j_1,k_2,j_2}$, we inject (A.4) into (A.7):

$$\begin{aligned} &\left\{E^x_{k_1,j_1,k_2,j_2}, E^y_{k_1,j_1,k_2,j_2}\right\} \\ = \;& \frac{-4i}{\omega_{j_1}\omega_{j_2}N_xN_y}\sum_{k'_1,j'_1,k'_2,j'_2} f_{k'_1,j'_1,k'_2,j'_2}\sum_{(p,q)\neq(0,0)}\left(\frac{\left\{p\frac{2\pi}{L_x}, q\frac{2\pi}{L_y}\right\}}{\left(p\frac{2\pi}{L_x}\right)^2+\left(q\frac{2\pi}{L_y}\right)^2}\right) \\ &\times e^{ip\Delta x(k_1-k'_1)\frac{2\pi}{L_x}}e^{iq\Delta y(k_2-k'_2)\frac{2\pi}{L_y}}I_{j_1,p\frac{2\pi}{N_x}}I_{j'_1,-p\frac{2\pi}{N_x}}I_{j_2,q\frac{2\pi}{N_y}}I_{j'_2,-q\frac{2\pi}{N_y}}, \end{aligned}$$

where we have used (A.2), (A.3), (A.5) and (A.6). Let

$$(p,q) = (p'N_x + p'', q'N_y + q'')$$

the Euclidean division of p and q, then

$$E^{\star}_{k_1,j_1,k_2,j_2} = \frac{-4i}{\omega_{j_1}\omega_{j_2}N_xN_y}\sum_{p'',q''}\left(\sum_{j_1',j_2'} S^{\star}_{p'',j_1',j_1,q'',j_2',j_2}\mathcal{F}[f]_{p'',j_1',q'',j_2'}\right)e^{ip''k_1\frac{2\pi}{N_x}}e^{iq''k_2\frac{2\pi}{N_y}}$$

where we have set

$$S^{\{x,y\}}_{p'',j_1',j_1,q'',j_2',j_2} := \sum_{(p',q')\in\mathbb{Z}\times\mathbb{Z}}\begin{cases} 0 & \text{if } (p',p'',q',q'')=(0,0,0,0) \\ \frac{\left\{(p'N_x+p'')\frac{2\pi}{L_x},(q'N_y+q'')\frac{2\pi}{L_y}\right\}}{\left((p'N_x+p'')\frac{2\pi}{L_x}\right)^2+\left((q'N_y+q'')\frac{2\pi}{L_y}\right)^2} & \\ \times I_{j_1,(p'N_x+p'')\frac{2\pi}{N_x}}I_{j_1',-(p'N_x+p'')\frac{2\pi}{N_x}} & \\ \times I_{j_2,(q'N_y+q'')\frac{2\pi}{N_y}}I_{j_2',-(q'N_y+q'')\frac{2\pi}{N_y}} & \text{else} \end{cases}.$$

Introducing the following notations for the 2D Discrete Fourier Transform and its non-normalized AntiTransform

$$\begin{aligned}\mathcal{F}[f]_{p,j_1,q,j_2} &= \sum_{k_1'=0}^{N_x-1}\sum_{k_2'=0}^{N_y-1} f_{k_1',j_1,k_2',j_2}e^{-\frac{2\pi}{N_x}ipk_1'}e^{-\frac{2\pi}{N_y}iqk_2'} \\ \mathcal{F}^{-1}[f]_{k_1,j_1,k_2,j_2} &= \sum_{p=0}^{N_x-1}\sum_{q=0}^{N_y-1} f_{p,j_1,q,j_2}e^{\frac{2\pi}{N_x}ik_1p}e^{\frac{2\pi}{N_y}ik_2q},\end{aligned}$$

we can rewrite

$$\begin{aligned}\tilde{\rho}^{\star}_{p'',j_1,q'',j_2} &:= \sum_{j_1'=0}^{d}\sum_{j_2'=0}^{d}\mathcal{F}[f]_{p'',j_1',q'',j_2'}S^{\star}_{p'',j_1',j_1,q'',j_2',j_2}, \\ E^{\star}_{k_1,j_1,k_2,j_2} &= \frac{-4i}{\omega_{j_1}\omega_{j_2}N_xN_y}\mathcal{F}^{-1}[\tilde{\rho}^{\star}]_{k_1,j_1,k_2,j_2}. \end{aligned} \tag{A.8}$$

Computing the potential $\Phi(x,y)$ goes much the same as the computation of the electric field $E^{\{x,y\}}(x,y)$, the sole difference being the definition of the magnitude S

$$S^{\Phi}_{p'',j_1',j_1,q'',j_2',j_2} := \sum_{(p',q')\in\mathbb{Z}\times\mathbb{Z}}\begin{cases} 0 & \text{if } (p',p'',q',q'')=(0,0,0,0) \\ \frac{1}{\left((p'N_x+p'')\frac{2\pi}{L_x}\right)^2+\left((q'N_y+q'')\frac{2\pi}{L_y}\right)^2} & \\ I_{j_1,(p'N_x+p'')\frac{2\pi}{N_x}}I_{j_1',-(p'N_x+p'')\frac{2\pi}{N_x}} & \\ I_{j_2,(q'N_y+q'')\frac{2\pi}{N_y}}I_{j_2',-(q'N_y+q'')\frac{2\pi}{N_y}} & \text{else} \end{cases}.$$

Consequently, we are led to a multiplication in the Fourier space as in (A.8) and then an antitransformation as in (A.8):

$$\begin{aligned}\tilde{\rho}^{\Phi}_{p'',j_1,q'',j_2} &:= \sum_{j_1'=0}^{d}\sum_{j_2'=0}^{d}\mathcal{F}[f]_{p'',j_1',q'',j_2'}S^{\Phi}_{p'',j_1',j_1,q'',j_2',j_2},\\ \Phi_{k_1,j_1,k_2,j_2} &= \frac{-4i}{\omega_{j_1}\omega_{j_2}N_xN_y}\mathcal{F}^{-1}[\tilde{\rho}^{\Phi}]_{k_1,j_1,k_2,j_2}.\end{aligned}$$

AppendixB. Divergence-free condition (first order)

In this part, we sketch our preliminary efforts in order to get a scheme which is numerically divergence-free, so as to conserve the constants, which is not by now the case because the area of the volumes are not preserved through the splitting method. Assuming that $f^n(x,y)=1$, we compute the conditions on the velocity field that would ensure that

$$f^n(x,y)=1 \quad\Longrightarrow\quad \fint_{I_i\times J_k} f^{n+1}(x,y)\mathrm{d}x\mathrm{d}y = 1 \quad \forall(i,k)$$

at the first order in Δt.

AppendixB.1. Preliminary computations

Assume our model is

$$\frac{\partial f}{\partial t}+a_x\frac{\partial f}{\partial x}+a_y\frac{\partial f}{\partial y}=0,$$

and assume $f^n_{i,j,k,\ell}=1, \quad \forall(i,j,k,\ell)$. The backward scheme (8) coupled to a first-order Strang-splitting translates into

$$f^{n+1}_{i,j,k,\ell} = 1-\Delta t(\mathcal{B}_{i,j,k,\ell}+\mathcal{A}_{i,j,k,\ell})+\mathcal{O}(\Delta t^2), \tag{B.1}$$

with

$$\begin{aligned}\mathcal{A}_{i,j,k,\ell} &= \frac{2}{\Delta x\omega_j}\sum_{m=-1}^{m=d}\int_{x_{i,m}}^{x_{i,m+1}}\varphi_{i,j}(x)\mathrm{d}x a'_{x,i,m+1/2,k,\ell}\\ \mathcal{B}_{i,j,k,\ell} &= \frac{2}{\Delta y\omega_\ell}\sum_{m=-1}^{m=d}\int_{y_{k,m}}^{y_{k,m+1}}\psi_{k,\ell}(y)\mathrm{d}y a'_{y,i,j,k,m+1/2}\\ a'_{x,i,m+1/2,k,\ell} &= (a_{x,i,m+1,k,\ell}-a_{x,i,m,k,\ell})/(x_{i,m+1}-x_{i,m})\\ a'_{y,i,j,k,m+1/2} &= (a_{y,i,j,k,m+1}-a_{y,i,j,k,m})/(y_{k,m+1}-y_{k,m}).\end{aligned}$$

Sketch of the proof. - First, perform the x-advection: use the representation of f on the DG basis, use an explicit Euler for the characteristics $x^b_{i,m} = x_{i,m} - \Delta t a_x(x_{i,m}, y)$ into the explicit expression for the Jacobian. Then, perform similar calculations for the y-advection by neglecting the terms of second order in time.
If we had

$$\sum_{j,\ell} \omega_j \omega_\ell (\mathcal{A}_{i,j,k,\ell} + \mathcal{B}_{i,j,k,\ell}) = 0 \quad \forall (i,k) \tag{B.2}$$

then we would have the property, through (B.1),

$$f^n_{i,j,k,\ell} = 1 \ \forall (i,j,k,\ell) \quad \Longrightarrow \quad \sum_{j,\ell} \omega_j \omega_\ell f^{n+1}_{i,j,k,\ell} = 1 + \mathcal{O}(\Delta t^2) \ \forall (i,k).$$

Therefore, we modify our scheme in order to fulfill (B.2): suppose we have computed the electric potential Φ on the DG basis

$$\Phi(x,y) = \sum_{i,j,k,\ell} \Phi_{i,j,k,\ell} \varphi_{i,j}(x) \psi_{k,\ell}(y),$$

then if we set, $\forall (k,r)$,

$$\begin{aligned}
\frac{\partial E_y}{\partial x}(x, y_{k,r}) &= \frac{\sum_\ell (\Phi_{i,m+1,k,\ell} - \Phi_{i,m,k,\ell}) \psi'_{k,\ell}(y_{k,r})}{x_{i,m+1} - x_{i,m}}, \ x \in [x_{i,m}, x_{i,m+1}] \\
\frac{\partial E_y}{\partial x}(x, y_{k,r}) &= \frac{\left(\sum_{j,\ell} \Phi_{i,j,k,\ell} \varphi_{i,j}(x_{i+1/2}) - \sum_\ell \Phi_{i,d,k,\ell} \right) \psi'_{k,\ell}(y_{k,r})}{x_{i+1/2} - x_{i,d}}, \ x \in [x_{i,d}, x_{i+1/2}] \\
\frac{\partial E_y}{\partial x}(x, y_{k,r}) &= \frac{\left(\sum_\ell \Phi_{i,0,k,\ell} - \sum_{j,\ell} \Phi_{i,j,k,\ell} \varphi_{i,j}(x_{i-1/2}) \right) \psi'_{k,\ell}(y_{k,r})}{x_{i,0} - x_{i-1/2}}, \ x \in [x_{i-1/2}, x_{i,0}]
\end{aligned}$$

and $\forall (i,s)$

$$\begin{aligned}
\frac{\partial E_x}{\partial y}(x_{i,s}, y) &= \frac{\sum_j (\Phi_{i,j,k,m+1} - \Phi_{i,j,k,m}) \varphi'_{i,j}(x_{i,s})}{y_{k,m+1} - y_{k,m}}, \ y \in [y_{k,m}, y_{k,m+1}] \\
\frac{\partial E_x}{\partial y}(x_{i,s}, y) &= \frac{\left(\sum_{j,\ell} \Phi_{i,j,k,\ell} \psi_{k,\ell}(y_{k+1/2}) - \sum_j \Phi_{i,j,k,d} \right) \varphi'_{i,j}(x_{i,s})}{y_{k+1/2} - y_{k,d}}, \ y \in [y_{k,d}, y_{k+1/2}] \\
\frac{\partial E_x}{\partial y}(x_{i,s}, y) &= \frac{\left(\sum_j \Phi_{i,j,k,0} - \sum_{j,\ell} \Phi_{i,j,k,\ell} \psi_{k,\ell}(y_{k-1/2}) \right) \varphi'_{i,j}(x_{i,s})}{y_{k,0} - y_{k-1/2}}, \ y \in [y_{k-1/2}, y_{k,0}]
\end{aligned}$$

it can be shown that we obtain the desired property (B.2).

[1] M. Bostan, The Vlasov-Maxwell system with strong initial magnetic field. Guiding-center approximation, SIAM J. Multiscale Model. Simul. 6 (3) 1026–1058.

[2] G. Strang, On the construction and comparison of difference schemes, SIAM J. Numer. Anal. (5) (1968) 506–517.

[3] C. Cheng, G. Knorr, The integration of the Vlasov equation in configuration space, J. Comput. Phys. (22) (1976) 330–351.

[4] N. Crouseilles, M. Mehrenberger, E. Sonnendrücker, Conservative semi-lagrangian schemes for Vlasov equations, J. Comput. Phys. (229) (2010) 1927–1953.

[5] B. A. de Dios, J. A. Carrillo, C. W. Shu, Discontinuous Galerkin methods for the multi-dimensional Vlasov-Poisson problem, Math. Mod. Meth. Appl. Sc. (to appear).

[6] B. A. de Dios, J. A. Carrillo, C. W. Shu, Discontinuous Galerkin methods for the One-Dimensional Vlasov-Poisson problem, Kinetic and Related Models 4 (2011) 955–989.

[7] A. Ern, J.-L. Guermond, Discontinuous Galerkin methods for Friedrichs' systems. I. General theory, SIAM J. Numer. Anal. 44 (2) (2006) 753–778. doi:10.1137/050624133.
URL http://dx.doi.org/10.1137/050624133

[8] B. Ayuso, L. D. Marini, Discontinuous Galerkin methods for advection-diffusion-reaction problems, SIAM J. Numer. Anal. 47 (2) (2009) 1391–1420. doi:10.1137/080719583.
URL http://dx.doi.org/10.1137/080719583

[9] P. Houston, C. Schwab, E. Süli, Discontinuous *hp*-finite element methods for advection-diffusion-reaction problems, SIAM J. Numer. Anal. 39 (6) (2002) 2133–2163 (electronic). doi:10.1137/S0036142900374111.
URL http://dx.doi.org/10.1137/S0036142900374111

[10] P. Castillo, B. Cockburn, D. Schötzau, C. Schwab, Optimal a priori error estimates for the *hp*-version of the local discontinuous Galerkin method

for convection-diffusion problems, Math. Comp. 71 (238) (2002) 455–478. doi:10.1090/S0025-5718-01-01317-5.
URL http://dx.doi.org/10.1090/S0025-5718-01-01317-5

[11] B. Cockburn, Discontinuous Galerkin methods for convection-dominated problems, in: High-order methods for computational physics, Vol. 9 of Lect. Notes Comput. Sci. Eng., Springer, Berlin, 1999, pp. 69–224.

[12] B. Cockburn, C. Dawson, Some extensions of the local discontinuous Galerkin method for convection-diffusion equations in multidimensions, in: The mathematics of finite elements and applications, X, MAFELAP 1999 (Uxbridge), Elsevier, Oxford, 2000, pp. 225–238. doi:10.1016/B978-008043568-8/50014-6.
URL http://dx.doi.org/10.1016/B978-008043568-8/50014-6

[13] A. Buffa, T. J. R. Hughes, G. Sangalli, Analysis of a multiscale discontinuous Galerkin method for convection-diffusion problems, SIAM J. Numer. Anal. 44 (4) (2006) 1420–1440 (electronic). doi:10.1137/050640382.
URL http://dx.doi.org/10.1137/050640382

[14] H. Zarin, H.-G. Roos, Interior penalty discontinuous approximations of convection-diffusion problems with parabolic layers, Numer. Math. 100 (4) (2005) 735–759. doi:10.1007/s00211-005-0598-1.
URL http://dx.doi.org/10.1007/s00211-005-0598-1

[15] D. N. Arnold, F. Brezzi, B. Cockburn, L. D. Marini, Unified analysis of discontinuous Galerkin methods for elliptic problems, SIAM J. Numer. Anal. 39 (5) (2001/02) 1749–1779. doi:10.1137/S0036142901384162.
URL http://dx.doi.org/10.1137/S0036142901384162

[16] P. Castillo, B. Cockburn, I. Perugia, D. Schötzau, An a priori error analysis of the local discontinuous Galerkin method for elliptic problems, SIAM J. Numer. Anal. 38 (5) (2000) 1676–1706 (electronic). doi:10.1137/S0036142900371003.
URL http://dx.doi.org/10.1137/S0036142900371003

[17] H. Chen, Pointwise error estimates of the local discontinuous Galerkin method for a second order elliptic problem, Math. Comp. 74 (251) (2005) 1097–1116 (electronic). doi:10.1090/S0025-5718-04-01700-4.
URL http://dx.doi.org/10.1090/S0025-5718-04-01700-4

[18] Z. Chen, H. Chen, Pointwise error estimates of discontinuous Galerkin methods with penalty for second-order elliptic problems, SIAM J. Numer. Anal. 42 (3) (2004) 1146–1166. doi:10.1137/S0036142903421527.
URL http://dx.doi.org/10.1137/S0036142903421527

[19] B. Cockburn, G. Kanschat, I. Perugia, D. Schötzau, Superconvergence of the local discontinuous Galerkin method for elliptic problems on Cartesian grids, SIAM J. Numer. Anal. 39 (1) (2001) 264–285 (electronic). doi:10.1137/S0036142900371544.
URL http://dx.doi.org/10.1137/S0036142900371544

[20] J. Xu, P. N. Ostroumov, B. Mustapha, J. Nolen, Scalable direct Vlasov solver with discontinuous Galerkin method on unstructured mesh, SIAM J. Sci. Comput. 32 (6) (2010) 3476–3494. doi:10.1137/10078904X.
URL http://dx.doi.org/10.1137/10078904X

[21] A. Mangeney, F. Califano, C. Cavazzoni, P. Travnicek, A numerical scheme for the integration of the Vlasov-Maxwell system of equations, J. Comput. Phys. 179 (2) (2002) 495–538. doi:10.1006/jcph.2002.7071.
URL http://dx.doi.org/10.1006/jcph.2002.7071

[22] M. Dumbser, C. Munz, Arbitrary high order discontinuous Galerkin schemes, Numerical Methods for Hyperbolic and Kinetic Problems (2005) 295–333.

[23] M. Dumbser, C. Munz, ADER discontinuous Galerkin schemes for aeroacoustics, Comptes Rendus Mćanique 333 (9) (2005) 683–687.

[24] M. Restelli, L. Bonaventura, R. Sacco, A semi-lagrangian discontinuous galerkin method for scalar advection by incompressible flows, Journal of Computational Physics 216 (2006) 195–215. doi:10.1016/j.jcp.2005.11.030.

[25] J. M. Qiu, C. W. Shu, Positivity preserving semi-lagrangian discontinuous Galerkin formulation: theoretical analysis and application to the Vlasov-Poisson system, J. Comput. Phys. 230 (23) (2011) 8386–8409.

[26] J. A. Rossmanith, D. C. Seal, A positivity-preserving high-order semi-Lagrangian discontinuous Galerkin scheme for the Vlasov-Poisson equations, J. Comput. Phys. 230 (2011) 6203–6232.

[27] N. Crouseilles, M. Mehrenberger, F. Vecil, Discontinuous Galerkin semi-lagrangian method for Vlasov-Poisson, ESAIM Proc. 32, CEMRACS'10 research achievements: Numerical modeling of fusion.

[28] F. Bouchut, F. Golse, M. Pulvirenti, Kinetic equations and asymptotic theory, Series in Appl. Math., Gauthiers-Villars 4.

[29] R. J. Leveque, High-resolution conservative algorithms for advection in incompressible flow, SIAM J. Numer. Anal. 33 (2) (1996) 627–665. doi:10.1137/0733033.
URL http://dx.doi.org/10.1137/0733033

[30] L. M. Harris, P. H. Lauritzen, R. Mittal, A Flux-form version of the Conservative Semi-Lagrangian Multi-tracer transport scheme (CSLAM) on the cubed sphere grid, J. Comput. Phys. 230 (4) 1215–1237. doi:10.1016/j.jcp.2010.11.001.

[31] E. Sonnendrücker, J. Roche, P. Bertrand, A. Ghizzo, The semi-lagrangian method for the numerical resolution of the Vlasov equation, J. Comput. Phys. (149) (1999) 201–220.