Long time behaviour of an exponential integrator for a Vlasov-Poisson system with strong magnetic field

Emmanuel Frenod, Sever Adrian Hirstoaga, Mathieu Lutz, Eric Sonnendrücker

Long time behaviour of an exponential integrator for a Vlasov-Poisson system with strong magnetic field

Emmanuel Frénod* Sever A. Hirstoaga† Mathieu Lutz ‡
Eric Sonnendrücker§

Abstract

With the aim of solving in a four dimensional phase space a multi-scale Vlasov-Poisson system, we propose in a Particle-In-Cell framework a robust time-stepping method that works uniformly when the small parameter vanishes. As an exponential integrator, the scheme is able to use large time steps with respect to the typical size of the solution's fast oscillations. In addition, we show numerically that the method has accurate long time behaviour and that it is asymptotic preserving with respect to the limiting Guiding Center system.

1 Introduction

In this paper we introduce a numerical scheme in order to simulate efficiently in time when the parameter ε vanishes the following four dimensional Vlasov equation

$$\partial_t f^\varepsilon + \mathbf{v} \cdot \nabla_{\mathbf{x}} f^\varepsilon + \left(\mathbf{\Xi}^\varepsilon + \frac{1}{\varepsilon} \mathbf{v}^\perp \right) \cdot \nabla_{\mathbf{v}} f^\varepsilon = 0, \tag{1.1}$$

$$f^\varepsilon \left(\mathbf{x}, \mathbf{v}, t = 0 \right) = f_0 \left(\mathbf{x}, \mathbf{v} \right), \tag{1.2}$$

where $\mathbf{x} = (x_1, x_2)$ stands for the position variable, $\mathbf{v} = (v_1, v_2)$ for the velocity variable, $\mathbf{v}^\perp$ for $(v_2, -v_1)$, $f^\varepsilon \equiv f^\varepsilon(\mathbf{x}, \mathbf{v}, t)$ is the distribution function, f_0 is given, and $\mathbf{\Xi}^\varepsilon \equiv \mathbf{\Xi}^\varepsilon(\mathbf{x}, t)$ corresponds to the electric field. Weak-$*$ and two-scale limits when ε goes to zero of this equation can be rigorously obtained following the methods introduced in [2] and [9]. We notice that equations (1.1)-(1.2) can be obtained from the six dimensional drift-kinetic regime by taking a constant magnetic field in the x_3-direction and an electric field evolving in the orthogonal plane to the magnetic field.

*Université Bretagne-Sud, UMR 6205, LMBA, F-56000 Vannes, France & Inria Nancy-Grand Est, CALVI Project & IRMA (UMR CNRS 7501), Université de Strasbourg, France

†Inria Nancy-Grand Est, CALVI Project & IRMA (UMR CNRS 7501), Université de Strasbourg, France

‡IRMA (UMR CNRS 7501) Université de Strasbourg, 7 rue René Descartes, F-67084 Strasbourg & Inria Nancy-Grand Est, CALVI Project, France

§Max Planck Institute for Plasma Physics, EURATOM Association, Boltzmannstr. 2, 85748 Garching, Germany

The main application will be the case when the electric field $\mathbf{\Xi}^\varepsilon$ is obtained by solving the Poisson equation. In this case we will rather denote by $\mathbf{E}^\varepsilon$ the electric field and thus, we will have to solve the following nonlinear system of equations:

$$\partial_t f^\varepsilon + \mathbf{v}\cdot\nabla_{\mathbf{x}} f^\varepsilon + \left(\mathbf{E}^\varepsilon + \frac{1}{\varepsilon}\mathbf{v}^\perp\right)\cdot\nabla_{\mathbf{v}} f^\varepsilon = 0, \tag{1.3}$$

$$\mathbf{E}^\varepsilon(\mathbf{x},t) = -\nabla_{\mathbf{x}}\phi^\varepsilon, \quad -\Delta_{\mathbf{x}}\phi^\varepsilon = \int_{\mathbb{R}^2} f^\varepsilon d\mathbf{v} - n_i, \tag{1.4}$$

$$f^\varepsilon(\mathbf{x},\mathbf{v},t=0) = f_0(\mathbf{x},\mathbf{v}), \tag{1.5}$$

where ϕ^ε is the electric potential and n_i is the background ion density. The system (1.3)-(1.4) is a first step towards a six dimensional model which can be used for the study of plasma under the influence of a strong magnetic field. The unknown $f^\varepsilon(\mathbf{x},\mathbf{v},t)$ represents the distribution of electrons in phase space at time t and thus, the system (1.3)-(1.4) describes the particle dynamics under the additional effect of the self-consistent electric field. The difficulty is that the large magnetic field, expressed by the $\mathbf{v}^\perp/\varepsilon$ term, introduces a new time scale, the rotation of particles around the magnetic field line, which is very small with respect to that of the electric field evolution. We are thus faced with a multi-scale problem whose numerical solution by standard methods requires heavy computational efforts.

We will also test our scheme when an external electric field in (1.1) is given by

$$\mathbf{\Xi}^\varepsilon(\mathbf{x},t) = \begin{pmatrix} 2x_1 + x_2 \\ x_1 + 2x_2 \end{pmatrix}, \tag{1.6}$$

which is the gradient of the potential $\varphi(x_1,x_2) = x_1^2 + x_1x_2 + x_2^2$. The reason for this particular case is twofold. First, we are able to write down the analytic solution to system (1.1)-(1.2) which thus leads to the capabilities of a real error computation and of testing our algorithm's main (second step) approximation alone. Second, we can write analytically the slow manifold, an important issue when testing the scheme for any initial condition (see Section 2 in [3]). Indeed, it is interesting to see how the errors of the numerical scheme change when a different initial condition is used. Thus, if for some particles the scheme performs much better than for others, the corresponding errors might lead, when applying the method in the Vlasov-Poisson case, to different errors in the electric field computation and thus, to an amplification of the disparate errors.

In this work we perform the numerical solution of the Vlasov equation (1.1) by particle methods (see [1]), which consist in approximating the distribution function by a finite number of macroparticles. The trajectories of these particles are computed from the characteristic curves

$$\frac{\mathrm{d}\mathbf{X}^\varepsilon}{\mathrm{d}t} = \mathbf{V}^\varepsilon, \qquad \mathbf{X}^\varepsilon(0) = \mathbf{x}_0, \tag{1.7}$$

$$\frac{\mathrm{d}\mathbf{V}^\varepsilon}{\mathrm{d}t} = \frac{1}{\varepsilon}(\mathbf{V}^\varepsilon)^\perp + \mathbf{\Xi}^\varepsilon(\mathbf{X}^\varepsilon,t), \qquad \mathbf{V}^\varepsilon(0) = \mathbf{v}_0, \tag{1.8}$$

of the Vlasov equation, whereas the electric field is computed, when coupling with Poisson equation as in (1.3)-(1.4), on a mesh in the physical space. The contribution of this paper is to propose a numerical scheme in time for solving these characteristic curves when the parameter ε vanishes.

Before describing our strategy, we need to place it towards some existing approaches classically known to solve multi-scale problems. When the electric field $\mathbf{\Xi}^\varepsilon$ is zero, the physical trajectory associated with (1.7)-(1.8) is a circle of center $\mathbf{c}_0 = \mathbf{x}_0 + \varepsilon \mathbf{v}_0^\perp$ and of radius $\varepsilon |\mathbf{v}_0|$, and the time period of the trajectory is $2\pi\varepsilon$. Otherwise, the dynamical system (1.7)-(1.8) can be viewed as a perturbation of the system obtained when the electric field is zero. Hence, in the general case of an electric field depending on position and time, the evolution of a given particle's position is a combination of two disparate in time motions (a stiff problem): a slow evolution of what was the center of the circle in the case where $\mathbf{\Xi}^\varepsilon$ is zero, usually called the Guiding Center, and a fast rotation of period about $2\pi\varepsilon$ with a small radius around it (see Fig. 1). We refer to Lee [14] and Dubin *et al.* [5] for comprehensive physical viewpoint reviews about such questions. Consequently, if one wants to do accurate simulation of the problem (1.3)-(1.5) using classical numerical schemes, one needs small time steps, in particular smaller than $2\pi\varepsilon$. Another way is to use not stiff models instead of (1.3)-(1.5), which can be simulated using larger time steps. Nevertheless, in this case, such reduced models (as the Guiding Center model, see [2], [10]) need to incorporate information about the self-consistent electric field acting on particles position and the additional effect generated by particles oscillations. One usual way to do this is to use techniques based on Asymptotic Analysis and Homogenization Methods leading to a limit equation in which the mutual influence of the particles can be expressed in terms of their apparent motion, and afterwards to simulate this limit equation. We refer to Frénod, Sonnendrücker [9, 10], Frénod, Raviart, Sonnendrücker [7], and Golse, Saint-Raymond [11] for a theoretical point of view on these questions, and Frénod, Salvarani, Sonnendrücker [8] for numerical applications of such techniques. In this paper, we propose an alternative to such methods allowing to make direct simulations of systems (1.1)-(1.2) and (1.3)-(1.5) with large time steps with respect to $2\pi\varepsilon$. In addition, our scheme inherently incorporates information about the real small oscillations in the solution and thus, one can recover this information at a macroscopic time. This can not be reproduced by a reduced model or can be partially done by homogenization. Concluding, the algorithm we propose has a computational cost in time rather close to that of a reduced model but the accuracy close to that of a high-order standard scheme for computing a reference solution.

Now, we start to summarize the basis of the method and the results of this paper. The stiffness of equations (1.7)-(1.8) comes from the velocity equation and therefore we are interested in solving in $\mathbb{R}^2$ for several small values of ε the following type of ODE

$$u'(t) = \frac{1}{\varepsilon} M u(t) + F(t, u(t)), \quad u(0) = u_0, \tag{1.9}$$

where M is a matrix giving a $\pi/2$-rotation in $\mathbb{R}^2$ and where F represents a nonlinear term playing the role of the electric field. As already mentioned, standard numerical schemes require very small time step to capture the stiff behaviour. Following [6], in this paper, we propose a method which is based on an exponential integrator in velocity. An exponential integrator (see [12]) consists in solving exactly the linear (stiff) part by using the variation-of-constants formula

$$u(t) = e^{\frac{t}{\varepsilon} M} u_0 + \int_0^t e^{\frac{t-\tau}{\varepsilon} M} F(\tau, u(\tau))\, d\tau. \tag{1.10}$$

Once the stiff part is exactly solved, we proceed with the numerical treatement of the integral

term in (1.10) as explained in [6]: we solve the ODEs over one fast period using an explicit high-order solver and then, thanks to (1.10), we compute an approximation of the solution over a large whole number of periods. Then, we introduce the following Guiding Center decomposition

$$\text{let } \mathbf{C}^\varepsilon \text{ be such that } \mathbf{X}^\varepsilon(t) = \mathbf{C}^\varepsilon(t) - \varepsilon\left(\mathbf{V}^\varepsilon(t)\right)^\perp \tag{1.11}$$

and we show the main algorithm's approximation to be equivalent to a linear approximation of $\mathbf{C}^\varepsilon$'s trajectory, an interesting issue when studying the particles' long time behaviour.

Afterwards, we start applications with the linear case of $\mathbf{\Xi}^\varepsilon$ given by (1.6). We thus calculate the analytic solution to (1.1)-(1.2)-(1.6), we check whether the scheme gives accurate solutions on, close to, and far from the slow manifold (as in [3]), and eventually, we obtain the same order of error for these three numerical solutions, in both short time and long time simulations. Recalling that the fast oscillation is of order ε, let us remark that from now on, by short time we mean of order 1 and by long time, of order $1/\varepsilon$.

Finally, our numerical results underline that the scheme is robust when using various large time steps compared to the fast oscillations and that it works uniformly when the parameter ε goes to zero. In addition, in the Vlasov-Poisson case, within long time simulations, we show that the method is asymptotic preserving, meaning that it is accurate in time in the limit $\varepsilon \to 0$, capturing the Guiding Center model in this limit (there is a huge literature about asymptotic preserving schemes, we cite only the classical paper [13]).

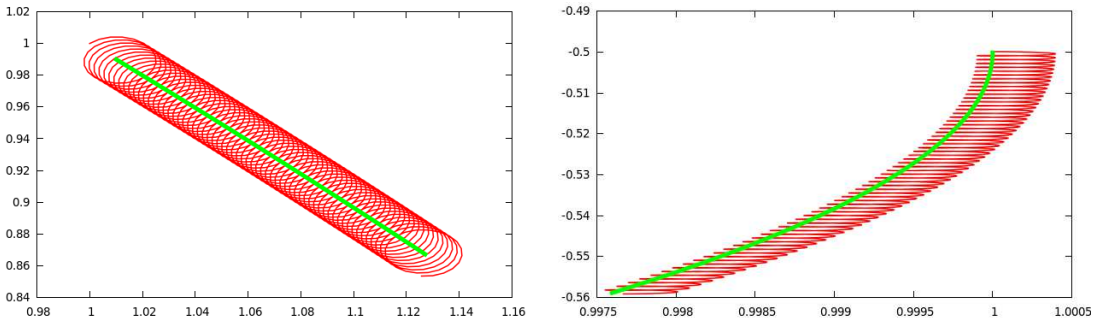

Figure 1: Illustration of formula (1.11) in the case when $\varepsilon = 0.01$ and the electric field is given by (1.6). The initial positions and velocities are $\left(\mathbf{x}_0^3, \mathbf{v}_0^3\right)$ (left) and $\left(\mathbf{x}_0^2, \mathbf{v}_0^2\right)$ (right) introduced in (5.8). In green the evolution of the Guiding Center and in red the position's evolution. The final time is $t = 4$.

The paper is organized as follows. In Section 2 we briefly recall the main steps of the Particle-In-Cell (PIC) method for solving the Vlasov-Poisson system in which we are interested. Then, Section 3 is devoted to the construction of the exponential integrator, named the ETD-PIC algorithm, for advancing in time the particles' position and velocity. In Section 4 we write the algorithm in terms of the Guiding Center position. Eventually, in Section 5, we implement our method in the cases presented above and we validate it in both, short and long times, simulations.

2 A Particle-In-Cell method

The numerical scheme that we describe in the next section is proposed in the framework of a Particle-In-Cell method. A PIC method consists first in approximating the initial condition f_0 in (1.2) by the following Dirac mass sum

$$f_0^{N_p}(\mathbf{x},\mathbf{v}) = \sum_{k=1}^{N_p} \omega_k \delta(\mathbf{x}-\mathbf{x}_{k,0})\,\delta(\mathbf{v}-\mathbf{v}_{k,0}), \tag{2.1}$$

where $\{(\mathbf{x}_{k,0},\mathbf{v}_{k,0})\}_{k=1}^{N_p}$ is a beam of N_p macroparticles distributed in the four dimensional phase space according to the density function f_0. Afterwards, one approximates the solution of (1.1)-(1.2), by

$$f_{N_p}^{\varepsilon}(\mathbf{x},\mathbf{v},t) = \sum_{k=1}^{N_p} \omega_k \delta(\mathbf{x}-\mathbf{X}_k^{\varepsilon}(t))\,\delta(\mathbf{v}-\mathbf{V}_k^{\varepsilon}(t)), \tag{2.2}$$

where $(\mathbf{X}_k^{\varepsilon}(t), \mathbf{V}_k^{\varepsilon}(t))$ is the position in phase space of macroparticle k moving along a characteristic curve of equation

$$\frac{\mathrm{d}\mathbf{X}_k^{\varepsilon}}{\mathrm{d}t} = \mathbf{V}_k^{\varepsilon}, \tag{2.3}$$

$$\frac{\mathrm{d}\mathbf{V}_k^{\varepsilon}}{\mathrm{d}t} = \frac{1}{\varepsilon}(\mathbf{V}_k^{\varepsilon})^{\perp} + \mathbf{\Xi}^{\varepsilon}(\mathbf{X}_k^{\varepsilon},t), \tag{2.4}$$

$$\mathbf{X}_k^{\varepsilon}(0) = \mathbf{x}_{k,0}, \qquad \mathbf{V}_k^{\varepsilon}(0) = \mathbf{v}_{k,0}. \tag{2.5}$$

Therefore, the problem consists in finding the positions and velocities $\left(\mathbf{X}_{k,n+1}^{\varepsilon}, \mathbf{V}_{k,n+1}^{\varepsilon}\right)$ at time t_{n+1} from their values at time t_n, by solving (2.3)-(2.4) with the initial condition $\left(\mathbf{X}_{k,n}^{\varepsilon}, \mathbf{V}_{k,n}^{\varepsilon}\right)$.

When the problem (2.3)-(2.4) is coupled to the Poisson equation, the electric field term in (2.4) is numerically computed in a macroparticle position at time t as follows:

1. Construct a spatial grid (the Poisson grid).
2. Compute on this grid

$$\rho^S(\mathbf{x},t) = \sum_{k=1}^{N_p} \omega_k \mathbf{S}(\mathbf{x}-\mathbf{X}_k^{\varepsilon}(t)), \tag{2.6}$$

   where $\mathbf{S}$ is a first order two dimensional spline.
3. Solve the Poisson equation $-\Delta_{\mathbf{x}}\phi(\mathbf{x},t) = \rho^S(\mathbf{x},t) - n_i$ on this grid and deduce the grid electric field.
4. Interpolate the grid electric field with the same first order spline yielding the density ρ^S in order to obtain the electric field at the macroparticle position.

Eventually, an important question in the PIC method is the numerical integration of the dynamical system (2.3)-(2.4). Here is the contribution of this paper, to propose an accurate numerical scheme when using large time steps compared to the fast oscillation. We thus introduce in the next section a method based on exponential time differencing, following the ideas in [6].

3 The exponential integrator in velocity for the Particle-In-Cell method

We first detail the exponential time differencing (ETD) method for solving the stiff velocity equation (2.4). Then, we describe the exponential integrator that we have implemented for solving (2.3)-(2.4) in the framework of the PIC algorithm.

3.1 The exponential integrator in velocity

One way to solve efficiently stiff ODEs is to use an exponential time differencing approach (see [3, 6, 12] and the references therein). Such a method is recognized to be accurate while avoiding simulation with small time steps. In order to write down the scheme in our case we follow the steps in [3]. Let M be the matrix defined by

$$M = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}, \tag{3.1}$$

and let

$$e^{\tau M} = \begin{pmatrix} \cos(\tau) & \sin(\tau) \\ -\sin(\tau) & \cos(\tau) \end{pmatrix} \tag{3.2}$$

be the exponential of M. Multiplying (1.8) by $e^{-\frac{\tau}{\varepsilon}M}$, we obtain

$$\frac{\mathrm{d}}{\mathrm{d}\tau}\left(e^{-\frac{\tau}{\varepsilon}M}\mathbf{V}^{\varepsilon}\right) = e^{-\frac{\tau}{\varepsilon}M}\left(-\frac{1}{\varepsilon}M\mathbf{V}^{\varepsilon}\right) + e^{-\frac{\tau}{\varepsilon}M}\frac{\mathrm{d}\mathbf{V}^{\varepsilon}}{\mathrm{d}\tau} \tag{3.3}$$

$$= e^{-\frac{\tau}{\varepsilon}M}\mathbf{\Xi}^{\varepsilon}\left(\mathbf{X}^{\varepsilon},\tau\right). \tag{3.4}$$

Integrating this equality between s and t (where $s < t$) yields

$$\mathbf{V}^{\varepsilon}(t) = e^{\frac{t-s}{\varepsilon}M}\mathbf{V}^{\varepsilon}(s) + e^{\frac{t-s}{\varepsilon}M}\int_s^t e^{\frac{s-\tau}{\varepsilon}M}\mathbf{\Xi}^{\varepsilon}\left(\mathbf{X}^{\varepsilon}(\tau),\tau\right)d\tau. \tag{3.5}$$

Concerning the position equation, an integration between s and t of (1.7) yields

$$\mathbf{X}^{\varepsilon}(t) = \mathbf{X}^{\varepsilon}(s) + \int_s^t \mathbf{V}^{\varepsilon}(\tau)\,d\tau. \tag{3.6}$$

Equation (3.5) has the merit to solve exactly the stiff part in the velocity equation and thus, we are left with the numerical treatement of the integral term.

3.2 The ETD-PIC method with large time steps

In this section we establish the time-stepping scheme following Section 4.2 in [6]. We write (3.5)-(3.6) with $s = t_n$ and $t = t_{n+1} = t_n + \Delta t$ in order to specify how the solution is computed at time t_{n+1} from its known value at time t_n. We are thus faced with the numerical computation of two integrals from t_n to t_{n+1}.
Since we want to build a scheme with a time step Δt much larger than the fast oscillation, we first need to find the unique positive integer N and the unique real $r \in [0, 2\pi\varepsilon)$ such that

$$\Delta t = N \cdot (2\pi\varepsilon) + r. \tag{3.7}$$

The derivation of the scheme, Algorithm 3.5, is based on the following approximations.

Approximation 3.1. *We have*

$$\int_{t_n}^{t_n + N\cdot(2\pi\varepsilon)} e^{\frac{t_n - \tau}{\varepsilon} M} \boldsymbol{\Xi}^\varepsilon \left(\mathbf{X}^\varepsilon(\tau), \tau\right) d\tau \simeq N \cdot \boldsymbol{\mathcal{I}}_1^\varepsilon, \tag{3.8}$$

where $\boldsymbol{\mathcal{I}}_1^\varepsilon$ *is defined by*

$$\boldsymbol{\mathcal{I}}_1^\varepsilon = \int_{t_n}^{t_n + 2\pi\varepsilon} e^{\frac{t_n - \tau}{\varepsilon} M} \boldsymbol{\Xi}^\varepsilon \left(\mathbf{X}^\varepsilon(\tau), \tau\right) d\tau. \tag{3.9}$$

Approximation 3.2. *We have*

$$\int_{t_n}^{t_n + N\cdot(2\pi\varepsilon)} \mathbf{V}^\varepsilon(\tau)\, d\tau \simeq N \cdot \boldsymbol{\mathcal{J}}_1^\varepsilon, \tag{3.10}$$

where $\boldsymbol{\mathcal{J}}_1^\varepsilon$ *is defined by*

$$\boldsymbol{\mathcal{J}}_1^\varepsilon = \int_{t_n}^{t_n + 2\pi\varepsilon} \mathbf{V}^\varepsilon(\tau)\, d\tau. \tag{3.11}$$

Remark 3.3. *Approximations 3.1 and 3.2 are valid if we made the assumptions that the velocity and the electric field evaluated at the particle position are quasi-periodic in time (with a period close to* $2\pi\varepsilon$*) and that this period does not change significantly with time. We will see in the next section that the assumption of quasi-periodicity and small variations in the period of the particle electric field* only *is enough to validate Approximations 3.1 and 3.2.*

Lemma 3.4. *Under Approximations 3.1 and 3.2 we obtain*

$$\begin{pmatrix} \mathbf{X}^\varepsilon(t_n + N\cdot(2\pi\varepsilon)) \\ \mathbf{V}^\varepsilon(t_n + N\cdot(2\pi\varepsilon)) \end{pmatrix} \simeq \begin{pmatrix} \mathbf{X}_n^\varepsilon \\ \mathbf{V}_n^\varepsilon \end{pmatrix} + N \cdot \begin{pmatrix} \mathbf{X}^\varepsilon(t_n + 2\pi\varepsilon) - \mathbf{X}_n^\varepsilon \\ \mathbf{V}^\varepsilon(t_n + 2\pi\varepsilon) - \mathbf{V}_n^\varepsilon \end{pmatrix}. \tag{3.12}$$

Proof. Applying formulas (3.5) and (3.6) with $s = t_n$ and $t = t_n + 2\pi\varepsilon$ we obtain

$$\begin{pmatrix} \mathbf{X}^\varepsilon(t_n + 2\pi\varepsilon) \\ \mathbf{V}^\varepsilon(t_n + 2\pi\varepsilon) \end{pmatrix} = \begin{pmatrix} \mathbf{X}_n^\varepsilon \\ \mathbf{V}_n^\varepsilon \end{pmatrix} + \begin{pmatrix} \boldsymbol{\mathcal{J}}_1^\varepsilon \\ \boldsymbol{\mathcal{I}}_1^\varepsilon \end{pmatrix}. \tag{3.13}$$

Applying again formulas (3.5) and (3.6) with $s = t_n$ and $t = t_n + N \cdot (2\pi\varepsilon)$ yields

$$\begin{pmatrix} \mathbf{X}^\varepsilon (t_n + N \cdot (2\pi\varepsilon)) \\ \mathbf{V}^\varepsilon (t_n + N \cdot (2\pi\varepsilon)) \end{pmatrix} = \begin{pmatrix} \mathbf{X}_n^\varepsilon \\ \mathbf{V}_n^\varepsilon \end{pmatrix} + \int_{t_n}^{t_n + N\cdot(2\pi\varepsilon)} \begin{pmatrix} \mathbf{V}^\varepsilon (\tau) \\ e^{\frac{t_n - \tau}{\varepsilon} M} \boldsymbol{\Xi}^\varepsilon (\mathbf{X}^\varepsilon (\tau), \tau) \end{pmatrix} d\tau. \quad (3.14)$$

Injecting (3.10) and (3.8) in (3.14), we obtain

$$\begin{pmatrix} \mathbf{X}^\varepsilon (t_n + N \cdot (2\pi\varepsilon)) \\ \mathbf{V}^\varepsilon (t_n + N \cdot (2\pi\varepsilon)) \end{pmatrix} \simeq \begin{pmatrix} \mathbf{X}_n^\varepsilon \\ \mathbf{V}_n^\varepsilon \end{pmatrix} + N \cdot \begin{pmatrix} \boldsymbol{\mathcal{J}}_1^\varepsilon \\ \boldsymbol{\mathcal{I}}_1^\varepsilon \end{pmatrix}. \quad (3.15)$$

Injecting (3.13) in (3.15) we obtain (3.12). This ends the proof of Lemma 3.4. □

Using Lemma 3.4, we deduce the following algorithm to compute $\left(\mathbf{X}_{n+1}^\varepsilon, \mathbf{V}_{n+1}^\varepsilon\right)$ from $(\mathbf{X}_n^\varepsilon, \mathbf{V}_n^\varepsilon)$:

Algorithm 3.5. *Assume that* $(\mathbf{X}_n^\varepsilon, \mathbf{V}_n^\varepsilon)$ *the solution of* (1.7)-(1.8) *at time* t_n *is given.*

1. *Compute* $(\mathbf{X}^\varepsilon (t_n + 2\pi\varepsilon), \mathbf{V}^\varepsilon (t_n + 2\pi\varepsilon))$ *by using a fine Runge-Kutta solver with initial condition* $(\mathbf{X}_n^\varepsilon, \mathbf{V}_n^\varepsilon)$.

2. *Compute* $(\mathbf{X}^\varepsilon (t_n + N \cdot (2\pi\varepsilon)), \mathbf{V}^\varepsilon (t_n + N \cdot (2\pi\varepsilon)))$ *thanks to formula* (3.12), i.e., *by setting*

$$\begin{pmatrix} \mathbf{X}^\varepsilon (t_n + N \cdot (2\pi\varepsilon)) \\ \mathbf{V}^\varepsilon (t_n + N \cdot (2\pi\varepsilon)) \end{pmatrix} = \begin{pmatrix} \mathbf{X}_n^\varepsilon \\ \mathbf{V}_n^\varepsilon \end{pmatrix} + N \cdot \begin{pmatrix} \mathbf{X}^\varepsilon (t_n + 2\pi\varepsilon) - \mathbf{X}_n^\varepsilon \\ \mathbf{V}^\varepsilon (t_n + 2\pi\varepsilon) - \mathbf{V}_n^\varepsilon \end{pmatrix}. \quad (3.16)$$

3. *Compute* $(\mathbf{X}^\varepsilon, \mathbf{V}^\varepsilon)$ *at time* t_{n+1} *by using a fine Runge-Kutta solver with initial condition* $(\mathbf{X}^\varepsilon, \mathbf{V}^\varepsilon)$ *at time* $t_n + N \cdot (2\pi\varepsilon)$, *obtained at the previous step.*

4 Link with the Guiding Center Decomposition

We have seen in Introduction that the time evolution of a particle's position following (1.7)-(1.8) can be split into two parts: the slow motion of the Guiding Center $\mathbf{C}^\varepsilon$ (see formula (1.11)) and a fast oscillation about it. In this section, we mainly see that this decomposition can be used to show that Approximation 3.1 is sufficient to justify the second step of Algorithm 3.5.

With this attempt, we first recall the formula giving the Guiding Center position

$$\mathbf{C}^\varepsilon (t) = \mathbf{X}^\varepsilon (t) + \varepsilon (\mathbf{V}^\varepsilon (t))^\perp. \quad (4.1)$$

Then, it is an easy fact to see that the rule in (3.16) is equivalent to

$$\begin{pmatrix} \mathbf{C}^\varepsilon (t_n + N \cdot (2\pi\varepsilon)) \\ \mathbf{V}^\varepsilon (t_n + N \cdot (2\pi\varepsilon)) \end{pmatrix} = \begin{pmatrix} \mathbf{C}_n^\varepsilon \\ \mathbf{V}_n^\varepsilon \end{pmatrix} + N \cdot \begin{pmatrix} \mathbf{C}^\varepsilon (t_n + 2\pi\varepsilon) - \mathbf{C}_n^\varepsilon \\ \mathbf{V}^\varepsilon (t_n + 2\pi\varepsilon) - \mathbf{V}_n^\varepsilon \end{pmatrix}. \quad (4.2)$$

In the following, we see that the rule for the Guiding Center in (4.2) may be obtained directly from the evolution of $\mathbf{C}^\varepsilon$ under an approximation similar to that in (3.8). To this end, we derive in time equation (4.1) and making use of equations (1.7)-(1.8) leads to

$$\frac{\mathrm{d}\mathbf{C}^\varepsilon}{\mathrm{d}t} (t) = \varepsilon M \boldsymbol{\Xi}^\varepsilon (\mathbf{X}^\varepsilon (t), t), \quad (4.3)$$

where $M\boldsymbol{\Xi}^\varepsilon$ is $\left(\boldsymbol{\Xi}^\varepsilon\right)^\perp = (\boldsymbol{\Xi}_2^\varepsilon, -\boldsymbol{\Xi}_1^\varepsilon)$. Thus, we see that the Guiding Center experiences a slow motion in time. Then, we integrate this equation between s and t (where $s < t$)

$$\mathbf{C}^\varepsilon(t) = \mathbf{C}^\varepsilon(s) + \varepsilon M \int_s^t \boldsymbol{\Xi}^\varepsilon\left(\mathbf{X}^\varepsilon(\tau), \tau\right) d\tau, \tag{4.4}$$

and using this equality with $s = t_n$ and $t = t_n + N \cdot (2\pi\varepsilon)$ yields

$$\mathbf{C}^\varepsilon(t_n + N \cdot (2\pi\varepsilon)) = \mathbf{C}_n^\varepsilon + \varepsilon M \int_{t_n}^{t_n + N\cdot(2\pi\varepsilon)} \boldsymbol{\Xi}^\varepsilon\left(\mathbf{X}^\varepsilon(\tau), \tau\right) d\tau. \tag{4.5}$$

Therefore, as done in Section 3.2, under the assumption

$$\int_{t_n}^{t_n + N\cdot(2\pi\varepsilon)} \boldsymbol{\Xi}^\varepsilon\left(\mathbf{X}^\varepsilon(\tau), \tau\right) d\tau \simeq N \cdot \int_{t_n}^{t_n + 2\pi\varepsilon} \boldsymbol{\Xi}^\varepsilon\left(\mathbf{X}^\varepsilon(\tau), \tau\right) d\tau \tag{4.6}$$

we deduce from (4.5) that

$$\mathbf{C}^\varepsilon(t_n + N \cdot (2\pi\varepsilon)) \simeq \mathbf{C}_n^\varepsilon + N \cdot \left(\mathbf{C}^\varepsilon(t_n + 2\pi\varepsilon) - \mathbf{C}_n^\varepsilon\right). \tag{4.7}$$

In conclusion, assuming that the time period of the electric field *only* does not change significantly in time leads the approximation (3.12) to be valid. Indeed, this assumption allows us to use the approximations in (4.6) and in (3.8). Then, following the lines of the proof of Lemma 3.4, we obtain that (4.7) is satisfied and thus, that approximation (3.12) is valid.

Finally, the Guiding Center gives information about the qualitative behaviour of the long time position's evolution. Indeed, being almost free of fast oscillations, the evolution of $\mathbf{C}^\varepsilon$ easily brings out the curvature of the macroscopic motion of the particle position (see Figs. 1 and 2). In the case of equations (1.1), (1.2), (1.6), this macroscopic evolution is periodic and the large period can be explicitly computed, being about $2\pi/\varepsilon$ (see Section 5.1.1).

5 Validation of the numerical method

We now validate our algorithm in the test cases presented in Introduction. First, in all three sections, our numerical experiments show that the scheme performs very well in robustness and accuracy when using very large time steps with respect to the small scale of oscillations. Then, in sections 5.1 and 5.2, within short and long time simulations, we show that the scheme works uniformly when the small parameter vanishes. More precisely, Section 5.1 is concerned with an analytic case that allows us to compute the real errors of the method and, in addition, to validate it when starting simulations with several types of initial conditions. In Section 5.2 we solve the Vlasov-Poisson system (1.3)-(1.5) and compare our method against a solution estimated with a very small time step, a reference solution. At the end, in Section 5.3, we do long time numerical experiments and show that the scheme is asymptotic preserving by comparisons with the limit Guiding Center model.

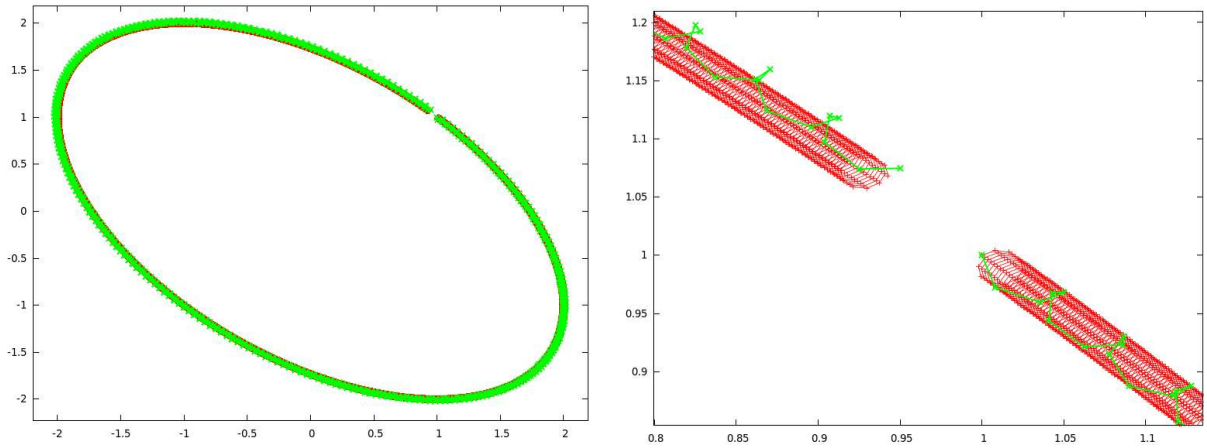

Figure 2: The linear case in Section 5.1 with $\varepsilon = 0.01$ and the initial condition $(1, 1, 1, 1)$: the position's evolution in time until $t = 360$; the entire trajectory (left) and a zoom at the begining of the dynamics (right); In green the result of the ETD scheme using a time step $\Delta t = 30\varepsilon$ and in red the analytic solution given in formulas (5.1)

5.1 A linear case

In this section we consider the Vlasov equation (1.1)-(1.2) provided with the external electric field $\mathbf{\Xi}^\varepsilon$ given by (1.6). In order to test our algorithm it will be interesting to localize the initial conditions for which the fast oscillations disappear. This domain is usually called the slow manifold (see [3] and the references therein). In Section 5.1.1 we compute an analytic expression of the solution, which allows us to compute the slow manifold. Then, in Section 5.1.2 we compare the outcome of the ETD-PIC method to the solution, starting with several initial conditions.

5.1.1 Analytic solution

Let ε be such that $0 < \varepsilon < \sqrt{1 - \frac{\sqrt{3}}{2}} \simeq 0.366$. Then, the solution of (1.7)-(1.8) is given by:

$$
\begin{aligned}
X_1^\varepsilon(t; \mathbf{x}_0, \mathbf{v}_0) =& K_1^\varepsilon \left(\cos(a_\varepsilon t) - \frac{a_\varepsilon}{\varepsilon} \sin(a_\varepsilon t) \right) + K_2^\varepsilon \left(\sin(a_\varepsilon t) + \frac{a_\varepsilon}{\varepsilon} \cos(a_\varepsilon t) \right) \\
&+ K_3^\varepsilon \left(\cos(b_\varepsilon t) - \frac{b_\varepsilon}{\varepsilon} \sin(b_\varepsilon t) \right) + K_4^\varepsilon \left(\sin(b_\varepsilon t) + \frac{b_\varepsilon}{\varepsilon} \cos(b_\varepsilon t) \right), \\
X_2^\varepsilon(t; \mathbf{x}_0, \mathbf{v}_0) =& - K_1^\varepsilon u_\varepsilon \cos(a_\varepsilon t) - K_2^\varepsilon u_\varepsilon \sin(a_\varepsilon t) - K_3^\varepsilon v_\varepsilon \cos(b_\varepsilon t) - K_4^\varepsilon v_\varepsilon \sin(b_\varepsilon t), \\
V_1^\varepsilon(t; \mathbf{x}_0, \mathbf{v}_0) =& - K_1^\varepsilon a_\varepsilon \left(\frac{a_\varepsilon}{\varepsilon} \cos(a_\varepsilon t) + \sin(a_\varepsilon t) \right) + K_2^\varepsilon a_\varepsilon \left(\cos(a_\varepsilon t) - \frac{a_\varepsilon}{\varepsilon} \sin(a_\varepsilon t) \right) \\
&- K_3^\varepsilon b_\varepsilon \left(\frac{b_\varepsilon}{\varepsilon} \cos(b_\varepsilon t) + \sin(b_\varepsilon t) \right) + K_4^\varepsilon b_\varepsilon \left(\cos(b_\varepsilon t) - \frac{b_\varepsilon}{\varepsilon} \sin(b_\varepsilon t) \right), \\
V_2^\varepsilon(t; \mathbf{x}_0, \mathbf{v}_0) =& K_1^\varepsilon a_\varepsilon u_\varepsilon \sin(a_\varepsilon t) - K_2^\varepsilon a_\varepsilon u_\varepsilon \cos(a_\varepsilon t) + K_3^\varepsilon b_\varepsilon v_\varepsilon \sin(b_\varepsilon t) - K_4^\varepsilon b_\varepsilon v_\varepsilon \cos(b_\varepsilon t),
\end{aligned}
\tag{5.1}
$$

where

$$a_\varepsilon = \sqrt{\frac{1-4\varepsilon^2-\sqrt{1-8\varepsilon^2+4\varepsilon^4}}{2\varepsilon^2}}, \tag{5.2}$$

$$b_\varepsilon = \sqrt{\frac{1-4\varepsilon^2+\sqrt{1-8\varepsilon^2+4\varepsilon^4}}{2\varepsilon^2}}, \tag{5.3}$$

$$\begin{aligned} u_\varepsilon &= 2+a_\varepsilon^2, \\ v_\varepsilon &= 2+b_\varepsilon^2, \\ w_\varepsilon &= 1+\frac{a_\varepsilon^2}{\varepsilon^2}, \\ x_\varepsilon &= a_\varepsilon^2-b_\varepsilon^2, \end{aligned} \tag{5.4}$$

and

$$\begin{pmatrix} K_1^\varepsilon \\ K_2^\varepsilon \\ K_3^\varepsilon \\ K_4^\varepsilon \end{pmatrix} = \begin{pmatrix} \frac{\varepsilon^2 v_\varepsilon}{(2-\varepsilon^2)x_\varepsilon} & -\frac{1}{u_\varepsilon}+\frac{\varepsilon^2 v_\varepsilon w_\varepsilon}{(2-\varepsilon^2)x_\varepsilon u_\varepsilon} & -\frac{\varepsilon v_\varepsilon}{(2-\varepsilon^2)x_\varepsilon} & 0 \\ \frac{\varepsilon}{a_\varepsilon}+\frac{\varepsilon^3}{(2-\varepsilon^2)a_\varepsilon}-\frac{2\varepsilon u_\varepsilon}{x_\varepsilon a_\varepsilon(2-\varepsilon^2)} & -\frac{\varepsilon v_\varepsilon}{u_\varepsilon a_\varepsilon x_\varepsilon}\left(1+\frac{\varepsilon^2 w_\varepsilon}{2-\varepsilon^2}\right) & \frac{\varepsilon^2 v_\varepsilon}{(2-\varepsilon^2)a_\varepsilon x_\varepsilon} & -\frac{1}{a_\varepsilon x_\varepsilon} \\ -\frac{\varepsilon^2 u_\varepsilon}{(2-\varepsilon^2)x_\varepsilon} & -\frac{\varepsilon^2 w_\varepsilon}{(2-\varepsilon^2)x_\varepsilon} & \frac{\varepsilon u_\varepsilon}{(2-\varepsilon^2)x_\varepsilon} & 0 \\ \frac{2\varepsilon u_\varepsilon}{(2-\varepsilon^2)b_\varepsilon x_\varepsilon} & \frac{\varepsilon}{b_\varepsilon x_\varepsilon}+\frac{\varepsilon^3 w_\varepsilon}{(2-\varepsilon^2)b_\varepsilon x_\varepsilon} & -\frac{\varepsilon^2 u_\varepsilon}{(2-\varepsilon^2)b_\varepsilon x_\varepsilon} & \frac{1}{b_\varepsilon x_\varepsilon} \end{pmatrix} \begin{pmatrix} x_{0,1} \\ x_{0,2} \\ v_{0,1} \\ v_{0,2} \end{pmatrix}. \tag{5.5}$$

We can thus observe that, in addition to the fast oscillations of period $\frac{2\pi}{b_\varepsilon} \sim 2\pi\varepsilon$, the solution of (1.7)-(1.8) contains slow oscillations of period $\frac{2\pi}{a_\varepsilon} \sim \frac{2\pi}{\sqrt{3}\varepsilon}$.

In this case, the slow manifold corresponds to the intersection between the hyperplanes $\{(\mathbf{x}_0, \mathbf{v}_0) \text{ such that } K_3^\varepsilon(\mathbf{x}_0, \mathbf{v}_0) = 0\}$ and $\{(\mathbf{x}_0, \mathbf{v}_0) \text{ such that } K_4^\varepsilon(\mathbf{x}_0, \mathbf{v}_0) = 0\}$. Since the two hyperplanes are different, the intersection is of dimension two. Straightforward computations yield that

$$\left\{ \begin{pmatrix} 1 \\ -\frac{u_\varepsilon}{w_\varepsilon} \\ 0 \\ -\frac{2\varepsilon u_\varepsilon}{2-\varepsilon^2}+\frac{\varepsilon u_\varepsilon}{w_\varepsilon}+\frac{\varepsilon^3 u_\varepsilon}{2-\varepsilon^2} \end{pmatrix}, \begin{pmatrix} 1 \\ 0 \\ \varepsilon \\ -\frac{2\varepsilon u_\varepsilon}{2-\varepsilon^2}+\frac{\varepsilon^3 u_\varepsilon}{2-\varepsilon^2} \end{pmatrix} \right\} \tag{5.6}$$

form a basis of this vector space. Subsequently, we denote by $\boldsymbol{\mathcal{D}}_2$ this space.

5.1.2 Short and long time numerical simulations

In this section we test the ETD-PIC method against the solution previously obtained. We consider two different types of initial condition f_0. The first one is with one macroparticle, alternatively located on, close to, and far from the slow manifold. In the second case, we consider a beam of macroparticles and we compute the maximum in time of the mean of the Euclidean errors.

Considering one particle alternatively on, close to, and far from the slow manifold means that we take initial conditions

$$f_0^i(\mathbf{x},\mathbf{v}) = \delta\left(\mathbf{x}-\mathbf{x}_0^i\right)\delta\left(\mathbf{v}-\mathbf{v}_0^i\right), \tag{5.7}$$

where $i=1,2,3$, and $\left(\mathbf{x}_0^1,\mathbf{v}_0^1\right)$ is on the slow manifold, $\left(\mathbf{x}_0^2,\mathbf{v}_0^2\right)$ is close to the slow manifold, and $\left(\mathbf{x}_0^3,\mathbf{v}_0^3\right)$ is far from the slow manifold. For the numerical simulations we take

$$\begin{aligned}
\left(\mathbf{x}_0^1,\mathbf{v}_0^1\right) &= \left(1,0,\varepsilon,-\frac{2\varepsilon u_\varepsilon}{2-\varepsilon^2}+\frac{\varepsilon^3 u_\varepsilon}{2-\varepsilon^2}\right),\\
\left(\mathbf{x}_0^2,\mathbf{v}_0^2\right) &= \left(1,-\frac{u_\varepsilon}{w_\varepsilon},\varepsilon\frac{w_\varepsilon}{u_\varepsilon},-\frac{2\varepsilon u_\varepsilon}{2-\varepsilon^2}+\varepsilon\frac{w_\varepsilon}{u_\varepsilon}+\frac{\varepsilon^3 u_\varepsilon}{2-\varepsilon^2}\right),\\
\left(\mathbf{x}_0^3,\mathbf{v}_0^3\right) &= (1,1,1,1).
\end{aligned} \tag{5.8}$$

Starting from the analytic formulas derived in the previous section, we have plotted in Fig. 1 the physical trajectories of the particles of which initial positions and velocities are $\left(\mathbf{x}_0^2,\mathbf{v}_0^2\right)$ and $\left(\mathbf{x}_0^3,\mathbf{v}_0^3\right)$, until final time 4. Using general formulas for the distances to the slow manifold $\boldsymbol{\mathcal{D}}_2$ from these particles, we obtain the specific values in Table 1.

	$\varepsilon=0.01$	$\varepsilon=0.005$	$\varepsilon=0.001$	$\varepsilon=0.0005$	$\varepsilon=0.0001$
$i=1$	0.00000000	0.00000000	0.00000000	0.00000000	0.00000000
$i=2$	0.01999800	0.00999975	0.00199999	0.00099984	0.00018878
$i=3$	1.41477865	1.41435495	1.41421923	1.41421509	1.41422155

Table 1: Euclidean distances between the slow manifold and the points $\left(\mathbf{x}_0^i,\mathbf{v}_0^i\right)_{i\in\{1,2,3\}}$ in (5.8), for several values of ε.

Denoting by $\left(\mathbf{X}^\varepsilon(t),\mathbf{V}^\varepsilon(t)\right)$ the result of the ETD-PIC method and by $\left(\mathbf{X}_{\text{sol}}^\varepsilon(t),\mathbf{V}_{\text{sol}}^\varepsilon(t)\right)$ the solution, we compute the global Euclidean errors at final time 10,

$$e_n = \max_{k\in\{0,\ldots,n\}} \left\|\left(\mathbf{X}^\varepsilon,\mathbf{V}^\varepsilon\right)(t_k) - \left(\mathbf{X}_{\text{sol}}^\varepsilon,\mathbf{V}_{\text{sol}}^\varepsilon\right)(t_k)\right\|_2, \tag{5.9}$$

where $n\in\mathbb{N}$ corresponds to the ratio between the final time of simulation and the time step Δt, for several values of ε and of Δt (see Figs. 3 and 4). In Fig. 4 we have plotted the error when starting with the close to the slow manifold particle introduced in (5.8). Similar error curves have been obtained for the particles on and far from the slow manifold in (5.8). Finally, in Fig. 6, we represent the global Euclidean errors at a large final time, for the three types of particles.

Second, preparating the test case in Section 5.2, we consider the following initial condition

$$f_0(\mathbf{x},\mathbf{v}) = \frac{1}{8\pi^2 v_{th}^2}\left(1+\eta\cos\left(k_{x_1}x_1+k_{x_2}x_2\right)\right)\boldsymbol{\chi}(\mathbf{x})\exp\left(-\frac{v_1^2+v_2^2}{2v_{th}^2}\right), \tag{5.10}$$

where $k_{x_1}=0,\ k_{x_2}=0.5,\ v_{th}=1,\ \eta=0.1$, and

$$\boldsymbol{\chi}(\mathbf{x}) = \chi_{[0,1]}(x_1)\,\chi_{[0,4\pi]}(x_2) \tag{5.11}$$

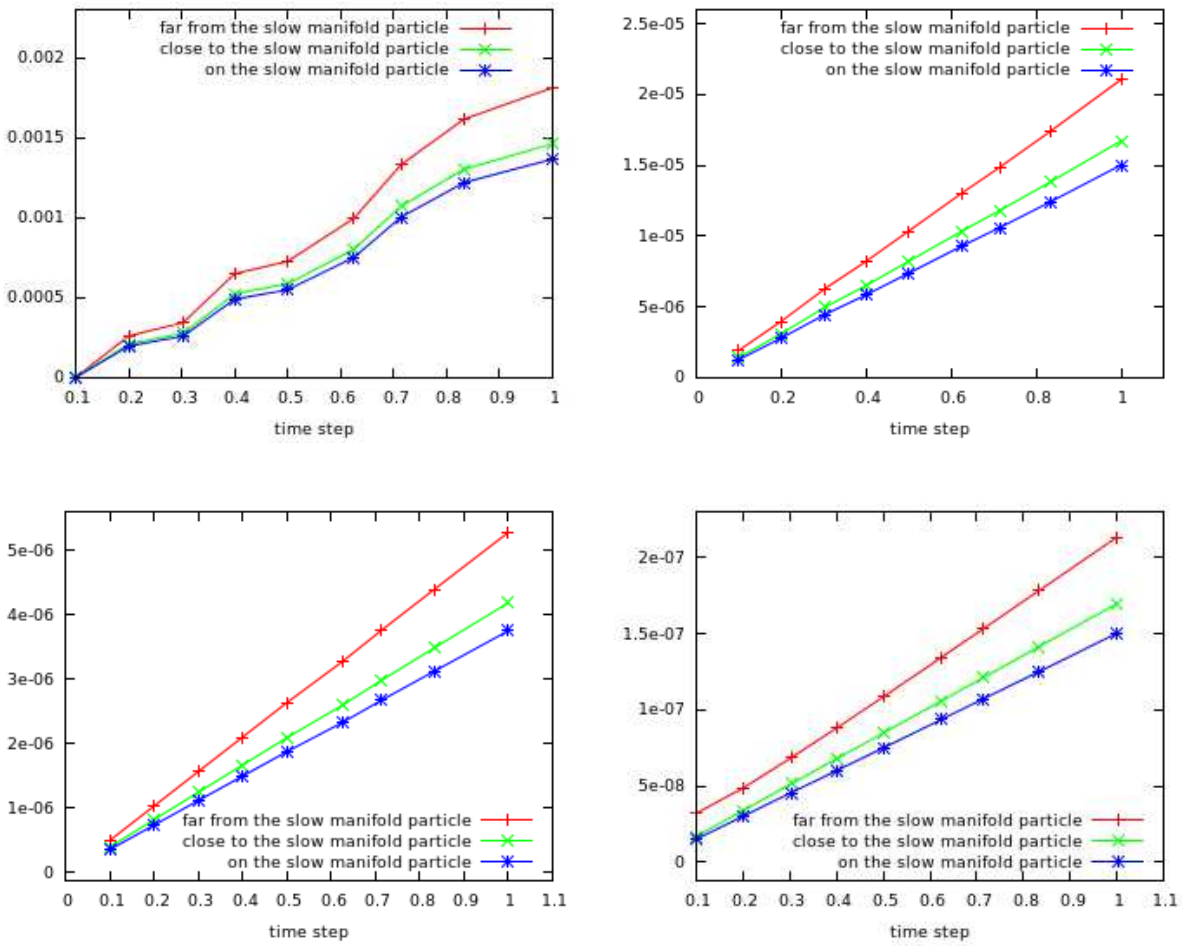

Figure 3: Global Euclidean errors of the ETD-PIC method at time 10 for four values of ε (on the top: $\varepsilon = 0.01$ (left) and $\varepsilon = 0.001$ (right); on the bottom: $\varepsilon = 0.0005$ (left) and $\varepsilon = 0.0001$ (right)), obtained with three initial conditions differently positioned with regard to the slow manifold $\boldsymbol{\mathcal{D}}_2$

where for any set A in $\mathbb{R}$, $\chi_A(x) = 1$ if $x \in A$ and 0 otherwise. We generate this distribution function using 10^4 particles in $\mathbb{R}^4$. Thus, in Fig. 5, we compute the maximum of the mean of the Euclidean errors

$$\mathcal{M}e_n = \max_{k\in\{0,\dots,n\}} \left(\frac{1}{N_p} \sum_{j=1}^{N_p} \left\| \left(\mathbf{X}_j^\varepsilon, \mathbf{V}_j^\varepsilon\right)(t_k) - \left(\mathbf{X}_{j,\mathrm{sol}}^\varepsilon, \mathbf{V}_{j,\mathrm{sol}}^\varepsilon\right)(t_k) \right\|_2 \right), \tag{5.12}$$

at final time 10, for several values of ε and of Δt.

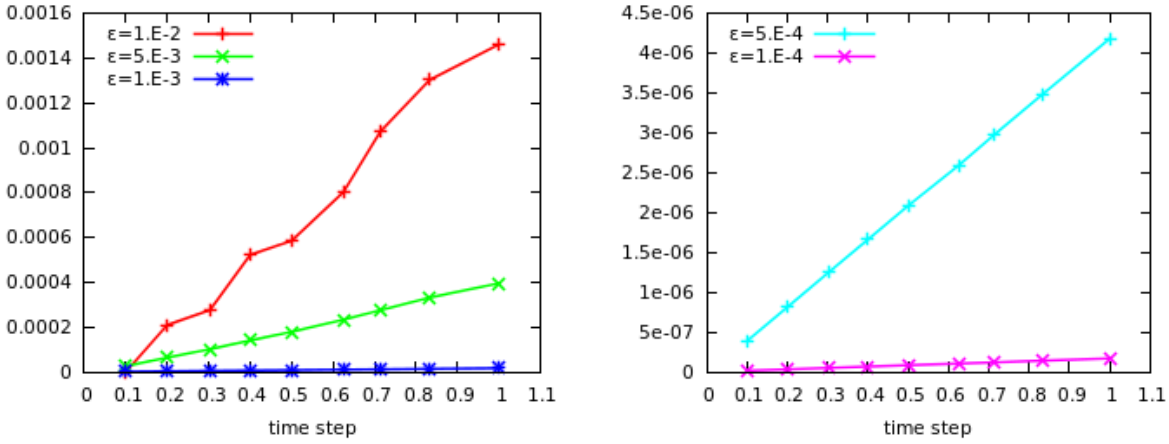

Figure 4: Global Euclidean errors of the ETD-PIC method at time 10 for several values of ε, obtained with an initial condition close to the slow manifold

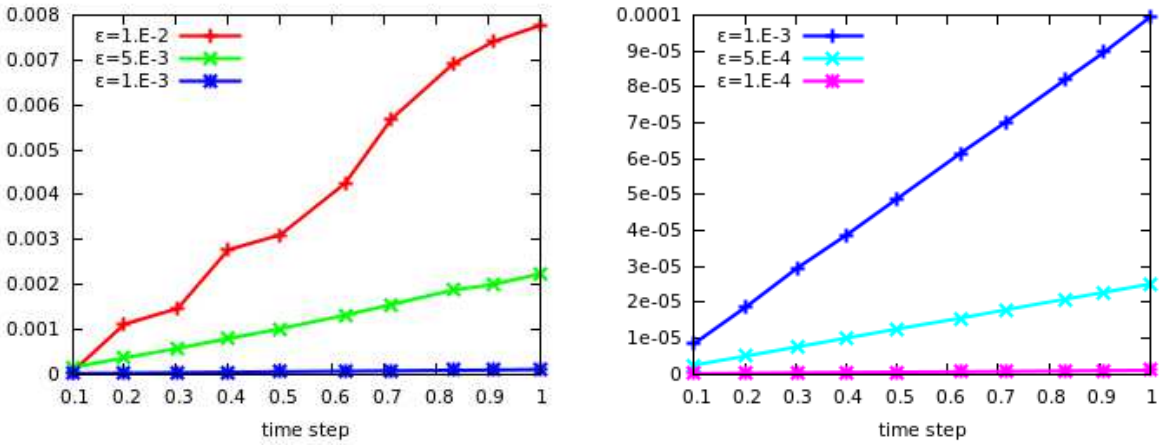

Figure 5: Maximum at time 10 of the mean of the Euclidean errors of the ETD-PIC method for several values of ε, obtained with the initial condition f_0 defined by (5.10)

5.1.3 Comments about the numerical results

In the simulations done in Section 5.1.2 we have implemented the Algorithm 3.5 as follows: we use $2\pi/b_\varepsilon$ instead of $2\pi\varepsilon$ in equation (3.7) and the same concerning the first two steps of the algorithm. Then, within the first and the third steps, no high-order scheme was used for solving the ODEs but the exact solution given by formulas (5.1). Thus, we can establish that the numerical error of our algorithm mainly consists of two parts: the error made in the first step, by replacing the real fast period of oscillation by $2\pi/b_\varepsilon$, denoted by E_P and the error made in the second step when following the macroscopic time evolution, denoted by E_M.

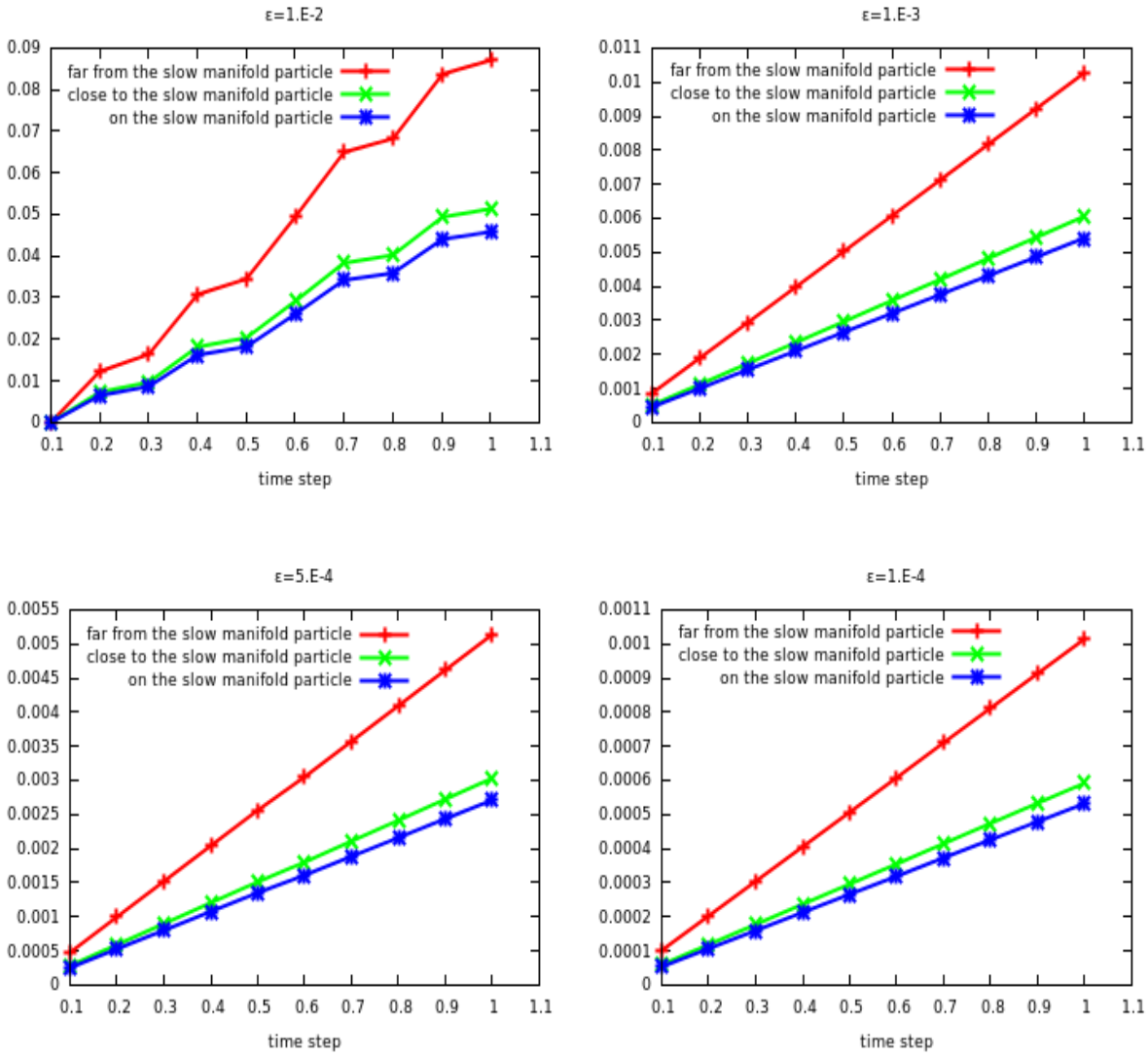

Figure 6: Global Euclidean errors of the ETD-PIC method at time $2\pi/a_\varepsilon \sim 1/\varepsilon$ for four values of ε, obtained with three initial conditions differently positioned with regard to the slow manifold $\boldsymbol{\mathcal{D}}_2$

In a first test, when keeping ε fixed, we calculate the errors when starting simulation with different particles (see Fig. 3 for short final time and Fig. 6 for long final time). When we take as initial condition f_0^1, the fast oscillations disappear and thus, E_P is zero. If we take as initial condition f_0^2 or f_0^3, we expect the error to be bigger for a particle off the slow manifold; this point of view is in accordance with our numerical results. Thus, we can observe that the smaller the distance to the slow manifold is, the smaller the corresponding error is. The reason is that, closer to the slow manifold a particle is, smaller the macroscopic position's displacement is and thus, smaller the error E_M is. Then, in long time simulations, the errors

are obviously significant, due to our simple linear approximation of the macroscopic time evolution. However, we note that for all values of ε, the three errors have the same order in magnitude (see the paragraph containing equation (1.6)). Finally, we point out that using very precise periods in the first step of the algorithm may be an important issue in order to reduce the error E_P that will propagate at macroscopic time when applying the second step (this idea was already stressed in [6]). Nevertheless, in this test case, we have first calculated the particles periods with the RK4 solver, as described in [6], and obtained that the difference between these values and $2\pi/b_\varepsilon$ are very small, of order ε^3. In addition, we have done simulations by using the computed periods instead of $2\pi/b_\varepsilon$ and the results are very similar to those obtained with the period $2\pi/b_\varepsilon$.

In a second test, for a fixed particle, we calculate errors (see Fig. 4) when using several values of ε. For the three types of particles considered above, we have obtained smaller errors with smaller ε, the reason being similar: the scheme performs better when the macroscopic change in position is smaller.

Eventually, the simulations (see Fig. 5) show that the scheme works uniformly when ε vanishes when using also the beam of particles defined in (5.10).

5.2 Short time Vlasov-Poisson test case

In the present section we check the accuracy of our numerical scheme for the nonlinear Vlasov-Poisson system at times of order 1, recalling that the period of the fast motion is of order ε. Thus, we consider equation (1.3)-(1.5) with the typical for the Landau damping case initial condition ([16]) given by

$$f_0(\mathbf{x},\mathbf{v}) = \frac{1}{2\pi v_{th}^2}\left(1+\eta\cos\left(k_1x_1+k_2x_2\right)\right)\exp\left(-\frac{v_1^2+v_2^2}{2v_{th}^2}\right), \tag{5.13}$$

where $v_{th}=1, \eta=0.1$, $k_1=0.5$, $k_2=0$, and

$$\mathbf{\Omega_x} = [0;T_1]\times[0;T_2], \tag{5.14}$$

with $T_1=2\pi/k_1$ and $T_2=1$. Next, in Section 5.2.1, we give the parameters used for the implementation of the reference solution and of our algorithm, the ETD-PIC method. The simulations concerning the reference solution and the first and third step of Algorithm 3.5 are done by following the four steps described in Section 2. Eventually, we comment the numerical results of the comparisons ETD-PIC method versus a reference solution.

5.2.1 Numerical issues

We implement the initial condition above with $N_p = 2\cdot 10^5$ macroparticles. The weights are

$$\omega_k = \frac{T_1T_2}{N_p}, \quad k\in\{1,\dots,N_p\},$$

so that the numerical initial condition have the total mass of that in (5.13):

$$\int_{\mathbf{\Omega_x}\times\mathbb{R}^2} f_0(\mathbf{x},\mathbf{v})\,d\mathbf{x}d\mathbf{v} = \int_{\mathbf{\Omega_x}\times\mathbb{R}^2} f_0^{N_p}(\mathbf{x},\mathbf{v})\,d\mathbf{x}d\mathbf{v} = T_1T_2. \tag{5.15}$$

We solve numerically (1.3)-(1.5) by using classical periodic boundary conditions on the physical domain ([16]). In this way, we take $n_i = 1$ in (1.4) ensuring that

$$\int_{\mathbf{\Omega_x}} \left(\rho^S(\mathbf{x},t) - n_i\right) d\mathbf{x} = 0, \tag{5.16}$$

and thus, that the Poisson equation in (1.4) has solution. Then, the Poisson equation is solved by means of a Fast Fourier Transform method using 128 cells in the x_1-direction and 16 cells in the x_2-direction. As for the particles' advection in time, when computing the reference solution or within the first and the third step of Algorithm 3.5, we use the explicit fourth order Runge-Kutta scheme with $\Delta t = 2\pi\varepsilon/100$.

For several small ε, in Fig. 7 we plot the global error in densities, at final time $t = 4$. More precisely, after each time step we compute the grid densities corresponding to the reference solution and to the ETD-PIC scheme, by using cubic splines. The local error is thus the L^2-norm

$$||g||_{L^2} = \left(\frac{1}{T_1 T_2}\int_{\mathbf{\Omega_x}} |g(\mathbf{x})|^2 d\mathbf{x}\right)^{1/2}$$

of the difference between these densities; its calculation was done by the trapezoidal rule. We have done tests for several values of the time step going from 0.1 to 1. Thus, the smaller time step is sufficiently big so that N, the whole number of rapid periods enclosed in a time step, be greater than 1. More precisely, following equation (3.7), N varies from 1 to 1591 when the values of ε and of the time step are those in Fig. 7.

These results show that the ETD-PIC scheme works uniformly when ε vanishes. In addition, as already pointed out in the linear case in the previous section, the smaller ε is, the smaller the displacement of a particle's position is. Therefore, the smaller is the error due to the second step of Algorithm 3.5, and thus, the better the scheme performs.

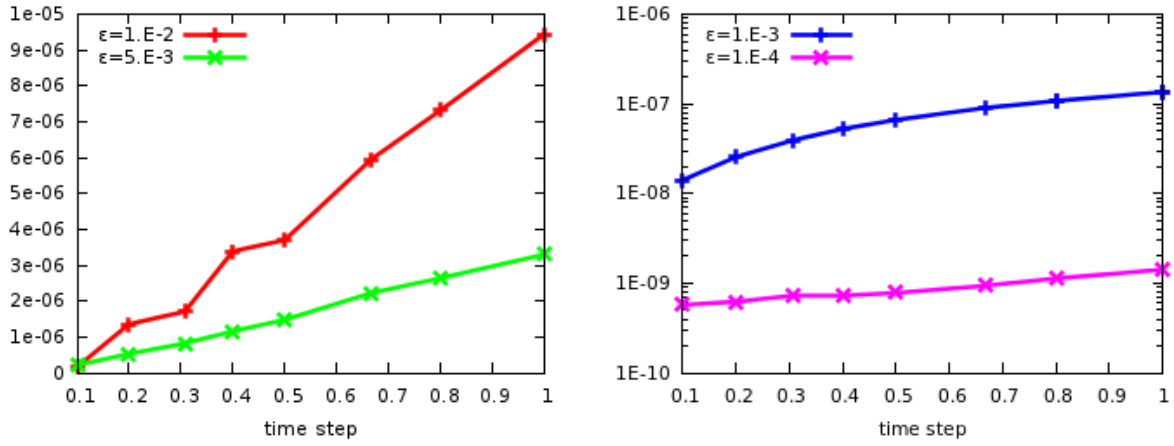

Figure 7: Vlasov Poisson case with $\varepsilon = 0.01$, $\varepsilon = 0.005$, $\varepsilon = 0.001$, and $\varepsilon = 0.0001$. The global error at time 4 of the ETD-PIC method with respect to a reference solution.

5.3 Long time Vlasov-Poisson test case

Now, we study the behaviour of the time-stepping scheme for long time simulation, more precisely for times of order $1/\varepsilon$ when the fast periodic motion is considered of order ε. The outcome of a simulation of this type was already illustrated in Fig. 2 in the linear case presented above. In this section we do not compare the results obtained with the ETD-PIC scheme to a reference solution since it would require very large CPU time. We will thus do numerical comparisons with respect to a macroscopic free of oscillations model, which can be simulated with time steps of order 1. Therefore, we are led to take into account the Guiding Center model which is a good approximation when ε vanishes of the Vlasov-Poisson system (1.3)-(1.5) (see [2], [10]).

5.3.1 The Vlasov model vs. the Guiding Center model

Let f^ε be the solution of (1.3)-(1.5). In order to see what happens for large final times, we introduce the function g^ε defined by :

$$g^\varepsilon(\mathbf{x},\mathbf{v},t) = f^\varepsilon(\mathbf{x},\mathbf{v},\frac{t}{\varepsilon}). \tag{5.17}$$

Then, the function g^ε satisfy the following system of equations :

$$\partial_t g^\varepsilon + \frac{\mathbf{v}}{\varepsilon}\cdot\nabla_{\mathbf{x}} g^\varepsilon + \frac{1}{\varepsilon}\left(\mathbf{E}^\varepsilon(\mathbf{x},\frac{t}{\varepsilon}) + \frac{1}{\varepsilon}\mathbf{v}^\perp\right)\cdot\nabla_{\mathbf{v}} g^\varepsilon = 0, \tag{5.18}$$

$$\mathbf{E}^\varepsilon(\mathbf{x},t) = -\nabla_{\mathbf{x}}\phi^\varepsilon(\mathbf{x},t), \quad -\Delta_{\mathbf{x}}\phi^\varepsilon(\mathbf{x},\frac{t}{\varepsilon}) = \int_{\mathbb{R}^2} g^\varepsilon(\mathbf{x},\mathbf{v},t)d\mathbf{v} - n_i, \tag{5.19}$$

$$g^\varepsilon(\mathbf{x},\mathbf{v},t=0) = f_0(\mathbf{x},\mathbf{v}). \tag{5.20}$$

Setting

$$\boldsymbol{\mathcal{E}}^\varepsilon(\mathbf{x},t) = \mathbf{E}^\varepsilon(\mathbf{x},\frac{t}{\varepsilon}), \tag{5.21}$$

$$\psi^\varepsilon(\mathbf{x},t) = \phi^\varepsilon(\mathbf{x},\frac{t}{\varepsilon}), \tag{5.22}$$

we obtain the following dimensionless Vlasov-Poisson system :

$$\partial_t g^\varepsilon + \frac{\mathbf{v}}{\varepsilon}\cdot\nabla_{\mathbf{x}} g^\varepsilon + \frac{1}{\varepsilon}\left(\boldsymbol{\mathcal{E}}^\varepsilon(\mathbf{x},t) + \frac{1}{\varepsilon}\mathbf{v}^\perp\right)\cdot\nabla_{\mathbf{v}} g^\varepsilon = 0, \tag{5.23}$$

$$\boldsymbol{\mathcal{E}}^\varepsilon(\mathbf{x},t) = -\nabla_{\mathbf{x}}\psi^\varepsilon(\mathbf{x},t), \quad -\Delta_{\mathbf{x}}\psi^\varepsilon(\mathbf{x},t) = \int_{\mathbb{R}^2} g^\varepsilon(\mathbf{x},\mathbf{v},t)d\mathbf{v} - n_i, \tag{5.24}$$

$$g^\varepsilon(\mathbf{x},\mathbf{v},t=0) = f_0(\mathbf{x},\mathbf{v}). \tag{5.25}$$

It is well known (see [2]) that, under some hypotheses for f_0, the particle density associated to the dynamical system (5.23)-(5.25) weak-$*$ converges when ε goes to zero towards the unique solution to the Guiding Center equation :

$$\partial_t f_{GC} + \mathbf{E}^\perp\cdot\nabla_{\mathbf{x}} f_{GC} = 0, \tag{5.26}$$

$$\mathbf{E}(\mathbf{x},t) = -\nabla_{\mathbf{x}}\phi(\mathbf{x},t), \quad -\Delta_{\mathbf{x}}\phi(\mathbf{x},t) = f_{GC} - n_i, \tag{5.27}$$

$$f_{GC}(\mathbf{x},t=0) = \int_{\mathbb{R}^2} f_0(\mathbf{x},\mathbf{v})\,d\mathbf{v}. \tag{5.28}$$

In order to test the long time accuracy of the ETD-PIC scheme we compute

$$\boldsymbol{\rho}_g^{\varepsilon}(\mathbf{x},t) = \int_{\mathbb{R}^2} g^{\varepsilon}(\mathbf{x},\mathbf{v},t)\, d\mathbf{v} \tag{5.29}$$

and we compare the result with f_{GC}. Notice that we need to replace $2\pi\varepsilon$ by $2\pi\varepsilon^2$ in Algorithm 3.5, since this is the order of the oscillations period in the Vlasov equation in (5.23). Therefore, the large final times appearing in this section (*e.g.* Fig. 8) are of order 1.

5.3.2 Numerical results

Now, we compare the numerical result of the ETD-PIC scheme for solving the ε-dependent system (5.23)-(5.25) to that of a standard scheme for the Guiding Center model (5.26)-(5.28). For the numerical simulations of these equations we follow the steps of the classical PIC method described in Section 2. We choose the following initial condition (the Kelvin-Helmholtz instability test case, see [15], [4], [16])

$$f_0(\mathbf{x},\mathbf{v}) = \frac{1}{2\pi T_1 T_2} \exp\left(-\frac{v_1^2+v_2^2}{2}\right)\big(\sin(x_2) + \eta\,\cos(k_1 x_1)\big), \tag{5.30}$$

defined in $\boldsymbol{\Omega}_{\mathbf{x}} \times \mathbb{R}^2$, where $\boldsymbol{\Omega}_{\mathbf{x}} = [0;T_1]\times[0;T_2]$, $T_1 = 4\pi$, $T_2 = 2\pi$, $\eta = 0.05$, and $k_1 = 2\pi/T_1$. We take $N_p = 10^5$ macroparticles. Moreover we take 32 cells in the x_1-direction and 32 cells in the x_2-direction for the construction of the physical mesh. In order that the Poisson equation with periodic boundary conditions be solvable, we take $n_i = 0$ in equations (5.24) and (5.27), since the integral over $\boldsymbol{\Omega}_{\mathbf{x}} \times \mathbb{R}^2$ in $(\mathbf{x},\mathbf{v})$ of the initial condition in (5.30) is 0. As in the previous section, we solve the Poisson equation by an usual Fast Fourier Transform. The time step used for the fourth order Runge-Kutta scheme for pushing particles within the first and the third step of Algorithm 3.5 is $2\pi\varepsilon/100$. Then, the particles used for the Guiding Center model are pushed in time with the fourth order Runge-Kutta scheme. We have done tests with several time steps and several values of ε going from 0.001 to 0.01. In Fig. 8 we observe the time evolution of the particles, first, by using the Guiding Center model and second, the ETD-PIC method for the Vlasov-Poisson model with $\varepsilon = 0.005$. The time step is $\Delta t = 0.01$. More precisely, we represent in the physical space the contours of the particle densities. These smooth densities in Fig. 8 are calculated by depositing $8\cdot 10^5$ macroparticles on a mesh with 128×64 cells by using cubic splines.

In Fig. 9 we represent the "global error" in densities (see Section 5.2.1) at $t = 5$. Now, the "local error" is the difference between the discretization of the solution to the Vlasov-Poisson system, computed with the ETD-PIC strategy, and the discretization of the solution to the Guiding Center equation. Eventually, we show in Tables 2 and 3 how big is the time step of the ETD-PIC method with respect to the fastest periodic motion (see formula (3.7) with ε^2 instead of ε for the calculation of these numbers).

As a last validation of our scheme, we follow the time evolution of the Fourier coefficient $(1,1)$ of the electric field, solution to the Poisson equation in (5.24). In order to be in agreement with works [15] and [4], we rather take

$$f_0(\mathbf{x},\mathbf{v}) = \frac{1}{2\pi T_1 T_2} \exp\left(-\frac{v_1^2+v_2^2}{2}\right)\big(\sin(x_1) + \eta\,\cos(k_2 x_2)\big) \tag{5.31}$$

as initial condition. In the sequel, for simplicity we denote k_2 by k. More precisely, for small η, we can use (see [16]) a linear approximation of the Vlasov-Poisson system and thus we can approximate the electric field by an analytic solution

$$\boldsymbol{\mathcal{E}}^{\varepsilon}(\mathbf{x},t) \approx 4\eta r e^{\omega_i t}\sin(kx_2)\cos(\omega_r t - \Phi), \tag{5.32}$$

where ω_r and ω_i are the real and the imaginary parts of ω, the dominant complex root of the function involved in the dispertion relation and $re^{i\Phi}$ is the residue associated to $\omega = \omega_r + i\omega_i$ (see [16]). The approximation in (5.32) turns out to be a very good one on some time interval, between $t = 5$ and $t = 10$. From (5.32) we can easily deduce ω_i, the slope of the line approximating the evolution in time of the electric field. The growth rate corresponds to this slope and it can be calculated from the Fourier coefficient $(1,1)$ of the electric field $\boldsymbol{\mathcal{E}}^{\varepsilon}$. It has been noticed in [15] that the numerical growth rate can also be obtained through an eigenvalue equation (see the Appendix). In Figs. 10 and 12, for a fixed ε and several values of k, we show that the evolution in time of the logarithm of the absolute value of the real part of the $(1,1)$ Fourier coefficient of ψ^{ε} (obtained with the ETD-PIC scheme) converges numerically towards the corresponding numerical growth rates obtained through the eigenvalue equation. In Fig. 11, we can observe for a fixed k and several values of ε that the linear phase of the time evolution of the Fourier coefficient has the good slope given by the eigenvalue equation.

5.3.3 Comments about the numerical results

First, we can see in Fig. 8 that for several large final times, the particle densities obtained with the Guiding Center equation and with the ETD-PIC scheme for Vlasov-Poisson model with $\varepsilon = 0.005$ are very close. This is a first step of validation of our method in the context of long time simulations.

	Δt=1E-3	Δt=3E-3	Δt=5E-3	Δt= 7E-3	Δt=9E-3	Δt=1E-2
ε =5.E-3	6	19	31	44	57	63
ε =2.5E-3	25	76	127	178	229	254
ε =1.E-3	159	477	795	1 114	1 432	1 591
ε =1.E-4	15 915	47 746	79 577	111 408	143 239	159 154

Table 2: The whole number of rapid full tours enclosed in a time step of the ETD-PIC scheme, for several values of the time step and of ε; related to the *left* panel in Fig. 9

	ε =1E-3	ε =3E-3	ε =5E-3	ε = 7E-3	ε =9E-3	ε =1E-2
Δt=1.E-2	1591	176	63	32	19	15
Δt=5.E-3	795	88	31	16	9	7

Table 3: The whole number of rapid full tours enclosed in a time step of the ETD-PIC scheme, for several values of ε and of the time step; related to the *right* panel in Fig. 9

Next, we discuss the results concerning the behaviour of the global error. Recalling that N is the integral number of the rapid full tours appearing in the second step of Algorithm 3.5, we remark the following:

1. First, in Fig. 9 (left panel), we can see that for each fixed ε, the error decreases with decreasing time step, although N is changing. Thus, the scheme works for small time steps compared to the fast oscillation and is robust with respect to N. Second, we observe that, the smaller ε is, the smaller the error is, despite that N is significantly increasing when ε is smaller (see Table 2). This is an expected behaviour since the Guiding Center model becomes a better approximation of the Vlasov-Poisson model when ε goes to 0. Thus, the scheme works for big time steps with respect to the fast oscillation also.

2. In Table 3 and the right panel in Fig. 9, we detail the above comments by taking several values of ε. First, justifying as in the item before, when the time step is kept fixed, the error decreases with decreasing ε. Second, we notice once again the robustness of the scheme: the errors are stable when N is widely varying from 7 to 1 591.

3. Concluding, the left panel in Fig. 9 shows that the ETD-PIC scheme is convergent when $\Delta t \to 0$, uniformly in ε. Also, the right panel shows that the discretization of the solution to the Vlasov-Poisson system converges when $\varepsilon \to 0$ to a discretization of the Guiding Center model, independently of Δt. These facts underline the asymptotic preserving behaviour of our scheme.

$\frac{\omega}{k}$	0.1556	0.2699	0.3657	0.4492	0.5223	0.5850	0.6361	0.6735	0.6920
$1-k$	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9

Table 4: The numerical values of ω/k (the growth rate is ω) as a function of $1-k$, obtained by solving the eigenvalue equation (see Appendix)

A Appendix: the eigenvalue equation

The eigenvalue equation allowing to obtain the growth rates of instability related to the Guiding-center problem (5.26)-(5.28) has been derived in [15]. We recall this derivation, in our notations.
We consider an inhomogeneous equilibrium solution $f^0 = f^0(x_1)$ and the corresponding equilibrium potential $\phi^0 = \phi^0(x_1)$, which satisfies :

$$\frac{\partial^2 \phi^0}{\partial x_1^2} = -f^0. \tag{A.1}$$

In practice, according to our choice of initial condition in (5.31), we will take $f^0(x_1) = \sin(x_1)$ and we will work on the torus, *i.e.*, with periodic boundary conditions.

Then we will linearize (5.26)-(5.27) around this equilibrium solution. In other words we are looking for a solution of the form :

$$f_{GC}(t,\mathbf{x}) \simeq f^0(x_1) + \eta f^1(t,\mathbf{x}), \tag{A.2}$$
$$\phi(t,\mathbf{x}) \simeq \phi^0(x_1) + \eta \phi^1(t,\mathbf{x}), \tag{A.3}$$

where η is a small parameter. Injecting (A.2)-(A.3) in (5.26)-(5.27) we obtain :

$$\partial_t f^1 - \partial_{x_2}\phi^1 \partial_{x_1} f^0 + \partial_{x_1}\phi^0 \partial_{x_2} f^1 = O(\eta), \tag{A.4}$$
$$-\frac{\partial^2 \phi^1}{\partial x_1^2} - \frac{\partial^2 \phi^1}{\partial x_2^2} = f^1. \tag{A.5}$$

Neglecting the $O(\eta)$ terms in (A.4) yields the following linearized problem :

$$\partial_t f^1 - \partial_{x_2}\phi^1 \partial_{x_1} f^0 + \partial_{x_1}\phi^0 \partial_{x_2} f^1 = 0, \tag{A.6}$$
$$-\frac{\partial^2 \phi^1}{\partial x_1^2} - \frac{\partial^2 \phi^1}{\partial x_2^2} = f^1. \tag{A.7}$$

Now, we are looking for a couple of solution (f^1, ϕ^1) of (A.6)-(A.7) in the form :

$$f^1(t,\mathbf{x}) = \widehat{f}_{1,l}(x_1)\, e^{ilx_2} e^{-i\omega t}, \tag{A.8}$$
$$\phi^1(t,\mathbf{x}) = \widehat{\phi}_{1,l}(x_1)\, e^{ilx_2} e^{-i\omega t}. \tag{A.9}$$

Injecting (A.8)-(A.9) in (A.6)-(A.7) yields :

$$-i\omega \widehat{f}_{1,l} + il\widehat{\phi}_{1,l}\partial_{x_1} f^0 + il\widehat{f}_{1,l}\partial_{x_1}\phi^0 = 0, \tag{A.10}$$
$$-\frac{\partial^2 \widehat{\phi}_{1,l}}{\partial x_1^2} + l^2 \widehat{\phi}_{1,l} = \widehat{f}_{1,l}. \tag{A.11}$$

Using (A.10) we express $\widehat{f}_{1,l}$ in terms of $\widehat{\phi}_{1,l}$. Afterwards, injecting this expression in (A.11) we get the following equation on $\widehat{\phi}_{1,l}$:

$$(c - v_0)\left(\frac{\partial^2 \widehat{\phi}_{1,l}}{\partial x_1^2} - l^2 \widehat{\phi}_{1,l}\right) - \frac{\partial^2 v_0}{\partial x_1^2}\widehat{\phi}_{1,l} = 0, \tag{A.12}$$

where

$$c = \frac{\omega}{l}, \tag{A.13}$$
$$v_0(x_1) = \partial_{x_1}\phi^0(x_1). \tag{A.14}$$

In practice we will take :

$$\begin{aligned} f^0(x_1) &= \sin(x_1), \\ \phi^0(x_1) &= \sin(x_1), \\ v_0(x_1) &= \cos(x_1). \end{aligned} \tag{A.15}$$

Constructing an uniform grid of $[0, T_1]$, where T_1 is the period of f^0 and ϕ^0, we can proceed in the numerical resolution of (A.12). We make the following discretisation:

$$\widehat{\phi}_{1,l}\left(x_1^i\right) = \phi_i^l, \quad \partial_{x_1}\widehat{\phi}_{1,l}\left(x_1^i\right) = \frac{\phi_{i+1}^l - \phi_{i-1}^l}{2\Delta x_1}, \quad \frac{\partial^2\widehat{\phi}_{1,l}}{\partial x_1^2}\left(x_1^i\right) = \frac{\phi_{i+1}^l - 2\phi_i^l + \phi_{i-1}^l}{\Delta x_1^2} \tag{A.16}$$

and we obtain :

$$\begin{aligned}&\left(v_0\left(x_1^i\right)\phi_{i+1}^l + \left(-2v_0\left(x_1^i\right) - l^2\Delta x_1^2 v_0\left(x_1^i\right) - \frac{\partial^2 v_0}{\partial x_1^2}\left(x_1^i\right)\Delta x_1^2\right)\phi_i^l + v_0\left(x_1^i\right)\phi_{i-1}^l\right)\\ =&c\left(\phi_{i+1}^l - \left(2 + l^2\Delta x_2^2\right)\phi_i^l + \phi_{i-1}^l\right).\end{aligned} \tag{A.17}$$

The problem can then be written as $A^l\phi^l = cB^l\phi^l$. Consequently, the initial problem consisting in finding ω and $\widehat{\phi}_{1,l}$ satisfying equation (A.12) can be rewritten under the following eigenvalues problem : find a vector ϕ^l and a complex number c such that $(B^l)^{-1}A^l\phi^l = c\phi^l$. Then the instability growth rate corresponds to the greatest imaginary part of the eigenvalues. Considering different values of the wave number k, it is possible to plot the quantity ω/k (where ω is the growth rate) as a function of $1 - k$. This is performed in Table 4.

References

[1] C.K. Birdsall and A.B. Langdon. *Plasma Physics via Computer Simulation.* Institute of Physics, Bristol and Philadelphia, 1991.

[2] M. Bostan. The Vlasov-Maxwell System with Strong Initial Magnetic Field: Guiding-Center Approximation. *Multiscale Model. Simul.*, 6(3):1026–1058, 2007.

[3] S. M. Cox and P. C. Matthews. Exponential time differencing for stiff systems. *J. Comput. Phys.*, 176(2):430–455, 2002.

[4] N. Crouseilles, M. Mehrenberger, and E. Sonnendrücker. Conservative semi-Lagrangian schemes for Vlasov equations. *J. Comput. Phys.*, 229(6):1927–1953, 2010.

[5] D. H. E. Dubin, J. A. Krommes, C. Oberman, and W. W. Lee. Nonlinear gyrokinetic equations. *Phys. Fluids*, 26(12):3524–3535, 1983.

[6] E. Frénod, S. A. Hirstoaga, and E. Sonnendrücker. An exponential integrator for a highly oscillatory Vlasov equation. *Discrete Contin. Dyn. Syst. Ser. S*, 2014 (In Press).

[7] E. Frénod, P. A. Raviart, and E. Sonnendrücker. Asymptotic expansion of the Vlasov equation in a large external magnetic field. *J. Math. Pures et Appl.*, 80(8):815–843, 2001.

[8] E. Frénod, F. Salvarani, and E. Sonnendrücker. Long time simulation of a beam in a periodic focusing channel via a two-scale PIC-method. *Math. Models Methods Appl. Sci.*, 19(2):175–197, 2009.

[9] E. Frénod and E. Sonnendrücker. Homogenization of the Vlasov equation and of the Vlasov-Poisson system with a strong external magnetic field. *Asymptot. Anal.*, 18(3,4):193–214, 1998.

[10] E. Frénod and E. Sonnendrücker. Long time behavior of the two dimensionnal Vlasov equation with a strong external magnetic field. *Math. Models Methods Appl. Sci.*, 10(4):539–553, 2000.

[11] F. Golse and L. Saint Raymond. The Vlasov-Poisson system with strong magnetic field. *J. Math. Pures. Appl.*, 78:791–817, 1999.

[12] M. Hochbrück and A. Ostermann. Exponential integrators. *Acta Numer.*, 19:209–286, 2010.

[13] S. Jin. Efficient asymptotic-preserving (AP) schemes for some multiscale kinetic equations. *SIAM J. Sci. Comput.*, 21:441–454, 1999.

[14] W. W. Lee. Gyrokinetic approach in particle simulation. *Phys. Fluids*, 26(2):556–562, 1983.

[15] M. Shoucri. A two-level implicit scheme for the numerical solution of the linearized vorticity equation. *Internat. J. Numer. Methods Engrg.*, 17:1525–1538, 1981.

[16] E. Sonnendrücker. *Approximation numérique des équations de Vlasov-Maxwell*, 2010. notes de cours de M2, Université de Strasbourg.

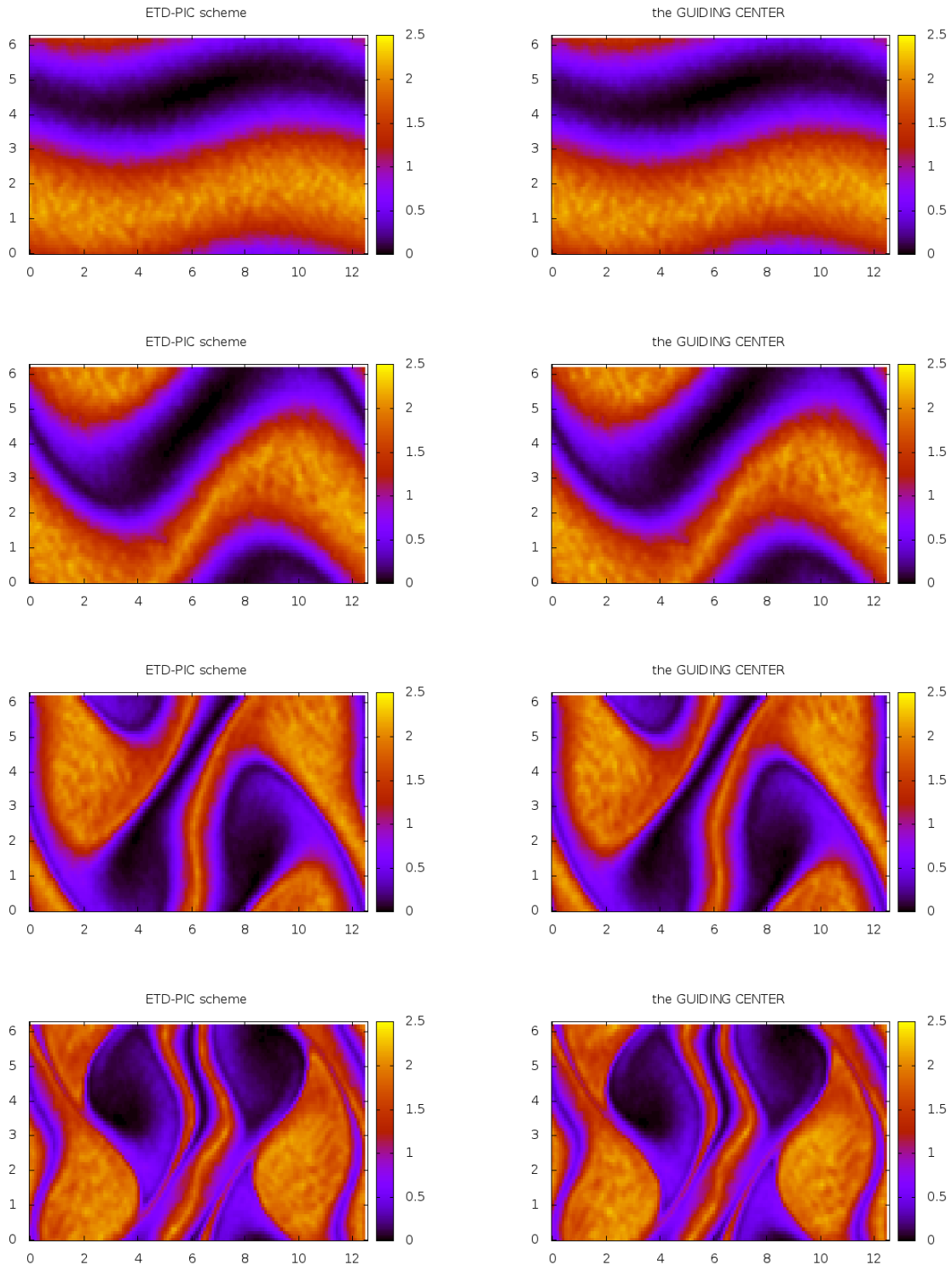

Figure 8: Simulations of $\boldsymbol{\rho}_g^\varepsilon$ with the ETD-PIC method when $\varepsilon = 0.005$ (left) and of the Guiding Center distribution function (right). From top to bottom we represent the densities' contours at times $t = 5$, $t = 10$, $t = 15$, and $t = 20$. The time step is $\Delta t = 0.01$ (The number of fast periods in Δt is $N = 63$).

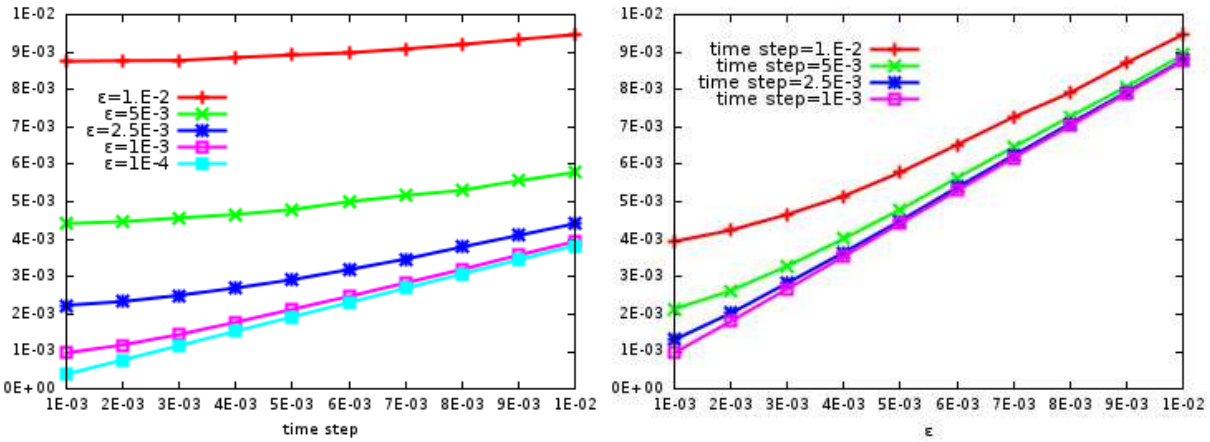

Figure 9: The difference between the solution computed with the ETD-PIC method and the solution to the Guiding Center equation as a function of the time step (*left* panel) and of ε (*right* panel) at final time $t = 5$

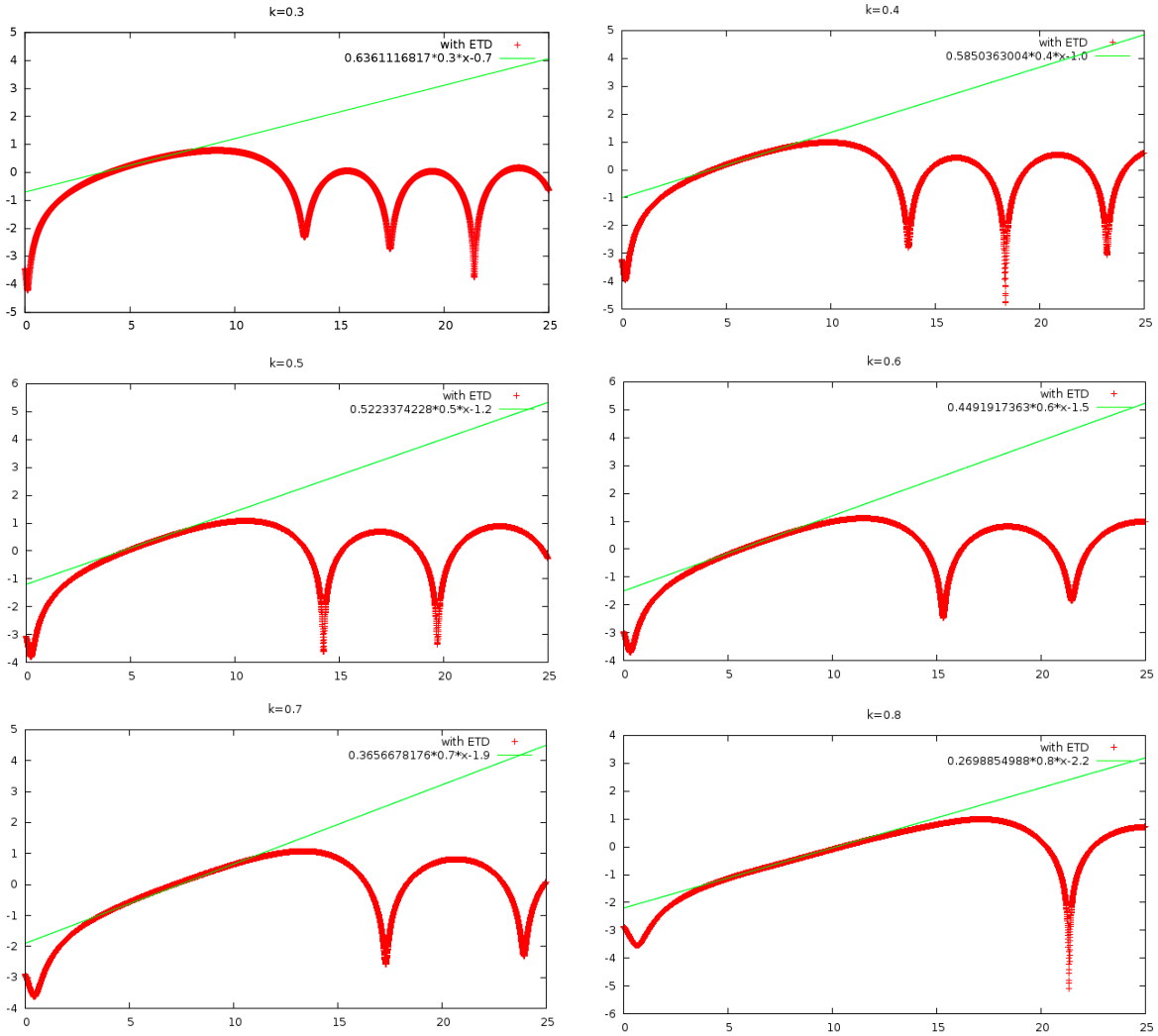

Figure 10: In red, the evolution in time of the logarithm of the absolute value of the $(1,1)$ Fourier coefficient of ψ^ε. In green, the growth rate obtained through the eigenvalue equation (see Table 4). Simulations with fixed $\Delta t = \varepsilon = 0.005$ and several values of k between 0.3 and 0.8 in the definition of the initial condition in (5.31)

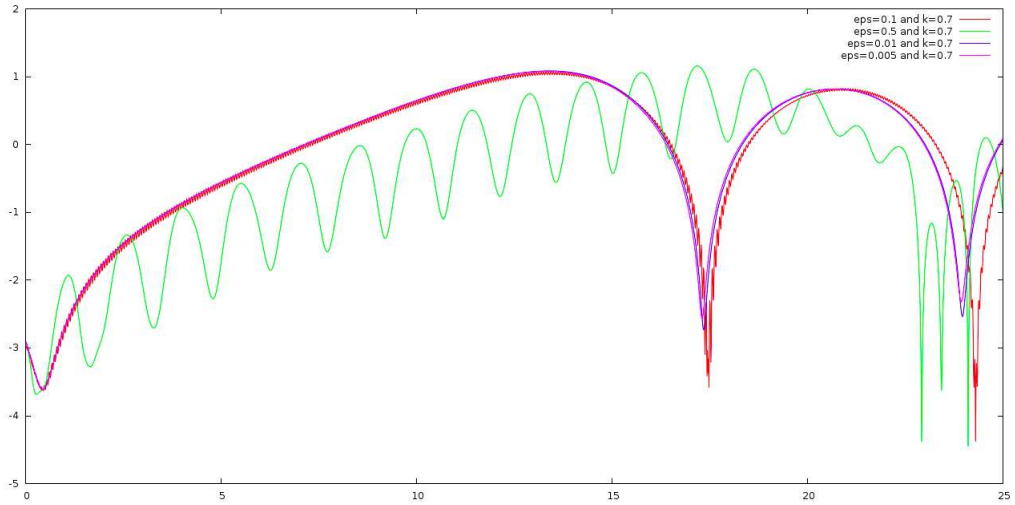

Figure 11: The growth rates for fixed $k = 0.7$ and several values of ε from 0.5 to 0.005.

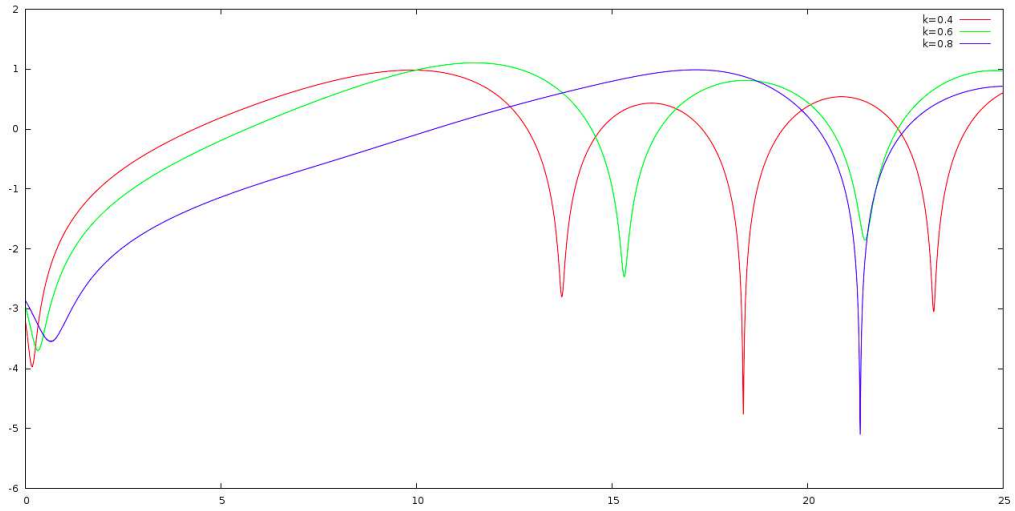

Figure 12: The growth rates for fixed $\varepsilon = 0.005$ and three values of k.