Uniformly accurate forward semi-Lagrangian methods for highly oscillatory Vlasov-Poisson equations.

Nicolas Crouseilles, Mohammed Lemou, Florian Méhats, Xiaofei Zhao

UNIFORMLY ACCURATE FORWARD SEMI-LAGRANGIAN METHODS FOR HIGHLY OSCILLATORY VLASOV-POISSON EQUATIONS

NICOLAS CROUSEILLES*, MOHAMMED LEMOU †, FLORIAN MÉHATS ‡, AND XIAOFEI ZHAO §

Abstract. This work is devoted to the numerical simulation of a Vlasov-Poisson equation modeling charged particles in a beam submitted to a highly oscillatory external electric field. A numerical scheme is constructed for this model. This scheme is uniformly accurate with respect to the size of the fast time oscillations of the solution, which means that no time step refinement is required to simulate the problem. The scheme combines the forward semi-Lagrangian method with a class of Uniformly Accurate (UA) time integrators to solve the characteristics. These UA time integrators are derived by means of a two-scale formulation of the characteristics, with the introduction of an additional periodic variable. Numerical experiments are done to show the efficiency of the proposed methods compared to conventional approaches.

Key words. Kinetic models, Vlasov-Poisson equation, Highly oscillatory, Forward semi-Lagrangian method, Uniformly accurate, Two-scale formulation

AMS subject classifications. 65L05, 65L20, 65L70

1. Introduction.

This work is devoted to the construction of efficient numerical schemes for collisionless kinetic equations which involve high oscillations in time. As in [11], we focus on the case of a charged particle beam in the paraxial approximation. This model corresponds to an approximation of the Vlasov-Maxwell equations (see [13, 15] for more details). If in addition, the beam is assumed axisymmetric, the model writes

$$\partial_t f + \frac{v}{\varepsilon}\partial_r f + \left(E - \frac{r}{\varepsilon} + E_{\rm ext}\right)\partial_v f = 0, \quad t > 0,\ r, v \in \mathbb{R}, \tag{1.1a}$$

$$\partial_r (rE) = r \int_{\mathbb{R}} f \mathrm{d}v, \quad t > 0,\ r \in \mathbb{R}, \tag{1.1b}$$

$$E_{\rm ext}(t, r) = a\left(\frac{t}{\varepsilon}\right) r, \quad t > 0,\ r \in \mathbb{R}, \tag{1.1c}$$

$$f(0, r, v) = f_0(r, v), \quad r, v \in \mathbb{R}. \tag{1.1d}$$

where the unknown $f = f(t, r, v)$ is a real-valued scalar field denoting the distribution function of the charged particles, $t \geq 0$ is the longitudinal position coordinate (interpreted as the time variable), $r \in \mathbb{R}$ is the radial coordinate and v is the radial velocity. The parameter $\varepsilon > 0$ denotes the ratio between the characteristic lengths in the transverse and the longitudinal directions, and $f_0(r, v)$ is a given initial data independent of ε. The total electric field is decomposed into two parts: first, an external part $(-r/\varepsilon + E_{\rm ext}(t, r))$ in which $a(\cdot)$ is a given 2π-periodic function known

*INRIA-Rennes, Université de Rennes 1 and IRMAR, Campus de Beaulieu, 35042 Rennes Cedex, France. (nicolas.crouseilles@inria.fr)

†CNRS, Université de Rennes 1, INRIA-Rennes and IRMAR, Campus de Beaulieu, 35042 Rennes Cedex, France. (mohammed.lemou@univ-rennes1.fr)

‡Université de Rennes 1, INRIA-Rennes and IRMAR, Campus de Beaulieu, 35042 Rennes Cedex, France. (florian.mehats@univ-rennes1.fr)

§Université de Rennes 1, IRMAR, Campus de Beaulieu, 35042 Rennes Cedex, France. (zhxfnus@gmail.com)

as the tension function and second $E(t,r)$ is the so-called self-consistent electric field defined by the Poisson equation (1.1b) and given by

$$E(t,r) = \frac{1}{r}\int_0^r \int_{\mathbb{R}} sf(t,s,v)\mathrm{d}v\mathrm{d}s, \quad t>0,\ r>0. \tag{1.2}$$

When the dimensionless parameter ε is small, the Vlasov-Poisson equation (1.1) will exhibit high oscillations in time direction with wavelength $O(\varepsilon)$. In this regime, conventional numerical methods suffer from severe numerical burden, since the time step needs to be chosen small enough in order to fully resolve the oscillations wavelength. Besides the prohibitive computational cost, most of the classical methods need some interpolations in phase space at each time step which would potentially cause numerical diffusions. Thus, conventional methods have indeed difficulties to capture the correct long time dynamics of the Vlasov-Poisson equation in the highly oscillatory limit regime, due to the trade-off of the temporal and spatial error. When the asymptotic regime is considered, some efficient numerical methods can be constructed by considering the asymptotic averaged model ($\varepsilon \to 0$ in (1.1)) derived using the notion of two-scale convergence (see [1, 27, 20]). This is the purpose of [19, 25]. In [11], a scheme is proposed to solve the problem (1.1) for arbitrary value of ε. In particular, the so-obtained scheme allows the use of step size independent of ε, and it is consistent with the two-scale limit. In particular, this scheme enjoys the so-called *Asymptotic Preserving* property (see [22]). Actually, this scheme is derived by applying a two-scale formulation (see also [9]) to (1.1), which extends the two dimensional problem (1.1) into three dimensions by considering a fast variable $\tau = t/\varepsilon$. Then, considering well-chosen initial data enables to confer on the scheme a stronger property of *uniform accuracy*. This means that the order of accuracy of the scheme is independent of the parameter ε. Note that such a property has been proved in [9] for another class of equations. However, the scheme introduced in [11] presents some drawbacks: first, the use of Lax-Wendroff type finite difference discretization makes appear a standard CFL condition linking the time step to the size of the spatial mesh; second, strong numerical diffusion affects the long time simulations.

In this paper, we construct a uniformly accurate semi-Lagrangian based numerical scheme for (1.1). The use of a semi-Lagrangian method enables to overcome the drawbacks of the method introduced in [11] mentioned above, leading to a very accurate and efficient way to solve (1.1) for arbitrarily small ε, using fixed numerical parameters. To do so, we consider the characteristic equations associated to (1.1), on which the two-scale formulation is applied. Then, the so-obtained augmented characteristics equation depends not only on time but also on the fast variable τ. By evaluating the solution $U(t,\tau)$ of the augmented problem on the diagonal $\tau = t/\varepsilon$ enables to recover the original time-dependent characteristics solution. In addition to a semi-implicit time integrator, an initial condition has to be chosen for the augmented problem. A suitable initial data for the augmented problem is constructed using a micro-macro decomposition (following [11, 9]) so that U is smooth with respect to time, uniformly in ε. Of course, this initial data for the augmented problem fits with any initial data of the original problem. Hence, using Forward Semi-Lagrangian method (FSL), the distribution function is reconstructed on the phase space. This results in a class of uniformly accurate (UA) numerical methods for the Vlasov-Poisson equation (1.1) for all $\varepsilon \in (0,1]$ which do not contain any CFL conditions and which perform well for long time simulations of (1.1). Let us remark that another approach based on an exponential time integrator of the characteristics has been proposed in [16, 17, 18]

within the framework of Particle In Cell (see [7]) approximation of highly-oscillatory Vlasov equations.

The rest of the paper is organized as follows. In Section 2, we introduce the framework of forward semi-Lagrangian method. In Section 3, we derive the UA schemes. Numerical results are reported in Section 4. Finally, some conclusions are drawn in Section 5.

2. Forward semi-Lagrangian method. Semi-Lagrangian methods which were introduced firstly for meteorology in [29], are very popular in solving the kinetic equations. They take advantage of both Lagrangian (which consider macro-particles moving according to the characteristics) and Eulerian (which look at values of the unknown on a phase space grid) approaches. By making use of the invariance of the distribution function along the trajectories, the semi-Lagrangian methods do not need restrictions on the step size. Most of the semi-Lagrangian methods in the literature are *back* tracing the mesh grid in time and end up with an interpolation step. A forward version of the semi-Lagrangian method was proposed in [12] (see also [14, 26, 28]), which propagates the mesh grids *forward* in time and use a deposition step to update the function values on the phase space mesh. This approach is not restricted by CFL conditions, preserves the total mass and enables to use classical time integrator for the approximation of the characteristics. Then, it makes possible to combine it with the framework of two-scale formulation introduced in [11, 9].

An important step in the framework of the forward semi-Lagrangian method is the reconstruction of the distribution function values on the mesh grids by using the information on some non-grid points. Considering as initial data the mesh grid points, the characteristics are propagated during one time step, so that the ending points of the mesh grids do not coincide with grid points. To reconstruct the distribution function on the grid points, high order piecewise polynomials have to be used, like cubic spline in [12]. However, to guarantee the expected order, one needs to make sure displacement of the mesh grids along the characteristics for one time step is not too large (see [24]). For the highly oscillatory Vlasov-Poisson equation (1.1) when $0 < \varepsilon \ll 1$, the rotating speed in the characteristics is very large, which would actually cause large displacement. In order to ensure a good behavior of the forward semi-Lagrangian method, we consider the Vlasov-Poisson equation in a rotating frame, which filters out the leading order oscillations (see also [11]). To do so, let us consider the following change of variables

$$\begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} = \begin{pmatrix} \cos(t/\varepsilon) & -\sin(t/\varepsilon) \\ \sin(t/\varepsilon) & \cos(t/\varepsilon) \end{pmatrix} \begin{pmatrix} r \\ v \end{pmatrix}, \qquad t > 0,\ r,\, v \in \mathbb{R}, \tag{2.1}$$

and define the new unknown

$$g(t,\xi) := f\left(t, \cos(t/\varepsilon)\xi_1 + \sin(t/\varepsilon)\xi_2, -\sin(t/\varepsilon)\xi_1 + \cos(t/\varepsilon)\xi_2\right),$$

for $\xi = (\xi_1, \xi_2)^T \in \mathbb{R}^2$, $t > 0$. The Vlasov-Poisson equation (1.1) in the rotating frame reads

$$\begin{aligned} &\partial_t g(t,\xi) + \left[\mathcal{E}_{\text{ext}}(t/\varepsilon,\xi) + \mathcal{E}(t,t/\varepsilon,\xi)\right]\left[-\sin(t/\varepsilon)\,\partial_{\xi_1} g(t,\xi) + \cos(t/\varepsilon)\,\partial_{\xi_2} g(t,\xi)\right] = 0, \\ &g(0,\xi) = f_0(\xi_1,\xi_2), \end{aligned} \tag{2.2}$$

where

$$\mathcal{E}_{\text{ext}}(t/\varepsilon,\xi) := a(t/\varepsilon)\left[\cos(t/\varepsilon)\,\xi_1 + \sin(t/\varepsilon)\,\xi_2\right],$$

and the self-consistent field (1.2) in rotating frame is given by

$$\begin{aligned}&\mathcal{E}(t,t/\varepsilon,\xi) := E(t,r(t/\varepsilon,\xi))\\ &= \frac{1}{r(t/\varepsilon,\xi)}\int_0^{r(t/\varepsilon,\xi)}\int_{\mathbb{R}} sg\left(t,\cos(t/\varepsilon)s-\sin(t/\varepsilon)v,\sin(t/\varepsilon)s+\cos(t/\varepsilon)v\right)\mathrm{d}v\mathrm{d}s,\end{aligned} \tag{2.3}$$

with $r(t/\varepsilon,\xi) := \cos(t/\varepsilon)\xi_1 + \sin(t/\varepsilon)\xi_2$.

As it is well known, the solution $g(t,\xi)$ to (2.2) is constant along the characteristics $\xi(t) = (\xi_1(t),\xi_2(t))^T \in \mathbb{R}^2$ defined by

$$\begin{cases}\dot{\xi}_1(t) = -\sin(t/\varepsilon)\left[\mathcal{E}_{\text{ext}}(t/\varepsilon,\xi)+\mathcal{E}(t,t/\varepsilon,\xi)\right],\\ \dot{\xi}_2(t) = \quad\cos(t/\varepsilon)\left[\mathcal{E}_{\text{ext}}(t/\varepsilon,\xi)+\mathcal{E}(t,t/\varepsilon,\xi)\right],\end{cases} \tag{2.4}$$

so that $g(t,\xi(t)) = g(0,\xi(0))$ for all $t \geq 0$ and arbitrary starting point $\xi(0) \in \mathbb{R}^2$. Making use of this property, the forward semi-Lagrangian method turns to solve the characteristic equations (2.4) beginning on a mesh grid $\xi(0)$ and uses a reconstruction operator to update the function value at next time. To present the details of the method, we truncate the whole space problem (2.2) onto a finite interval $[-L,L]^2 \subset \mathbb{R}^2$ which is large enough to impose the periodic boundary conditions without introducing remarkable aliasing error, i.e.

$$\begin{aligned}&\partial_t g(t,\xi) - \sin(t/\varepsilon)\left[\mathcal{E}_{\text{ext}}(t/\varepsilon,\xi)+\mathcal{E}(t,t/\varepsilon,\xi)\right]\partial_{\xi_1} g(t,\xi) \qquad (2.5a)\\ &\qquad + \cos(t/\varepsilon)\left[\mathcal{E}_{\text{ext}}(t/\varepsilon,\xi)+\mathcal{E}(t,t/\varepsilon,\xi)\right]\partial_{\xi_2} g(t,\xi) = 0, \quad t>0,\ \xi\in[-L,L]^2,\\ &g(0,\xi) = f_0(\xi), \quad \xi\in[-L,L]^2, \qquad (2.5b)\\ &g(t,\xi_1,-L) = g(t,\xi_1,L), \quad g(t,-L,\xi_2) = g(t,L,\xi_2), \quad t>0,\ \xi\in[-L,L]^2. \qquad (2.5c)\end{aligned}$$

Choose $\Delta t > 0$, and two integers $N_1 > 0$ and $N_2 > 0$, then let $t_n = n\Delta t$, $n \in \mathbb{N}$ and

$$\Delta\xi_1 = \frac{2L}{N_1}, \quad \xi_{1,k} = -L + k\Delta\xi_1, \quad \Delta\xi_2 = \frac{2L}{N_2}, \quad \xi_{2,j} = -L + j\Delta\xi_2, \quad (k,j)\in\mathcal{T},$$

with $\mathcal{T} := \{(k,j) :\ k = 0,\dots N_1, j = 0,\dots,N_2\}$. A detailed framework of the forward semi-Lagrangian method towards solving (2.5) reads as follows.

Denoting by $g^n_{k,j} \approx g(t^n,\xi_{1,k},\xi_{2,j})$ for $n\in\mathbb{N}$ and choosing $g^0_{j,k} = f_0(\xi_{1,j},\xi_{1,k})$, $(k,j)\in\mathcal{T}$, we compute the coefficients $\omega^n_{k,j}$ of the cubic splines (with standard zero normal derivative at boundaries [2]), such that

$$g^n_{k,j} = \sum_{(\ell,m)\in\mathcal{T}} \omega^n_{\ell,m} S(\xi_{1,k}-\xi_{1,\ell}) S(\xi_{2,j}-\xi_{2,m}), \qquad (k,j)\in\mathcal{T}, \tag{2.6}$$

where the cubic B-spline function $S(x)$ is defined as standard, i.e.

$$S(x) = \begin{cases}\dfrac{1}{6}(2-|x|)^3, & 1\leq|x|\leq 2,\\ \dfrac{1}{6}\left(4-6x^2+3|x|^3\right), & 0\leq|x|\leq 1,\\ 0, & \text{otherwise.}\end{cases}$$

Then we look for a numerical solution at time Δt of the characteristic equations

$$\begin{cases} \dot{\xi}^n_{1,k}(t) = -\sin\left((t+t_n)/\varepsilon\right)\left[\mathcal{E}_{\rm ext}((t+t_n)/\varepsilon, \xi^n_{k,j}) + \mathcal{E}(t+t_n, (t+t_n)/\varepsilon, \xi^n_{k,j})\right], \\ \dot{\xi}^n_{2,j}(t) = \quad \cos\left((t+t_n)/\varepsilon\right)\left[\mathcal{E}_{\rm ext}((t+t_n)/\varepsilon, \xi^n_{k,j}) + \mathcal{E}(t+t_n, (+t_n)t/\varepsilon, \xi^n_{k,j})\right], \\ \xi^n_{1,k}(0) = \xi_{1,k}, \quad \xi^n_{2,j}(0) = \xi_{2,j}, \end{cases} \tag{2.7}$$

where the initial condition coincides with grid points. The new numerical solution $g^{n+1}_{k,j}$ is given by the following deposition step

$$g^{n+1}_{k,j} = \sum_{(\ell,m)\in\mathbb{T}} \omega^n_{\ell,m} S\left(\xi_{1,k} - \xi^n_{1,\ell}(\Delta t)\right) S\left(\xi_{2,j} - \xi^n_{2,m}(\Delta t)\right), \qquad (k,j)\in\mathcal{T}. \tag{2.8}$$

After we solve (2.5) and obtain $g(t,\xi)$, we get the solution of the original Vlasov-Poisson equation (1.1) by considering the original frame

$$f(t,r,v) = g(t, \cos(t/\varepsilon)r - \sin(t/\varepsilon)v, \sin(t/\varepsilon)r + \cos(t/\varepsilon)v).$$

As a standard result, the cubic spline is known to offer fourth order accuracy. So in the deposition step, when the displacement of the mesh grids is not too large, the forward semi-Lagrangian method could give high order accuracy in spatial approximation. Optimally in practice, the spatial convergence order could be $O((\Delta\xi_1)^p + (\Delta\xi_2)^q)$ for $3 \le p, q \le 4$ (see [28, 6]).

In the above framework of the forward semi-Lagrangian method, the solution $\xi^n_{k,j}(t)$ of the characteristic equations (2.7) for $0 \le t \le \Delta t$ needs to be approximated by some accurate time integrators. However, when $0 < \varepsilon \ll 1$ in (2.4) or (2.7), conventional integrators suffers from the ε-scalability [3, 4] due to the high oscillations. Thus, in order to construct a uniformly accurate forward semi-Lagrangian method for the Vlasov-Poisson equation (1.1) in the highly oscillatory regime, we need to apply a uniformly accurate integrator to the characteristic equation (2.4).

3. Uniformly accurate solvers. In this section, we present a uniformly accurate time integrator to solve the characteristic equation (2.7) by using the two-scale formulation techniques. For the simplicity of notations, we present the method by using (2.4) (the application to (2.7) only requires a shift in time).

3.1. Two-scale formulation. By separating the fast time variable $\tau = t/\varepsilon$ and the slow time variable t in (2.4), we consider the augmented problem (see [11, 9])

$$\begin{cases} \partial_t U_1(t,\tau) + \dfrac{1}{\varepsilon}\partial_\tau U_1(t,\tau) = -\sin(\tau)\left[\mathcal{E}_{\rm ext}(\tau, U) + \mathcal{E}(t,\tau,U)\right], & \text{(3.1a)} \\ \partial_t U_2(t,\tau) + \dfrac{1}{\varepsilon}\partial_\tau U_2(t,\tau) = \cos(\tau)\left[\mathcal{E}_{\rm ext}(\tau, U) + \mathcal{E}(t,\tau,U)\right], \ t>0, \tau\in\mathbb{T}, & \text{(3.1b)} \\ U_1(0,\tau) = \eta_1(\tau), \quad U_2(0,\tau) = \eta_2(\tau), \quad \tau\in\mathbb{T}, & \text{(3.1c)} \end{cases}$$

where $U = (U_1, U_2)^T$ and $\mathbb{T} = \mathbb{R}/(2\pi\mathbb{Z})$ is the torus. For readers' convenience, we write down the two functions $\mathcal{E}_{\rm ext}$ and $\mathcal{E}$ explicitly in the two-scale form as

$$\mathcal{E}_{\rm ext}(\tau, U) = a(\tau)\left[\cos(\tau)U_1 + \sin(\tau)U_2\right], \tag{3.2}$$

$$\mathcal{E}(t,\tau,U) = \frac{1}{r(\tau,U)}\int_0^{r(\tau,U)}\int_{\mathbb{R}} sg\left(t, \cos(\tau)s - \sin(\tau)v, \sin(\tau)s + \cos(\tau)v\right) dv ds,$$

with $r(\tau, U) = \cos(\tau)U_1 + \sin(\tau)U_2$. The above augmented problem is a system of one space dimensional partial differential equations with periodic boundary conditions in τ, where τ an additional independent variable and has no dependence on the time variable t any more. The problem (3.1) has been proved to be well-posed for all $t > 0$ (see [9, 10]). With initial data $\eta_1(\tau)$ and $\eta_2(\tau)$ satisfying at $\tau = 0$,

$$\eta_1(0) = \xi_1(0), \quad \eta_2(0) = \xi_2(0), \tag{3.3}$$

the solution of (3.1) coincides with the solution $\xi(t)$ of (2.4) when $\tau = t/\varepsilon$, i.e.

$$U_1(t, t/\varepsilon) = \xi_1(t), \quad U_2(t, t/\varepsilon) = \xi_2(t), \qquad t \geq 0.$$

Thus, in order to solve (2.4), one can solve (3.1) with initial data (η_1, η_2) satisfying (3.3). The benefit we can have from the two-scale formulation (3.1) is that the initial data (3.1c) is only prescribed at one point, i.e. $\tau = 0$ in (3.3), so there is some freedom to choose it. One of the main point of this approach is to determine a 'well-prepared' initial data in such a way the time derivatives of U are bounded

$$\partial_t^p U(t, \tau) = O(1), \quad 0 < \varepsilon \ll 1, \qquad \text{for some } p \geq 1. \tag{3.4}$$

3.2. Initial data. A systematical way of constructing the well-prepared initial data with arbitrary order of correction in general form has been given in [9] (see also [10]). Here we are going to use that strategy and construct the data up to the second order correction which will ensures that $\partial_t^p U(t, \tau), p = 1, 2$ are bounded. The construction of the well-prepared initial data relies on the asymptotic expansion of the solution U by means of a Chapman-Enskog expansion. To present the expansion, we rewrite (3.1a) and (3.1b) in a more compact way as

$$\partial_t U + \frac{1}{\varepsilon}\partial_\tau U = \mathcal{F}(t, \tau, U), \qquad t > 0,\ \tau \in \mathbb{T}, \tag{3.5}$$

where we denote

$$\mathcal{F}(t, \tau, U) = \Big(-\sin(\tau)\left(\mathcal{E}_{\text{ext}}(\tau, U) + \mathcal{E}(t, \tau, U)\right),\ \cos(\tau)\left(\mathcal{E}_{\text{ext}}(\tau, U) + \mathcal{E}(t, \tau, U)\right) \Big)^T. \tag{3.6}$$

For some general periodic function $u(\tau)$ on $\mathbb{T}$, we introduce two operators

$$Lu(\tau) := \partial_\tau u(\tau), \qquad \Pi u := \frac{1}{2\pi}\int_0^{2\pi} u(\tau)\mathrm{d}\tau. \tag{3.7}$$

Actually, Π corresponds to the orthogonal projection in L^2 onto $\operatorname{Ker} L$ (which is the set of constant functions with respect to τ). It is easy to see that L is invertible on the set of periodic functions with zero average (i.e. $\Pi u = 0$), and its pseudo-inverse writes

$$L^{-1}u = (I - \Pi)\int_0^\tau u(\theta)\mathrm{d}\theta.$$

Then the Chapman-Enskog expansion of the solution reads

$$U(t, \tau) = \underline{U}(t) + h(t, \tau), \quad \text{with} \quad \underline{U}(t) = \Pi\left(U(t, \tau)\right). \tag{3.8}$$

Inserting this expansion into (3.5) leads to the following micro-macro model (see [5, 23])

$$\partial_t \underline{U} = \Pi\left(\mathcal{F}(t,\tau,\underline{U}+h)\right), \quad \partial_t h + \frac{1}{\varepsilon} L h = (I-\Pi)\left(\mathcal{F}(t,\tau,\underline{U}+h)\right).$$

The second (micro) equation gives

$$h = \varepsilon A\left(\mathcal{F}(t,\tau,\underline{U}+h)\right) - \varepsilon L^{-1}(\partial_t h), \quad A := L^{-1}(I-\Pi). \tag{3.9}$$

By assuming $\partial_t h = O(1)$, we can see $h = O(\varepsilon)$. If we further assume $\partial_t^2 h = O(1)$, we get by deriving in time (3.9)

$$\partial_t h = \varepsilon A\left[\partial_t \mathcal{F}(t,\tau,\underline{U}+h) + \partial_U \mathcal{F}(t,\tau,\underline{U}+h)(\partial_t \underline{U} + \partial_t h)\right] - \varepsilon L^{-1}(\partial_t^2 h) = O(\varepsilon),$$

so that we have

$$h(t,\tau) = h_1(t,\tau,\underline{U}) + O(\varepsilon^2), \quad \text{with} \quad h_1(t,\tau,U) := \varepsilon A\left(\mathcal{F}(t,\tau,U)\right). \tag{3.10}$$

If we continue to assume that $\partial_t^3 h = O(1)$, similarly as above, we would expect to have $\partial_t^2 h = O(\varepsilon)$. As a consequence, we can have the asymptotic expansion to the next order as

$$h(t,\tau) = \varepsilon A\left(\mathcal{F}(t,\tau,\underline{U}+h_1)\right) - \varepsilon^2 A^2\left[\partial_t \mathcal{F}(t,\tau,\underline{U}) + \partial_U \mathcal{F}(t,\tau,\underline{U})\Pi\left(\mathcal{F}(t,\tau,\underline{U})\right)\right] + O(\varepsilon^3). \tag{3.11}$$

Now let $t = 0$ in (3.8), we have $U(0,\tau) = \underline{U}(0) + h(0,\tau) = \eta(\tau)$ and evaluating this last equality at $\tau = 0$ leads to $U(0,0) = \underline{U}(0) + h(0,0) = \eta(0)$. Since $\eta(0) = \xi(0)$, we deduce

$$\eta(\tau) = \xi(0) + h(0,\tau) - h(0,0), \qquad \tau \in \mathbb{T}. \tag{3.12}$$

Thus, an initial data up to the first order correction (resp. the second order correction) in ε can be obtained by inserting (3.10) (resp. (3.11)) with $t = 0$ into (3.12). By noting the fact that $\underline{U}(0) = \xi(0) + O(\varepsilon)$ (since $\underline{U}(0) + h(0,0) = \eta(0)$ and $h = O(\varepsilon)$), the initial data with first order correction referred as the *first order initial data* in the following, can be given as

$$\eta^{1st}(\tau) = \xi(0) + h_1(0,\tau,\xi(0)) - h_1(0,0,\xi(0)), \tag{3.13}$$

with h_1 given in (3.10). To construct the second order correction, we further note that $\underline{U}(0) = \xi(0) - h_1(0,0,\xi(0)) + O(\varepsilon^2)$ (indeed, with (3.8) and (3.10) with $t = \tau = 0$, we get $\underline{U}(0) + h_1(0,0,\underline{U}(0)) + O(\varepsilon^2) = \eta(0)$ and we use $\underline{U}(0) = \xi(0) + O(\varepsilon)$). Then the initial data with second order correction referred as the *second order initial data* in the following, can be given as

$$\eta^{2nd}(\tau) = \xi(0) + h_1\left(0,\tau,\eta^{1st}(\tau)\right) - h_1\left(0,0,\eta^{1st}(\tau)\right) + h_2(0,\tau,\xi(0)) - h_2(0,0,\xi(0)), \tag{3.14}$$

where

$$h_2(t,\tau,U) := -\varepsilon^2 A^2\left[\partial_t \mathcal{F}(t,\tau,U) + \partial_U \mathcal{F}(t,\tau,U)\Pi\left(\mathcal{F}(t,\tau,U)\right)\right].$$

It is directly from (3.6) and (3.2) to get

$$\partial_t \mathcal{F}(t,\tau,U) = \partial_t \mathcal{E}(t,\tau,U) \cdot \Big(-\sin(\tau),\ \cos(\tau) \Big)^T,$$

$$\partial_U \mathcal{F}(t,\tau,U) = \left[a(\tau) + G\left(t,\tau,r(\tau,U)\right)\right] \cdot \begin{pmatrix} -\sin(\tau)\cos(\tau) & -\sin^2(\tau) \\ \cos^2(\tau) & \sin(\tau)\cos(\tau) \end{pmatrix},$$

with $r(\tau, U) = \cos(\tau)U_1 + \sin(\tau)U_2$, and

$$G(t,\tau,r) := -\frac{1}{r^2}\int_0^r \int_{\mathbb{R}} sg\left(t, \cos(\tau)s - \sin(\tau)v, \sin(\tau)s + \cos(\tau)v\right) \mathrm{d}v\mathrm{d}s + \int_{\mathbb{R}} g(t, \cos(\tau)r - \sin(\tau)v, \sin(\tau)r + \cos(\tau)v)\mathrm{d}v. \tag{3.15}$$

The nonlinear terms there like $\mathcal{F}, \partial_t\mathcal{F}, \partial_U\mathcal{F}$ essentially depend on the information of $g(0,\xi)$ or correspondingly $g(t_n,\xi)$ in (2.7) which is already known. Thus, the constructed initial data with ε correction up to the first or second order are given explicitly. The above computations already show that (3.10) and (3.11) are necessary conditions to guarantee $\partial_t^2 h = O(1)$ and $\partial_t^3 h = O(1)$ as $\varepsilon \to 0$, respectively. Indeed, they have also been shown rigorously in [9] to be sufficient for providing $\partial_t^2 U = O(1)$ and $\partial_t^3 U = O(1)$ as $\varepsilon \to 0$, respectively, which is enough for us to construct a first order or second order UA numerical integrator for (3.1).

Before presenting the numerical schemes, we briefly describe the evaluation of the nonlinear terms $\mathcal{E}(t,\tau,U)$, $\partial_t\mathcal{E}(t,\tau,U)$ and $G(t,\tau,r)$. To compute $\mathcal{E}(t,\tau,U)$, we define two auxiliary functions

$$\begin{aligned} &\mathcal{E}_g(t,\tau,r) := \frac{1}{r}\int_0^r \int_{\mathbb{R}} sw(t,\tau,s,v)\mathrm{d}v\mathrm{d}s, \\ &w(t,\tau,s,v) := g\left(t, \cos(\tau)s - \sin(\tau)v, \sin(\tau)s + \cos(\tau)v\right). \end{aligned} \tag{3.16}$$

Then

$$\frac{1}{r}\partial_r\left(r\mathcal{E}_g(t,\tau,r)\right) = \int_{\mathbb{R}} w(t,\tau,r,v)\mathrm{d}v. \tag{3.17}$$

With the truncations in (2.5) and $g(t,\xi)$ known on mesh grids $(\xi_{1,k}, \xi_{2,j}), (k,j) \in \mathcal{T}$, we use a cubic spline interpolation to find the function value of g on the points $(\cos(\tau)s - \sin(\tau)v,\ \sin(\tau)s + \cos(\tau)v)^T$, where (s,v) is on the uniform mesh grids of $[-L,L]^2$. Let us remark that the spline coefficients have already been computed in (2.6). So we can have $w(t,\tau,r,v)$ on the mesh grids simply by evaluation. Its integration on $\mathbb{R}$ can be reduced to the finite interval $[-L,L]$ and approximated by Fourier spectral method. Then we solve the Poisson type equation (3.17) by means of the Fourier spectral method with noticing $r\mathcal{E}_g$ should vanish at boundaries. Once $\mathcal{E}_g(t,\tau,r)$ is obtained for r in $[-L,L]$ on grid points, we use the cubic spline interpolation again to compute $\mathcal{E}_g(t,\tau,\cos(\tau)U_1 + \sin(\tau)U_2) = \mathcal{E}(t,\tau,U)$.

For $\partial_t\mathcal{E}$, we directly write down

$$\partial_t\mathcal{E}(t,\tau,U) = \frac{1}{r(\tau,U)}\int_0^{r(\tau,U)} \int_{\mathbb{R}} s\partial_t g\left(t, \cos(\tau)s - \sin(\tau)v, \sin(\tau)s + \cos(\tau)v\right) \mathrm{d}v\mathrm{d}s.$$

The time derivative $\partial_t g\left(t,\xi_1,\xi_2\right)$ can be given by (2.5a), i.e.

$$\partial_t g(t,\xi) = -\left[\mathcal{E}_{\mathrm{ext}}(t/\varepsilon,\xi) + \mathcal{E}(t,t/\varepsilon,\xi)\right] \cdot \left[-\sin\left(t/\varepsilon\right)\partial_{\xi_1} g(t,\xi) + \cos\left(t/\varepsilon\right)\partial_{\xi_2} g(t,\xi)\right].$$

The above spatial derivatives can be computed by Fourier spectral method. Then the rest can be done by similarly as in (3.17) with $\mathcal{E}_g$ replaced by $\mathcal{E}_{\partial_t g}$. As for $G(t,\tau,r)$, it is enough to notice with the auxiliary function (3.16), (3.15) becomes

$$G(t,\tau,r) = \int_{\mathbb{R}} w(t,\tau,r,v)\mathrm{d}v - \frac{1}{r}\mathcal{E}_g(t,\tau,r).$$

3.3. Numerical schemes. Now we give time discretizations for (3.1). We still use (3.5) to present the schemes. In the τ-direction, thanks to the periodicity, one will use FFT. In details, choose an even integer $N_\tau > 0$, denote $\Delta\tau = 2\pi/N_\tau$, $\tau_j = j\Delta\tau$, $j = 0, 1, \ldots, N_\tau$, the discrete Fourier transform of $(V_j)_{j=0,\ldots,N_\tau-1}$ is given by

$$\widetilde{V}_\ell = \sum_{j=0}^{N_\tau-1} V_j \mathrm{e}^{-i\ell\tau_j}, \quad \ell = -N_\tau/2, \ldots, N_\tau/2 - 1.$$

First order scheme. Let denote $U^n(\tau) \approx U(t_n, \tau)$ and choose $U^0(\tau) = \eta^{1st}(\tau)$ where η^{1st} is given by (3.13). Then the first order scheme writes for $n \geq 0$,

$$U^{n+1}(\tau) = U^n(\tau) + \Delta t \mathcal{F}\left(t_n, \tau, U^n(\tau)\right) - \frac{\Delta t}{\varepsilon}\partial_\tau U^{n+1}(\tau), \qquad \tau \in \mathbb{T}, \tag{3.18}$$

and Fourier transform in τ provides the following update of U^{n+1}

$$\widetilde{U}_\ell^{n+1} = \frac{\widetilde{U}_\ell^n + \Delta t \widetilde{\mathcal{F}}_\ell^n}{1 + i\ell\Delta t/\varepsilon}, \qquad \ell = -N_\tau/2, \ldots, N_\tau/2 - 1,$$

where $\widetilde{\mathcal{F}}_\ell^n$ the Fourier transform of $\mathcal{F}(t_n, \tau, U^n(\tau))$.

Second order scheme. Still denoting $U^n(\tau) \approx U(t_n, \tau)$ and choose $U^0(\tau) = \eta^{2nd}(\tau)$ where η^{2nd} is given by (3.14), then the second order scheme writes for $n \geq 0$,

$$\begin{cases} U^{n+1/2}(\tau) = U^n(\tau) + \dfrac{\Delta t}{2}\mathcal{F}\left(t_n, \tau, U^n(\tau)\right) - \dfrac{\Delta t}{2\varepsilon}\partial_\tau U^{n+1/2}(\tau), \quad \tau \in \mathbb{T}, \\ U^{n+1}(\tau) = U^n(\tau) + \Delta t \mathcal{F}\left(t_{n+1/2}, \tau, U^{n+1/2}(\tau)\right) - \dfrac{\Delta t}{2\varepsilon}\partial_\tau\left(U^{n+1}(\tau) + U^n(\tau)\right), \end{cases} \tag{3.19}$$

which gives, using Fourier in τ, for $\ell = -N_\tau/2, \ldots, N_\tau/2 - 1$,

$$\widetilde{U}_\ell^{n+1} = \frac{(2 - i\ell\Delta t/\varepsilon)\widetilde{U}_\ell^n + 2\Delta t\widetilde{\mathcal{F}}_\ell^{n+1/2}}{2 + i\ell\Delta t/\varepsilon}, \quad \widetilde{U}_\ell^{n+1/2} = \frac{2\widetilde{U}_\ell^n + \Delta t\widetilde{\mathcal{F}}_\ell^n}{2 + i\ell\Delta t/\varepsilon},$$

where $\widetilde{\mathcal{F}}_\ell^{n+1/2}$ the Fourier transform of $\mathcal{F}(t_{n+1/2}, \tau, U^{n+1/2}(\tau))$.

Algorithm. We end this section by writing the full algorithm to solve the original Vlasov-Poisson (2.5). Starting from $g_{k,j}^n$ and its cubic spline coefficients $\omega_{k,j}^n$, we compute $g_{k,j}^{n+1}$ thanks to

- compute $\mathcal{E}$ with (3.2)
- compute $\xi^n(\Delta t)$ by solving (2.7) by
  - compute the initial condition (3.13) or (3.14).
  - compute $U^{n+1}(\tau)$ using (3.18) or (3.19)
  - compute $\xi^n(\Delta t) = U^{n+1}(\Delta t/\varepsilon)$ using a trigonometric interpolation
- compute $g_{k,j}^{n+1}$ using (2.8) and its cubic spline coefficients $\omega_{k,j}^{n+1}$.

The original solution f can be recovered by considering the inverse change of variables of (2.1). This also requires an interpolation which is done using cubic spline. The resulting numerical scheme is free some any CFL condition and the temporal discretization error is $O(\Delta t)$ for (3.13)-(3.18) and $O(\Delta t^2)$ for (3.14)-(3.19), uniformly in ε.

After we solve (3.1) via (3.18) or (3.19) till $t = T_{final} = n_f\Delta t$, we take $U^{n_f}(T_{final}/\varepsilon) \approx \xi(T_{final})$. The above two schemes (3.18) and (3.19) associated with

the constructed initial data have been proved in general form [9] to offer uniform accuracy in time for finite time computing, i.e. the temporal discretization error is $O(\Delta t)$ for (3.18) and $O(\Delta t^2)$ for (3.19) for $0 \le t \le T_{final}$. The spatial error, due to the Fourier spectral discretization, would be of spectral accuracy since the solution of (3.1) is smooth in τ if the solution of the Vlasov-Poisson equation (1.1a) is smooth enough. Thanks to the implicit treatment of the transport operator $\partial_t + \frac{1}{\varepsilon}\partial_\tau$ in (3.5), there is no CFL condition on Δt and $\Delta\tau$ or restriction on ε for stability.

To complete the numerical method for the Vlasov-Poisson equation (2.5), we embed the proposed UA integrators into the forward semi-Lagrangian method for solving (2.7) for t from 0 to Δt, i.e. solve (2.7) for one time step with $T_{final} = \Delta t$. The construction of the well-prepared initial data is performed at each time level t_n. At each time level, the characteristic equation at every mesh grid point is independent from each other, as we mentioned before. Thus, it is quite natural to program with parallel computing and the method would be very efficient.

4. Numerical results. In this section, we present numerical results of testing the proposed first order and second order UA schemes. We always choose the tension function as $a(t) = \cos(2t)^2$, which is the same as that in [11], unless specified.

We are interested in the performance of the present method which combines the forward semi-Lagrangian framework with a two-scale treatment of the characteristics (FSL-UA). We also compare FSL-UA with the Asymptotic Preserving Finite Difference (APFD) method proposed in [11] and with a standard method based on a time splitting-Fourier method (TSFP) (see [11] for more details). Finally, we also perform comparisons with the asymptotic model; when $\varepsilon \to 0$, the solution $g(t,\xi)$ of the Vlasov-Poisson equation in the rotating frame (2.2) converges to $2\pi G(t,\xi)$, where $G(t,\xi)$ solves

$$\begin{aligned}\partial_t G(t,\xi) - \Pi\left[\sin(\tau)\left(\mathcal{E}_{\text{ext}}(\tau,\xi) + 2\pi\mathcal{E}_0(t,\tau,\xi)\right)\right]\partial_{\xi_1} G(t,\xi) \\ + \Pi\left[\cos(\tau)\left(\mathcal{E}_{\text{ext}}(\tau,\xi) + 2\pi\mathcal{E}_0(t,\tau,\xi)\right)\right]\partial_{\xi_2} G(t,\xi) = 0, \quad \xi \in \mathbb{R}^2,\ t > 0,\end{aligned} \tag{4.1a}$$

$$G(0,\xi) = \frac{1}{2\pi} f_0(\xi), \tag{4.1b}$$

where Π is defined in (3.7) and

$$\mathcal{E}_0(t,\tau,\xi) = \frac{1}{r(\tau,\xi)} \int_0^{r(\tau,\xi)} \int_{\mathbb{R}} sG\left(t, \cos(\tau)s - \sin(\tau)v, \sin(\tau)s + \cos(\tau)v\right) \mathrm{d}v\mathrm{d}s,$$

with $r(\tau,\xi) := \cos(\tau)\xi_1 + \sin(\tau)\xi_2$. With the inverse transform of (2.1), we also have an approximation of the solution $f(t,r,v)$ of the original Vlasov-Poisson equation (1.1)

$$F(t,\tau,r,v) := 2\pi G(t, \cos(\tau)r - \sin(\tau)v, \sin(\tau)r + \cos(\tau)v).$$

When $\varepsilon \to 0$, f is said to two-scale converge to F (see [1, 19, 25]). In the following, we refer the above problem as the "limit model". We can also solve it by a very similar forward semi-Lagrangian method with a Leap-Frog finite difference method for the characteristics.

Before looking at the numerical results, we compare the computation complexity of FSL-UA with APFD proposed in [11] and TSFP. A detailed comparison is given in Table 4.1 which resumes the advantages of the present method. In the following, some numerical results obtained by FSL-UA are given to illustrate the efficiency of the approach.

Table 4.1: Complexity comparisons between the FSL-UA schemes, the APFD schemes and the TSFP method. Here N_τ is the number of points in τ; N_{tol} is in total number of mesh grids in 2D plane; N_r is the number of points in r-direction and suppose it is larger than or equal to that in v-direction.

	Memory	Computational cost (per step)	Uniform	CFL condition
FSL-UA	N_{tol}	$N_\tau N_{tol}\log(N_\tau)$	Yes	No
APFD	$N_{tol}N_\tau$	$N_\tau N_{tol}\log(N_\tau)$	Yes	Yes
TSFP	N_{tol}	$N_r\log(N_r)$	No	No

4.1. Example 1. In our first example, we choose a symmetric initial data in (r,v)-domain for (1.1d) as

$$f_0(r,v) = \mathrm{e}^{-2(r^2+v^2)}, \qquad (r,v)\in\Omega=[-3,3]^2. \tag{4.2}$$

The reference solution of the Vlasov-Poisson equation (1.1a) at $t = 0.4$ is obtained by the TSFP with very small step size, e.g. $\Delta t = 10^{-8}$, $\Delta r = \Delta v = 1/32$. We test the proposed FSL-UA method. In particular, we look at the influence of the initial condition for the characteristics. To do so, the maximum error (between FSL-UA and the reference TSFP) is computed at time $t = 0.4$ with $N = N_1 = N_2 = 800$ and $N_\tau = 64$, for different Δt and different ε.

Considering the first order scheme (3.18), the influence of the correction of the initial data is studied in Fig. 4.1, namely $\eta(\tau) = \xi(0)$ (uncorrected initial data) and $\eta(\tau) = \eta^{1st}(\tau)$ given by (3.13). The results obtained using (3.13) are clearly better since a uniform first order accuracy is observed.

The numerical results indicate a first order UA convergence. Comparing the results obtained with the first and the uncorrected initial data, the results given by the first order initial data case are clearly better in the error.

Considering now the second order scheme (3.19), we consider three types of initial data:
- the *uncorrected initial data* $\eta(\tau) = \xi(0)$,
- the *first order initial data* $\eta(\tau) = \eta^{1st}(\tau)$ given by (3.13),
- the *second order initial data* $\eta(\tau) = \eta^{2nd}(\tau)$ given by (3.14).

The corresponding results are shown in Fig. 4.2. The results illustrate the uniform second order convergence when enough corrections are considered, while without enough corrections, the method is not uniformly accurate, as expected. Fig. 4.2 also clearly shows that there is no CFL restrictions on the time step.

We now are interested in the spatial error. We fix $\Delta t = 2.5\times10^{-3}$ (the final time is $t = 0.4$) and consider different mesh grid $\Delta\xi$ and ε ($N_\tau = 64$). The obtained results are given in Fig. 4.3. A spatial order between 2 and 3 is obtained which is in accordance with results obtained in [24].

Before we move on to the next numerical example, we would like to show numerically the reason why we work with the Vlasov-Poisson equation in the rotating frame (2.2) rather than the original (1.1). In fact for the original equation (1.1), it is more straightforward to solve it by means of the forward semi-Lagrangian method, and the

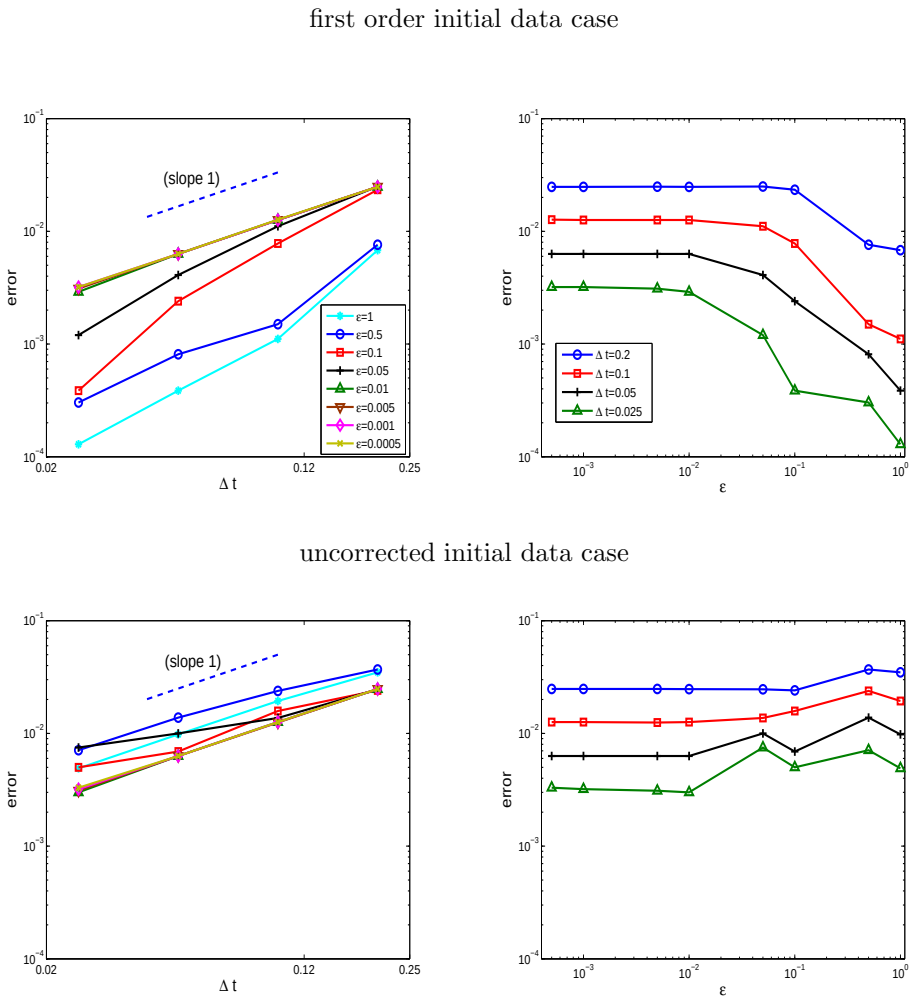


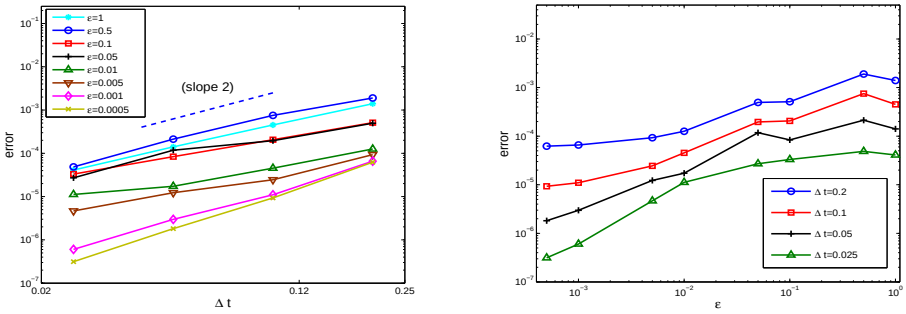
Fig. 4.1: Dependence of the temporal error in maximum norm on Δt (left panel) and on ε (right panel) at $t = 0.4$ for the 2nd order scheme with the three types of initial data in example (4.2): first order initial data (top row); uncorrected initial data (bottom row).

UA schemes can be derived similarly for the corresponding characteristic equations

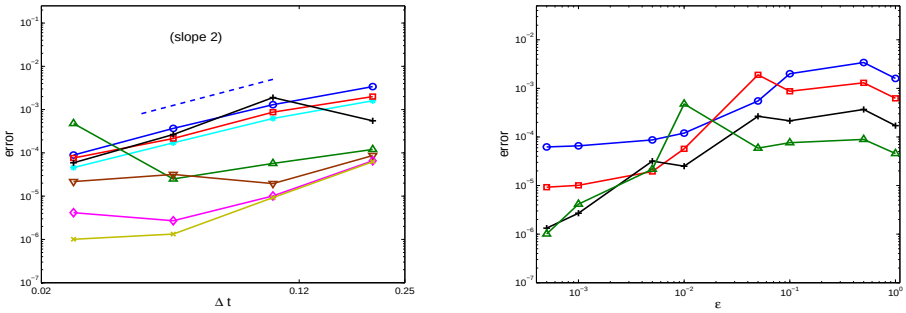
$$\begin{cases} \dot{r}(t) = \dfrac{v(t)}{\varepsilon}, \\ \dot{v}(t) = E(t,r) - \dfrac{r(t)}{\varepsilon} + a\left(\dfrac{t}{\varepsilon}\right) r(t). \end{cases} \tag{4.3}$$

However, as we mentioned in Section 2, the forward semi-Lagrangian method would be inefficient in the spatial approximation when the displacement of the mesh grid is too large. Since ε in (4.3) is proportional to the rotating speed, so in the highly oscillatory regime $0 < \varepsilon \ll 1$, the flow (4.3) would cause large displacements in mesh

second order initial data case



first order initial data case



uncorrected initial data case

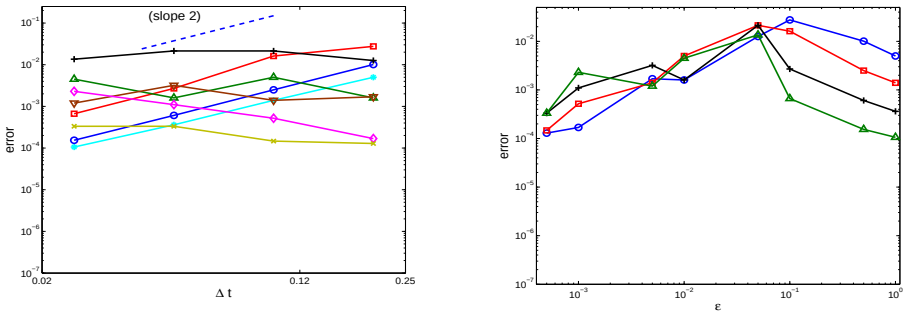


Fig. 4.2: Dependence of the temporal error in maximum norm on Δt (left panel) and on ε (right panel) at $t = 0.4$ for the 2nd order scheme with the three types of initial data in example (4.2): second order initial data (top row); first order initial data (middle row); uncorrected initial data (bottom row).

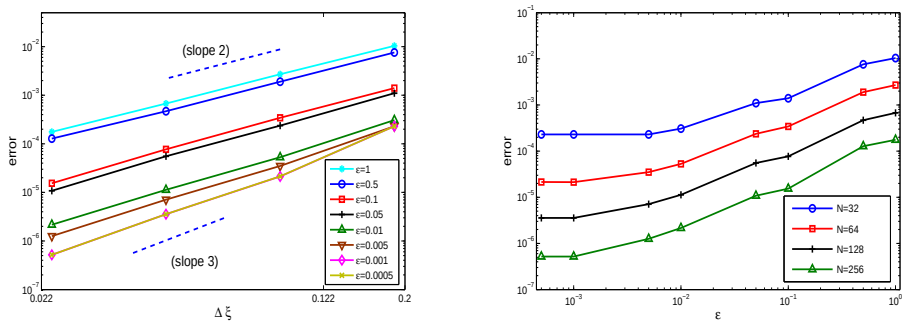


Fig. 4.3: Dependence of the spatial error in maximum norm on $\Delta\xi = \Delta\xi_1 = \Delta\xi_2$ ($N = N_1 = N_2$) (left panel) and on ε (right panel) at $t = 0.4$ for the FSL-UA scheme in example (4.2).

Table 4.2: Spatial error of the linear Vlasov-Poisson equation in (r, v)-domain at $t = 0.4$ with different mesh size $N = N_1 = N_2$ and $\Delta t = 0.01$ in example (4.2): the maximum error by working with the *original* frame (1.1) and with the *rotating* frame (2.2).

original	$N = 32$	$N = 64$	$N = 128$	$N = 256$
$\varepsilon = 0.5$	4.80E-3	1.20E-3	3.13E-4	7.80E-5
$\varepsilon = 0.1$	2.30E-3	4.73E-3	8.79E-4	1.98E-5
$\varepsilon = 0.05$	5.00E-3	1.10E-3	7.33E-4	6.44E-5
$\varepsilon = 0.01$	1.13E-1	1.01E-1	9.79E-2	9.70E-2
rotating	$N = 32$	$N = 64$	$N = 128$	$N = 256$
$\varepsilon = 0.5$	4.90E-3	1.20E-3	3.10E-4	7.74E-5
$\varepsilon = 0.1$	1.30E-3	3.09E-3	7.65E-5	1.90E-5
$\varepsilon = 0.05$	6.00E-4	1.41E-4	3.49E-5	8.66E-6
$\varepsilon = 0.01$	1.81E-4	3.10E-5	7.05E-6	1.72E-6

grids, which in turn would constrain the time step with ε if one wants to reduce the displacement. To illustrate this issue, we even drop the nonlinearity and consider the linear Vlasov-Poisson equation, i.e. let $E \equiv 0$ in (1.1).

Considering (4.2) as initial data, Table 4.2 shows the failure in the spatial error by solving the original equation (1.1) in the highly oscillatory regime, where we use the second order UA scheme with a fixed time step $\Delta t = 0.01$ as the temporal integrator, and the comparison with results in solving the rotating equation (2.2) shows the importance of going to the rotating frame.

4.2. Example 2. Our next numerical example begins with a non-symmetric initial data in (1.1d) as

$$f_0(r, v) = \frac{2}{\sqrt{0.4\pi}} \left[\mathrm{erf}\left(\frac{r + 1.2}{0.3}\right) - \mathrm{erf}\left(\frac{r - 1.2}{0.3}\right) \right] \mathrm{e}^{-2.5v^2}, \qquad (r, v) \in \Omega = [-4, 4]^2. \tag{4.4}$$

Table 4.3: Accuracy comparison of the second order FSL-UA and APFD: maximum error in (r, v)-domain at $t = \pi/16$ with $N_1 = N_2 = N$ for different ε in example (4.4).

APFD	$\Delta t = t/4, N = 2^6$	$\Delta t = t/8, N = 2^7$	$\Delta t = t/16, N = 2^8$
$\varepsilon = 0.5$	1.08E-1	3.44E-2	8.39E-3
$\varepsilon = 0.05$	6.15E-2	2.15E-2	5.75E-3
$\varepsilon = 0.005$	9.45E-2	2.73E-2	6.97E-3
$\varepsilon = 0.0005$	4.68E-2	1.42E-2	3.36E-3
FSL-UA	$\Delta t = t/4, N = 2^6$	$\Delta t = t/8, N = 2^7$	$\Delta t = t/16, N = 2^8$
$\varepsilon = 0.5$	2.57E-2	6.70E-3	1.80E-3
$\varepsilon = 0.05$	1.49E-2	3.20E-3	9.33E-4
$\varepsilon = 0.005$	7.70E-3	2.50E-3	1.40E-3
$\varepsilon = 0.0005$	5.70E-3	1.40E-3	4.09E-4

We first compare the accuracy of the FSL-UA method and the APFD method. In Table 4.3, we list the maximum error (TSFP is the reference solution) of the numerical methods at $t = \pi/16$ for different numerical parameters. Clearly, FSL-UA produces more accurate approximation than APFD under the same step size (around one order of magnitude better). Let us remark however that FSL-UA uses the second order initial data, while only first order initial data is considered in [11] for APFD.

We now look at the time evolution of the Root Mean Square (RMS) quantity associated to the filtered distribution function $g(t, \xi)$:

$$RMS(t) := \sqrt{\int_{\mathbb{R}^2} \xi_1^2 g(t, \xi_1, \xi_2) \mathrm{d}\xi}, \qquad t \geq 0.$$

Since the main oscillation has been filtered, this quantity contains oscillations of magnitude ε. Hence, the oscillations disappears in the limit $\varepsilon = 0$. We apply the FSL-UA method with the second order UA scheme; we then compare the second order initial data (3.14) and the uncorrected initial data $\eta(\tau) = \xi(0)$, but also with a reference solution given by TSFP and the limit model. Fig. 4.4 shows the dynamics of $RMS(t)$ with $\varepsilon = 0.05$ and $\varepsilon = 0.01$ with time step $\Delta t = 0.05$ and mesh size $N = 128$. We can observe the amplitude of the oscillations reduces to zero as $\varepsilon \to 0$, but the frequency of the oscillations increases dramatically. The numerical solution given by the FSL-UA scheme fits very well with the exact solution, even though the time step is not small enough to resolve the oscillations. As observed previously, the one with corrected data is more accurate than the uncorrected data case. As expected, the limit model is not able to capture the oscillations since it only follow the averaged dynamics by construction.

In Fig. 4.5, we plot the difference in maximum error between the limit model (4.1) and FSL-UA for different values of ε, at $t = 1.5$. As observed in [11], the error is $O(\varepsilon)$, illustrating the domain of validity of the asymptotic model.

Finally, the long time behavior of the solution for $\varepsilon = 0.001$ is shown in Fig. 4.6, by applying FSL-UA (with second order initial data (3.14)) with $\Delta t = 0.1,\ N = 512$. The results indicate that the FSL-UA scheme performs very well in the long time run. Compared to the results in [11], the filamentations in Fig. 4.6 do not have the spurious structures in the center of the beam, and the FSL-UA method does not have significant numerical diffusion error in the long time run.

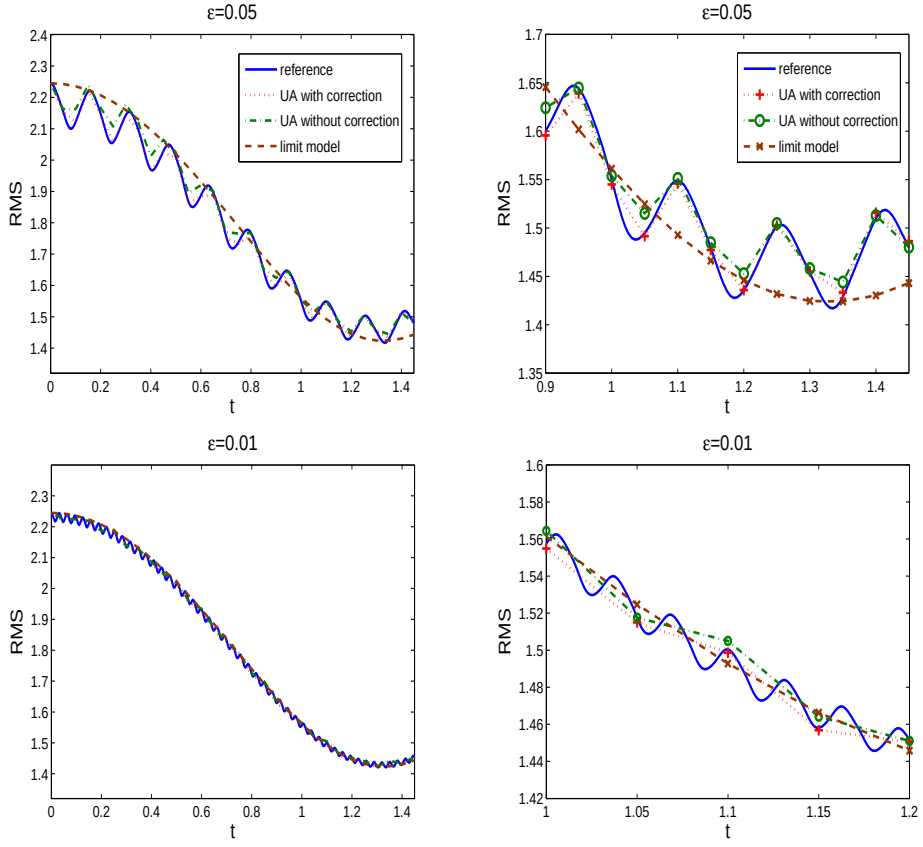


Fig. 4.4: Time history of $RMS(t)$ for $\varepsilon = 0.05$ and $\varepsilon = 0.01$ in example (4.4), computed by the limit model and the second order UA scheme with second order correction and without correction. On the right: zoom of the left figure.

4.3. Example 3. In our last example, we apply the proposed method to study dynamics in the Vlasov-Poisson equation with a discontinuous initial data (1.1d)

$$f_0(r,v) = \frac{n_0}{\sqrt{2\pi}v_0}\exp\left(-\frac{v^2}{2v_0^2}\right)\chi_{[-r_m,r_m]}(r), \quad (r,v)\in[-3,3]^2, \tag{4.5}$$

with $\chi_{[-r_m,r_m]}(r) = 1$ if $|r| \le r_m$ and 0 otherwise. This type of initial data comes from a semi-Gaussian beam in particle accelerator physics (see [19, 25]).

Case I:

Firstly, we take the tension function $a(t) = \cos^2(t)$ and choose $v_0 = 0.1, n_0 = 4, r_m = 1.85$, which has been considered in [25]. The problem is solved by the second order FSL-UA method with mesh size $N = 256, N_\tau = 32, \Delta t \approx 1\times 10^{-2}$. We also solve the limit model (4.1) with fine mesh $N = 256, \Delta t \approx 1 \times 10^{-2}$ and solve the original Vlasov-Poisson equation (1.1) by a classical backward semi-Lagrangian method [25] with the same N but ε-dependent small Δt as comparisons. The results for $\varepsilon = 0.05$ (resp. $\varepsilon = 0.01$) are shown in Fig. 4.7 (resp. Fig. 4.8) where the solutions of classical

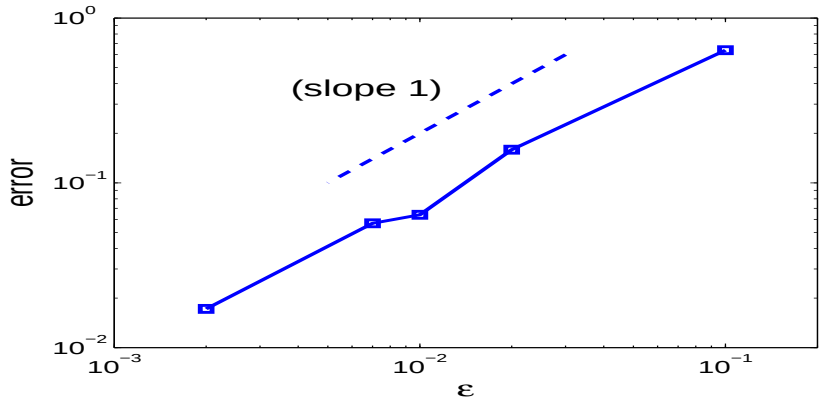


Fig. 4.5: The maximum error between FSL-UA and the limit model (4.1) at $t = 1.5$.

method are computed with $\Delta t \approx 2 \times 10^{-3}$ (resp. $\Delta t \approx 1 \times 10^{-5}$). First, we remark that the limit model gives rise to uncorrect results in the intermediate regime (when $\varepsilon = 0.05$). Second, when the asymptotic regime is considered ($\varepsilon = 0.01$), as pointed out in [25], the classical method suffers from numerical diffusion problems in a long time simulation which makes very difficult to provide accurate long time simulations. On the contrary, FSL-UA produces very good approximation for both values of ε. This is confirmed by Fig. 4.9 in which the results obtained by FSL-UA with refined numerical parameters are displayed ($\Delta t \approx 5 \times 10^{-3}$ and $N = 512$). The results are in very good agreement with the previous ones obtained with coarser numerical parameters.

Case II: Secondly, we take the tension function $a(t) = \cos^2(2t)$ and choose $v_0 = 0.1, n_0 = 4, r_m = 0.75$ to reproduce the results in [25] and show the computational efficiency. The numerical parameters are the same as in [25], that is $\Delta t \approx 7.392 \times 10^{-3}, N = 128$ and $N_\tau = 16$. The results are displayed in Fig. 4.10 and are in a very good agreement with the results shown in [25]. Moreover, with $t = 6.9864$, this run takes 8.0395 minutes (the program is written in a sequential way by Fortran and is run on an iMac with processor 2.5Ghz Intel Core i5).

Case III:

Thirdly, we take the tension function $a(t) = 0$ and choose $v_0 = 0.07275, n_0 = 0.3, r_m = 0.75$, which makes the initial beam stiffer than the previous two cases as considered in [19]. We apply the second order FSL-UA method to study long time dynamics of the solution. The problem is solved with mesh size $N = 512, N_\tau = 32$ and $\Delta t = 0.5$ or $\Delta t = 0.1$. The results for $\varepsilon = 0.01$ at different time are shown in Fig. 4.11. Even if the time step is two orders of magnitude larger than the period of the oscillations, the method is able to reproduce in a very good way the filamentation phenomena.

5. Conclusions. In this work, the two-scale formulation proposed in [11, 9] is adapted to the forward semi-Lagrangian framework. The resulting numerical scheme turns out to be very performant regarding the Vlasov-Poisson problem: it enjoys a Uniform Accuracy property with respect to ε thanks to a suitable choice of initial data, and it is free from any CFL condition. An extension of this approach to PIC method would enable to tackle high dimensional problems, like those encountered in

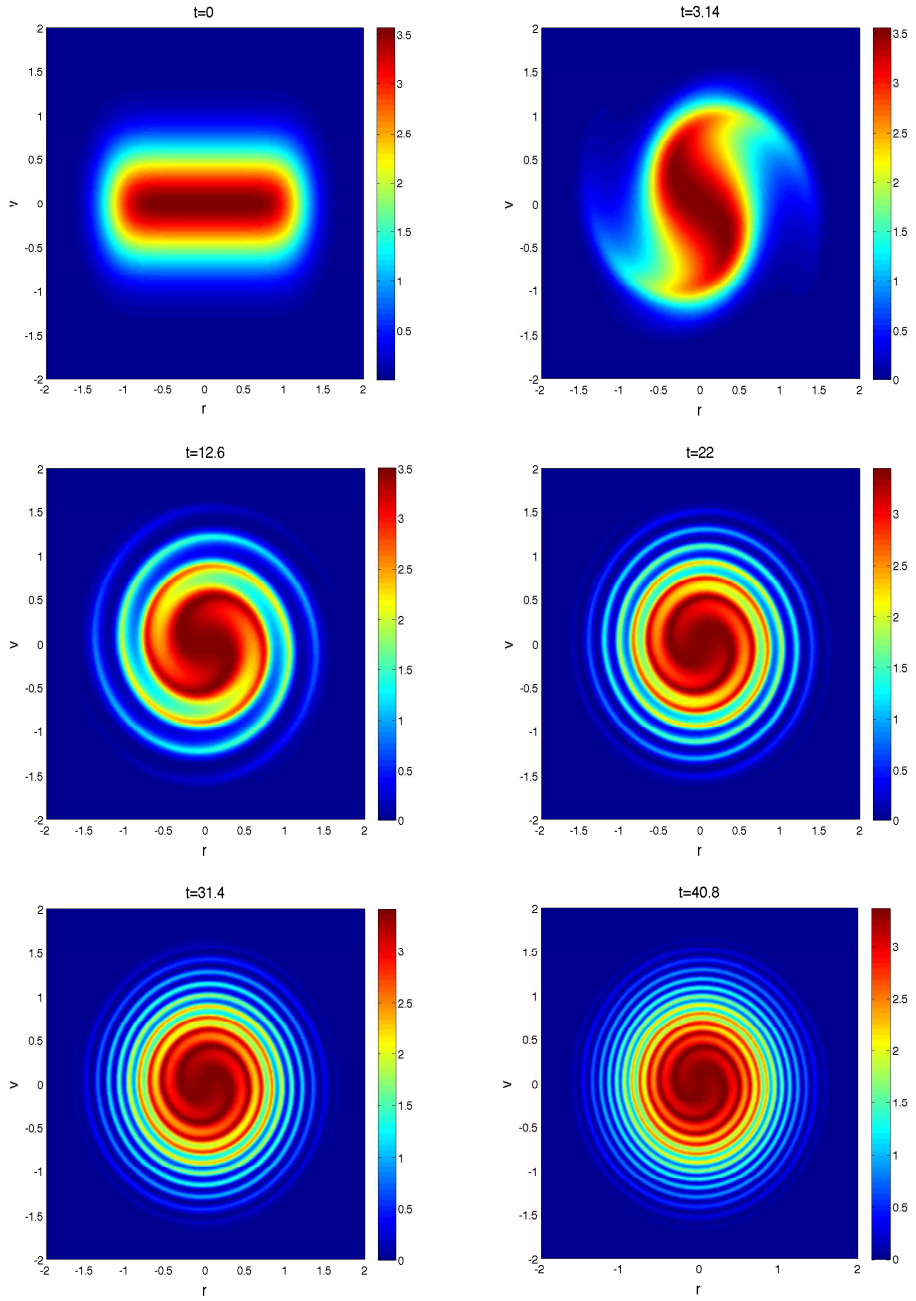


Fig. 4.6: Contour plot of the solution $f(t, r, v)$ with the second order FSL-UA scheme for $\varepsilon = 0.001$ at different time: $t = 0$, $t = 3.14$, $t = 12.6$, $t = 22$, $t = 31.4$ and $t = 40.8$.

strongly magnetized plasmas (see [16]).

Acknowledgements. This work was supported by the french ANR project MOONRISE ANR-14-CE23-0007-01. N. Crouseilles and M. Lemou are supported by the Enabling Research EUROFusion project CfP-WP14-ER-01/IPP-03.

REFERENCES

[1] G. Allaire, Homogeneization and two-scale convergence, SIAM J. Math. Analysis 23 (1992) pp. 1482-1518.
[2] L.R. Burden, J.F. Douglas, Numerical Analysis, Thomson/Brooks/Cole (2005).
[3] W. Bao, Y. Cai, X. Zhao, A uniformly accurate multiscale time integrator pseudospectral method for the Klein-Gordon equation in the nonrelativistic limit regime, SIAM J. Numer. Anal. 52 (2014) pp. 2488-2511.
[4] W. Bao, X. Dong, X. Zhao, Uniformly accurate multiscale time integrators for highly oscillatory second order differential equations, J. Math. Study 47 (2014) pp. 111-150.
[5] M. Bennoune, M. Lemou, L. Mieussens, Uniformly stable numerical schemes for the Boltzmann equation preserving compressible Navier-Stokes asymptotics, J. Comput. Phys. 227 (2008) pp. 781-803.
[6] N. Besse, M. Mehrenberger, Convergence of classes of high order semi-Lagrangian schemes for the Vlasov equation, Math. of Comp. 77 (2008) pp. 93-123.
[7] C. Birdsall, A. Langdon, Plasmas physics via computer simulations, Series in plasma physics, Taylor and Francis, New York, (2005).
[8] M. Bostan, The Vlasov-Maxwell system with strong initial magnetic field. Guiding-center approximation, Multiscale Model. Simul. 6 (2007) pp. 1026-1058.
[9] P. Chartier, N. Crouseilles, M. Lemou, F. Méhats, Uniformly accurate numerical schemes for highly oscillatory Klein-Gordon and nonlinear Schrödinger equations, Numer. Math. 129 (2015) pp. 211-250.
[10] T. Colin, B. Nkonga, Multiscale numerical method for nonlinear Maxwell equations, DCDS B 5 (2005) pp. 631-658.
[11] N. Crouseilles, M. Lemou, F. Méhats, Asymptotic Preserving schemes for highly oscillatory Vlasov-Poisson equations, J. Comput. Phys. 248 (2013) pp. 287-308.
[12] N. Crouseilles, Th. Respaud, E. Sonnendrücker, A forward semi-Lagrangian method for the numerical solution of the Vlasov equation, Comput. Phys. Comm. 180 (2009) pp. 1730-1745.
[13] P. Degond, P.A. Raviart, The paraxial approximation of the Vlasov-Maxwell equations, Math. Models Methods Appl. Sci. 3 (1993) pp. 513-562.
[14] J. Denavit, Numerical simulation of plasmas with periodic smoothing in phase space, J. Comput. Phys. 9 (1972) pp. 75-98.
[15] F. Filbet, E. Sonnendrücker, Modeling and numerical simulation of space charge dominated beams in the paraxial approximation, Math. Models Methods Appl. Sci. 16 (2005) pp. 763-791.
[16] E. Frénod, S.A. Hirstoaga, M. Lutz, E. Sonnendrücker, Long time behaviour of an exponential integrator for a Vlasov-Poisson system with strong magnetic field, Communications in Comput. Phys. 18 (2015) pp. 263-296.
[17] E. Frénod, S.A. Hirstoaga, M. Lutz, Long-time simulation of a highly oscillatory Vlasov equation with an exponential integrator, Comptes Rendus Mécanique, special issue on "Theoretical and numerical approaches for Vlasov-Maxwell equations" 342 (2014) pp. 595-609.
[18] E. Frénod, S.A. Hirstoaga, E. Sonnendrücker, An exponential integrator for a highly oscillatory Vlasov equation, DCDS-S, special issue on "Numerical Methods Based on Two-Scale Convergence and Homogenization" 8 (2015) pp. 169-183.
[19] 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 (2009) pp. 175-197.
[20] E. Frénod, E. Sonnendrücker, Long time behavior of the Vlasov equation with strong external magnetic field, Math. Models Methods Appl. Sci. 10 (2000) pp. 539-553.
[21] P.E Jabin, A. Tzavaras, Kinetic decomposition of homogenization problems, SIAM J. Math. Anal. 41 (2009) pp. 360-390.
[22] S. Jin, Efficient asymptotic-preserving (AP) schemes for some multiscale kinetic equations, SIAM J. Sci. Comput. 21 (1999) pp. 441-454.

[23] M. Lemou, L. Mieussens, A new asymptotic preserving scheme based on microacro formulation for linear kinetic equations in the diffusion limit, SIAM J. Sc. Comput. 31 (2008) pp. 334?68.
[24] M. Mehrenberger, Conservative semi-Lagrangian solvers on mapped meshes, 2012 IEEE 39th International Conference on Plasma Sciences.
[25] A. Mouton, Two-scale semi-Lagrangian simulation of a charged particles beam in a periodic focusing channel, Kinet. Relat. Models 2 (2009) pp. 251-274.
[26] R.D. Nair, J.S. Scroggs, F. H.M. Semazzi, A forward-trajectory global semi-Lagrangian transport scheme, J. Comput. Phys. 190 (2003) pp. 275-294.
[27] G. Nguetseng, A general convergence result for a functional related to the theory of homogenization, SIAM J. Math. Anal. 20 (1989) pp. 608-623.
[28] T. Respaud, E. Sonnendrücker, Analysis of a new class of forward semi-Lagrangian schemes for the 1D Vlasov Poisson equations, Numer. Math. 118 (2011) pp. 329-366.
[29] A. Staniforth, J. Coté, Semi-Lagrangian integration schemes for atmospheric models - A review, Mon. Weather Rev. 119 (1991) pp. 2206-2223.
[30] J. Shen, T. Tang, L. Wang, Spectral Methods: Algorithms, Analysis and Applications, Springer, 2011.

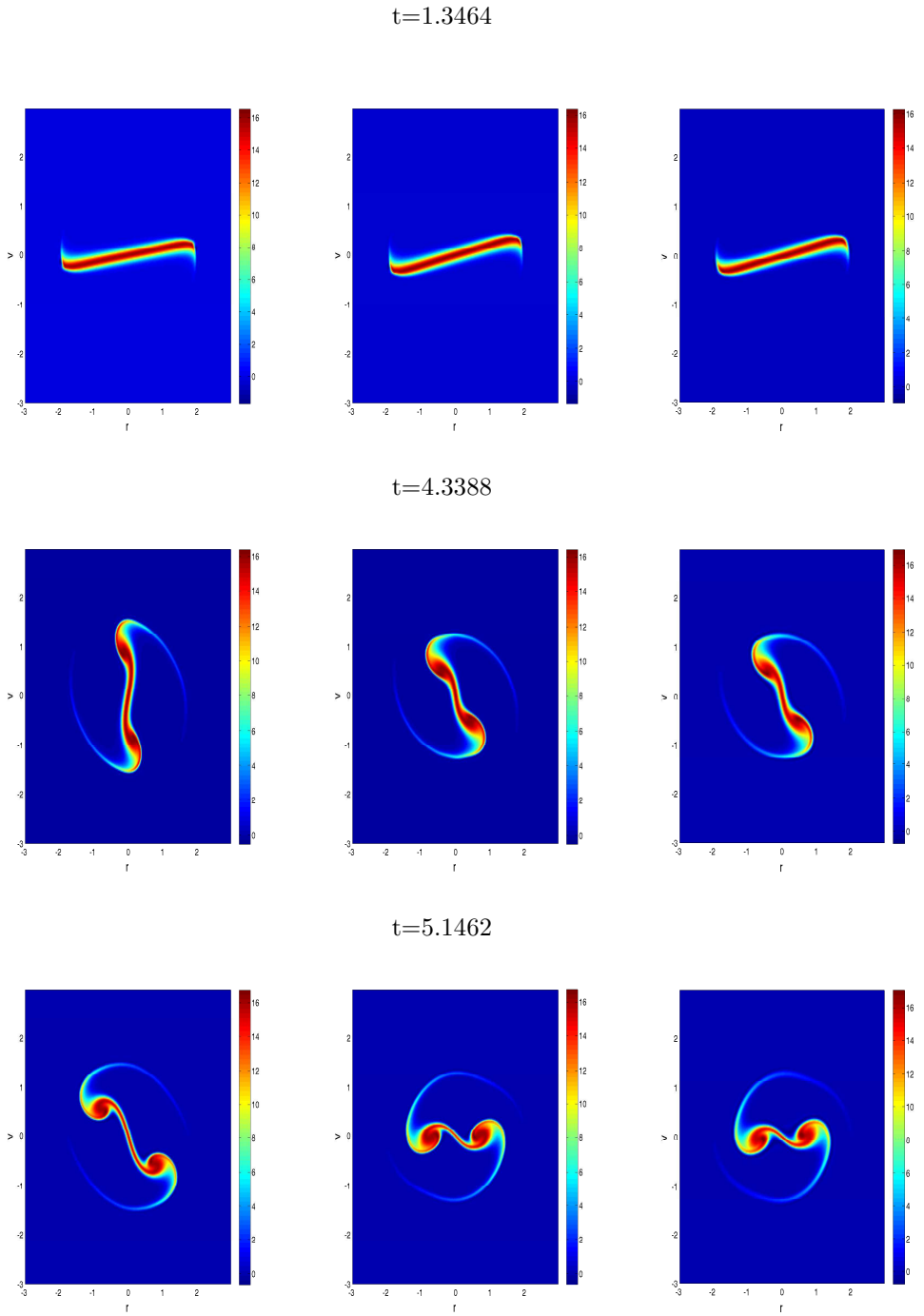
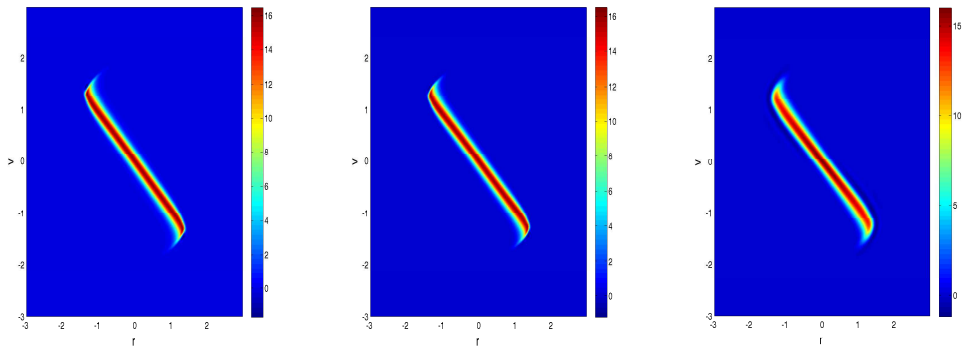
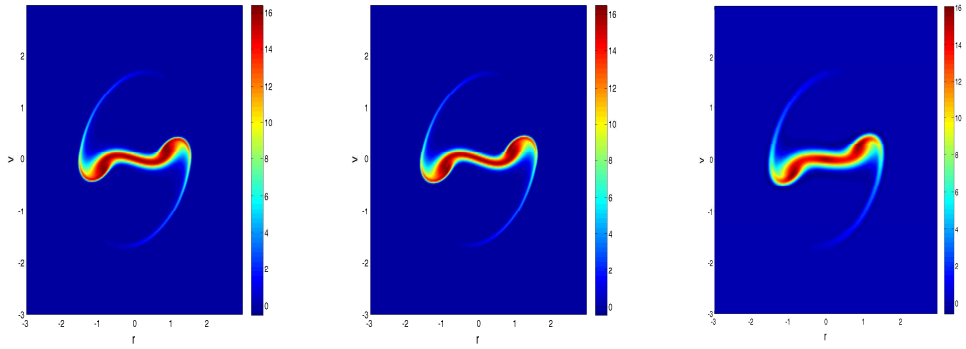


Fig. 4.7: Contour plot of the solution $f(t, r, v)$ at different time in example (4.5) case I for $\varepsilon = 0.05$: the solution given by limit model (left panel); the solution given by rotating Vlasov-Poisson equation with FSL-UA method (middle panel); the solution given by original Vlasov-Poisson equation with classical backward semi-Lagrangian method (right panel).

t=1.3464



t=4.3388



t=5.1462

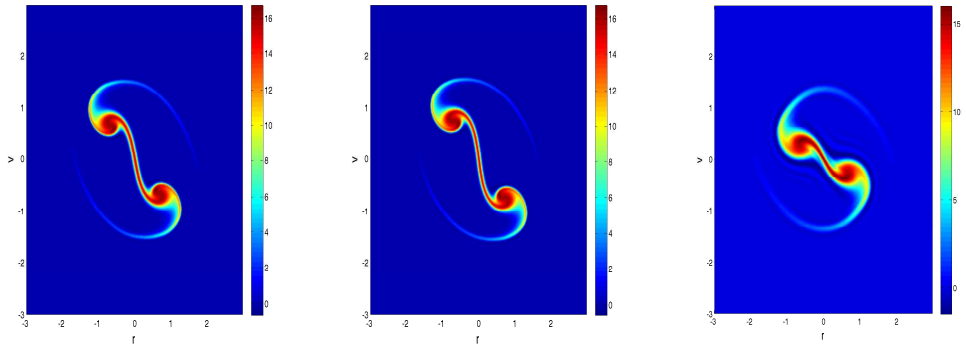


Fig. 4.8: Contour plot of the solution $f(t, r, v)$ at different time in example (4.5) case I for $\varepsilon = 0.01$: the solution given by limit model (left panel); the solution given by rotating Vlasov-Poisson equation with FSL-UA method (middle panel); the solution given by original Vlasov-Poisson equation with classical backward semi-Lagrangian method (right panel).

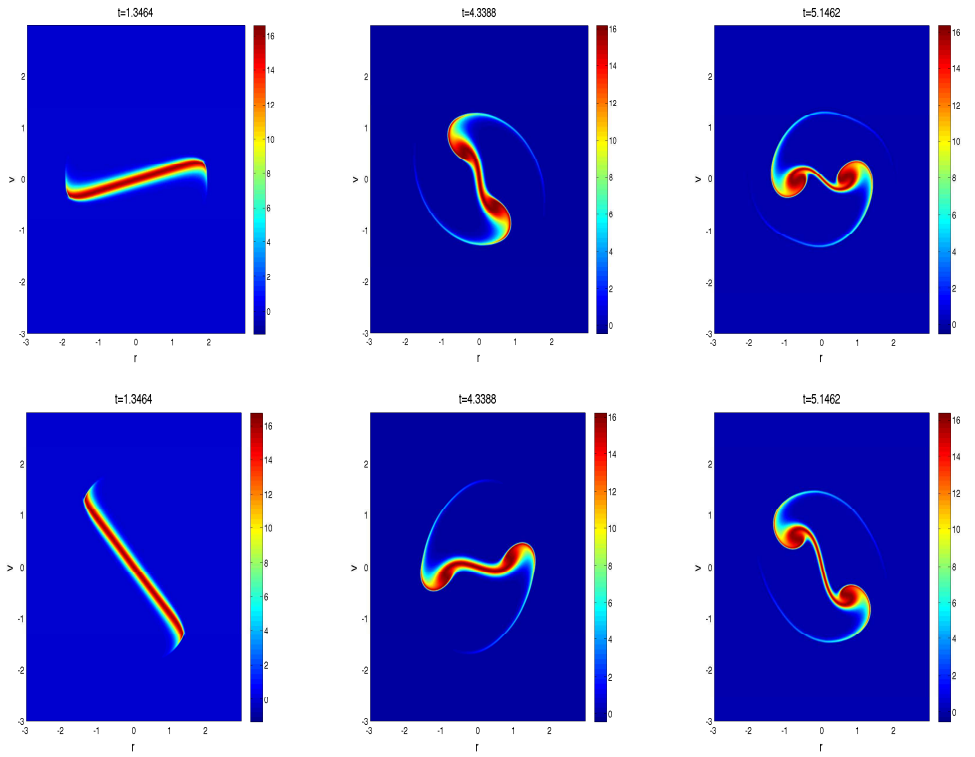


Fig. 4.9: Contour plot of the solution $f(t, r, v)$ at different time in example (4.5) case I by FSL-UA with refined time step $\Delta t \approx 5 \times 10^{-3}$ and spatial mesh $N = 512$: solutions for $\varepsilon = 0.05$ (first row); solutions for $\varepsilon = 0.01$ (second row).

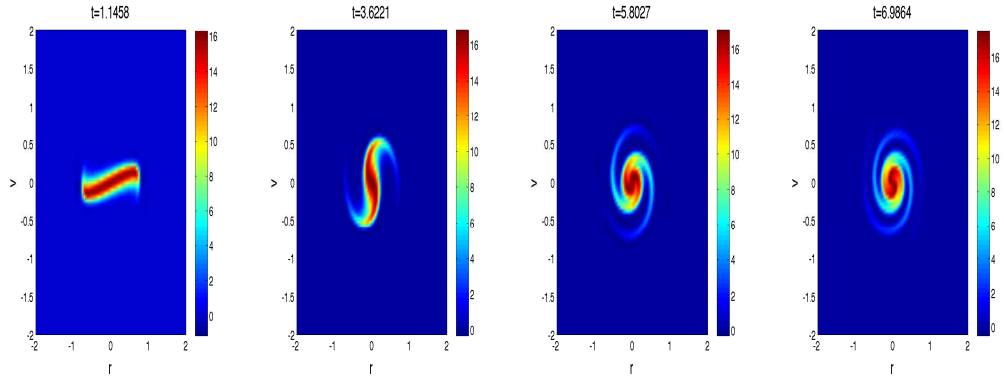


Fig. 4.10: Contour plot of the solution $f(t, r, v)$ at different time in example (4.5) case II.

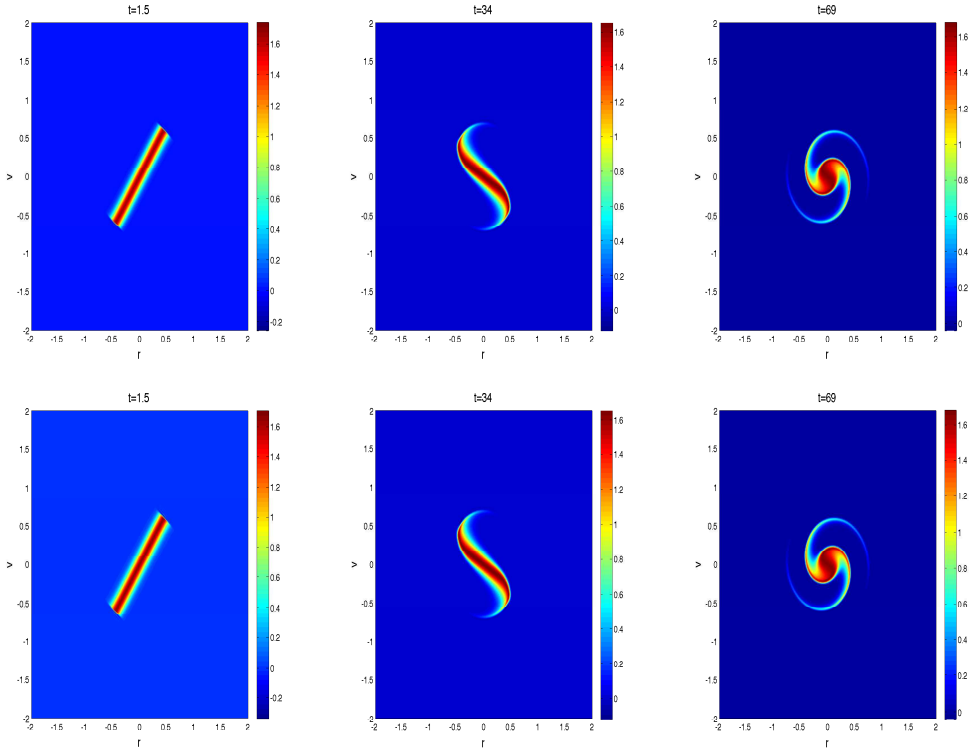


Fig. 4.11: Contour plot of the solution $f(t, r, v)$ at different time in example (4.5) case III: The first row obtained with $\Delta t = 0.5$; The second row obtained by with $\Delta t = 0.1$.