



Theoretical and numerical approaches for Vlasov–Maxwell equations

## Long-time simulation of a highly oscillatory Vlasov equation with an exponential integrator



Emmanuel Frénod<sup>a,b,\*</sup>, Sever A. Hirstoaga<sup>b,c</sup>, Mathieu Lutz<sup>b,c</sup>

<sup>a</sup> Université Bretagne-Sud, UMR 6205, LMBA, 56000 Vannes, France

<sup>b</sup> TONUS Project, Inria Nancy Grand-Est, France

<sup>c</sup> IRMA (UMR CNRS 7501), Université de Strasbourg, France

### ARTICLE INFO

#### Article history:

Received 11 March 2014

Accepted 8 June 2014

Available online 5 August 2014

#### Keywords:

Vlasov–Poisson system

Paraxial approximation

Particle-In-Cell method

Highly oscillatory ODEs

Exponential time differencing

Long-time simulation

### ABSTRACT

We change a previous time-stepping algorithm for solving a multi-scale Vlasov–Poisson system within a Particle-In-Cell method, in order to perform accurate long-time simulations. As an exponential integrator, the new scheme allows us to use large time steps compared to the size of the oscillations in the solution.

© 2014 Académie des sciences. Published by Elsevier Masson SAS. All rights reserved.

## 1. Introduction

In this paper, we improve the time numerical scheme introduced in [1], in order to make a long-time simulation of the following Vlasov equation:

$$\begin{cases} \frac{\partial f_\varepsilon}{\partial t} + \frac{v}{\varepsilon} \frac{\partial f_\varepsilon}{\partial r} + \left( -\frac{r}{\varepsilon} + E_\varepsilon(r, t) \right) \frac{\partial f_\varepsilon}{\partial v} = 0 \\ f_\varepsilon(0, r, v) = f_0(r, v) \end{cases} \quad (1)$$

where  $0 < \varepsilon \ll 1$ ,  $r \in \mathbb{R}$  is the position variable,  $v \in \mathbb{R}$  the velocity variable,  $f_\varepsilon = f_\varepsilon(r, v, t)$  is the distribution function,  $f_0$  is given, and  $-\frac{r}{\varepsilon} + E_\varepsilon(r, t)$  corresponds to the electric field. The difficulty in solving numerically this equation lies mainly in the multi-scale aspect due to the small parameter  $\varepsilon$ .

The main application will be the case when the electric field  $E_\varepsilon$  is obtained by solving the Poisson equation. We will thus have to solve the following nonlinear system of equations:

$$\begin{cases} \frac{\partial f_\varepsilon}{\partial t} + \frac{v}{\varepsilon} \frac{\partial f_\varepsilon}{\partial r} + \left( -\frac{r}{\varepsilon} + E_\varepsilon(r, t) \right) \frac{\partial f_\varepsilon}{\partial v} = 0 \\ \frac{\partial}{\partial r}(rE_\varepsilon) = \rho_\varepsilon(t, r), \quad \rho_\varepsilon(t, r) = \int_{\mathbb{R}} f_\varepsilon(t, r, v) dv \\ f_\varepsilon(0, r, v) = f_0(r, v) \end{cases} \quad (2)$$

\* Corresponding author.

This simplified model of the full Vlasov–Maxwell system is particularly adapted to the study of long and thin beams of charged particles within the paraxial approximation assuming that the beam is axisymmetric (see [2] and the references therein).

We will also test our scheme when

$$E_\varepsilon(r, t) = -r^3 \quad (3)$$

As in [1], we solve numerically the Vlasov equation by a particle method, which consists in approximating the distribution function by a finite number of macroparticles (see [3]). The trajectories of these particles are computed from the characteristic curves of the Vlasov equation, whereas the term  $E_\varepsilon$  is computed, when obtained from the Poisson equation, on a mesh in the physical space. The contribution of this work is to improve the time numerical scheme proposed in [1], for solving in long times the stiff characteristics associated with Eq. (1)

$$\begin{cases} \frac{dR}{dt} = \frac{V}{\varepsilon} \\ \frac{dV}{dt} = -\frac{R}{\varepsilon} + E_\varepsilon(R, t) \end{cases} \quad (4)$$

provided with the initial conditions  $R(0) = r$ ,  $V(0) = v$ . We note that when  $E_\varepsilon$  is given by (3), solving (4) reduces to solve the undamped and undriven Duffing equation

$$R''(t) + \frac{R(t)}{\varepsilon^2} + \frac{R^3(t)}{\varepsilon} = 0 \quad (5)$$

with initial conditions  $R(0) = r$ ,  $R'(0) = v/\varepsilon$ .

The solution of (1) is represented by a beam of particles in the phase space moving along the characteristic curves (4). When the electric field  $E_\varepsilon$  is zero, the beam rotates around the origin with a period  $2\pi\varepsilon$ . Otherwise, the dynamical system (4) can be viewed as a perturbation of the system obtained when the electric field is zero (see [4]). When the electric field  $E_\varepsilon$  is obtained by solving the Poisson equation or if it is given by (3), the beam evolves by rotating around the origin in the phase space, and in long times a bunch forms around the center of the beam, from which filaments of particles are going out. Classical explicit numerical methods require very small time step in order to capture the fast rotation. Consequently, long-time simulations are very tedious.

In [1] the authors introduced a new explicit scheme based on Exponential Time Differencing (ETD) integrators (see [5] and [6]). The idea of the algorithm was the following: first, for each macroparticle as initial condition, we solve the ODE in (4) over some fast (*i.e.* of order  $\varepsilon$ ) time by using a 4th-order Runge–Kutta solver. The fast time is the same for all the macroparticles. Then, by means of the variation-of-constants formula, we compute an approximation of the solution over a large whole number of fast times. Thus, this scheme (recalled in Algorithm 2.1) allows us to take time steps  $\Delta t$  varying from 0.1 to 1 and therefore, when  $\varepsilon$  varies from  $10^{-5}$  to  $10^{-2}$ , the numerical gain increases from 3 to  $10^5$  when comparing the scheme to a reference solution. In addition, for short-time simulation ( $t = 3.5$ ) the results are quite precise (the numerical error is about  $10^{-3}$  knowing that the beam is of size 1). For times of simulation less than 30, this ETD method still gives good qualitative results. Nevertheless, it fails to capture precisely the filamentation phenomena leading even to unstable simulation in long times.

In this paper we improve the time-stepping method following a simple idea. The fast time in the above description of the algorithm was chosen in [1] to be the mean of all the numerically computed periods for the particle trajectories. The authors remarked that this choice gives better results than using  $2\pi\varepsilon$ . The algorithm that we propose now is not to use a fixed value for the fast time, but to push each particle with its period. This means that the first step should consist in solving the ODE starting from each particle over different fast times (the computed periods of the particles). However, implementing this simple idea turns out to be not so easy and in addition asks for changing the order of the steps in the algorithm. We detail this problem in Section 3. Our simulations show that using precise periods for each particle and at each macroscopic time step results in a more accurate ETD scheme in long times. In addition, by following the proof lines in [1], we show formally that the algorithm is still an approximation when using any value for the fast time. We recall that in [1], the authors justified the algorithm when using only  $2\pi\varepsilon$  as fast time. From now on we will use the incorrect term of period to indicate the time that a particle takes to do a complete tour in the phase space.

The remainder of the paper is organized as follows. In Section 2 we briefly recall the ETD scheme introduced in [1] and we do long time simulation. Then, Section 3 is devoted to the improvement of the method. Eventually, in Section 4, we implement and test the new algorithm on the Vlasov equation in the two cases of electric field.

## 2. Long-time simulation of the exponential integrator

We first recall the exponential integrator introduced in [1] for solving efficiently in short times the Vlasov equation. Then, we discuss the long-time simulations done with this algorithm in the case of  $E_\varepsilon$  given by (3), in Section 2.2 and in the Vlasov–Poisson case in Section 2.3.

2.1. The previous exponential integrator for the Particle-In-Cell method

The time-stepping scheme that we describe in this section for solving the Vlasov equation (1) is proposed in the framework of a Particle-In-Cell method (PIC). A PIC method (see [3]) consists in approximating the distribution function  $f_\varepsilon$ , solution of (1), by a cloud of macroparticles  $(R_i, V_i)_{i \in \{1, \dots, N_p\}}$  (each one of them stands for a set of physical particles) of weight  $\omega_i$  advancing along the characteristics curves (4). In other words, we will approach the distribution function by a sum of Dirac masses centered at the macroparticles positions and velocities:

$$f_\varepsilon(r, v, t) \approx \sum_{i=1}^{N_p} \omega_i \delta(r - R_i(t)) \delta(v - V_i(t)) \tag{6}$$

The weights are chosen such that

$$\sum_{i=1}^{N_p} \omega_i = \int_{\mathbb{R}^2} f_0(r, v) dr dv \tag{7}$$

and the particles are initialized according to the probability density associated with  $f_0$ .

The aim of the exponential integrator introduced in [1] is to solve (4) by using big time steps with respect to the typical period of oscillation, which is about  $2\pi\varepsilon$ . Now, we recall the main ideas of this algorithm. For simplicity, we denote by  $\mathbf{y}$  the characteristics, i.e.

$$\mathbf{y} = \begin{pmatrix} R \\ V \end{pmatrix} \tag{8}$$

Under the assumption

$$\text{for all } n, \int_{t_n}^{t_n + N \cdot 2\pi\varepsilon} \mathcal{R}(\tau) \begin{pmatrix} 0 \\ E_\varepsilon(\tau, R(\tau)) \end{pmatrix} d\tau \approx N \cdot \int_{t_n}^{t_n + 2\pi\varepsilon} \mathcal{R}(\tau) \begin{pmatrix} 0 \\ E_\varepsilon(\tau, R(\tau)) \end{pmatrix} d\tau \tag{9}$$

where  $\mathcal{R}$  is some  $2\pi\varepsilon$ -periodic rotation in the phase space, the authors derived the following scheme. Let us fix a time step  $\Delta t \gg 2\pi\varepsilon$  and determine the unique integer  $N$  and the unique real  $o$  such that

$$\begin{cases} \Delta t = N \cdot 2\pi\varepsilon + o \\ 0 \leq o < 2\pi\varepsilon \end{cases} \tag{10}$$

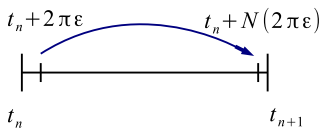
For each macroparticle  $\mathbf{y}_i$  (denoted by  $\mathbf{y}$  for simplicity), let us apply Algorithm 2.1 below.

**Algorithm 2.1.** Assume that  $\mathbf{y}_n$  the solution of (4) at time  $t_n$  is given.

1. Compute  $\mathbf{y}(t_n + 2\pi\varepsilon)$  by using a fine Runge–Kutta solver with initial condition  $\mathbf{y}_n$ .
2. Compute  $\mathbf{y}(t_n + N \cdot 2\pi\varepsilon)$  by the rule:

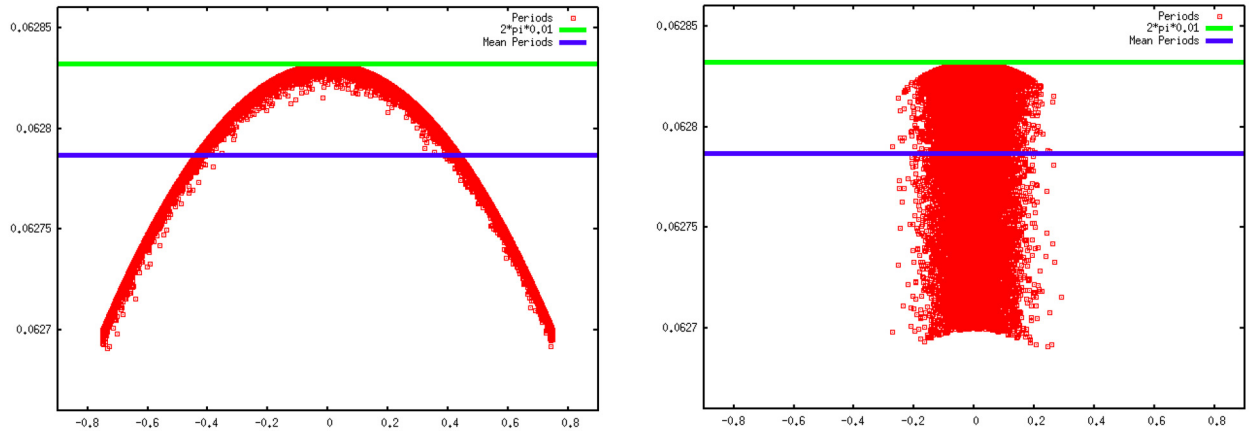
$$\mathbf{y}(t_n + N \cdot 2\pi\varepsilon) = \mathbf{y}_n + N(\mathbf{y}(t_n + 2\pi\varepsilon) - \mathbf{y}_n) \tag{11}$$

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



**Remark 2.2.** Hypothesis (9) is satisfied if the time to make one rapid complete tour (in the phase space) is quasi constant and close to  $2\pi\varepsilon$  and if  $\tau \mapsto E_\varepsilon(\tau, \mathbf{y}(\tau))$  is about  $2\pi\varepsilon$ -periodic.

This simple algorithm was formally proved in [1] to provide an approximation of the solution under the assumption in (9). The capability of the scheme relies on the following. When the solution is periodic, say  $\mathbf{y}(t_n + 2\pi\varepsilon) = \mathbf{y}_n$ , the second step of the algorithm reduces to an exact statement. Otherwise, it can happen that  $\|\mathbf{y}(t)\|_2 = \sqrt{R(t)^2 + V(t)^2}$  vary slowly in time, typically when the electric field is given by the Poisson equation (see [4]). In this case, the scheme allows us to capture the small variations of the amplitude of  $\mathbf{y}$  since it solves them during the first step. Nevertheless, it should be noted that the algorithm works provided that these variations do not change considerably during the second step.



**Fig. 1.** In red the computed periods with respect to the initial positions (left) and to the initial velocities (right). The case of  $E_\varepsilon = -r^3$  with  $\varepsilon = 0.01$ . In blue the mean of the periods, in green the  $2\pi\varepsilon$  value. (For interpretation of the references to color in this figure, the reader is referred to the online version of this article.)

Therefore, our aim now is to improve the algorithm in order to be able to capture also the small variations of the period. This viewpoint was outlined in [1]. More precisely, the authors noticed that the results are much better, especially in the Vlasov–Poisson case, when replacing  $2\pi\varepsilon$  in Algorithm 2.1 by the mean of the times that particles take to make the first complete tour. The so-obtained scheme was called the *modified ETD algorithm*. However, as we will see in the next two sections, this scheme still needs to be improved in order to do accurate long-time simulations. The idea is that using a more accurate period when pushing particles leads to better results.

2.2. Application to long-time simulation of the undamped and undriven Duffing equation

In this section, we consider the Vlasov equation (1) provided with the electric field given by (3) and with the initial condition (see [2])

$$f_0(r, v) = \frac{n_0}{\sqrt{2\pi} v_{th}} \exp\left(-\frac{v^2}{2v_{th}^2}\right) \chi_{[-0.75, 0.75]}(r) \tag{12}$$

where the thermal velocity is  $v_{th} = 0.0727518214392$  and  $\chi_{[-0.75, 0.75]}(r) = 1$  if  $r \in [-0.75, 0.75]$  and 0 otherwise. In the particle approximation, we implement this distribution function with  $N_p = 20000$  macroparticles and equal weights  $w_i = 1/N_p$ . Moreover, this initial condition will be used all along this work.

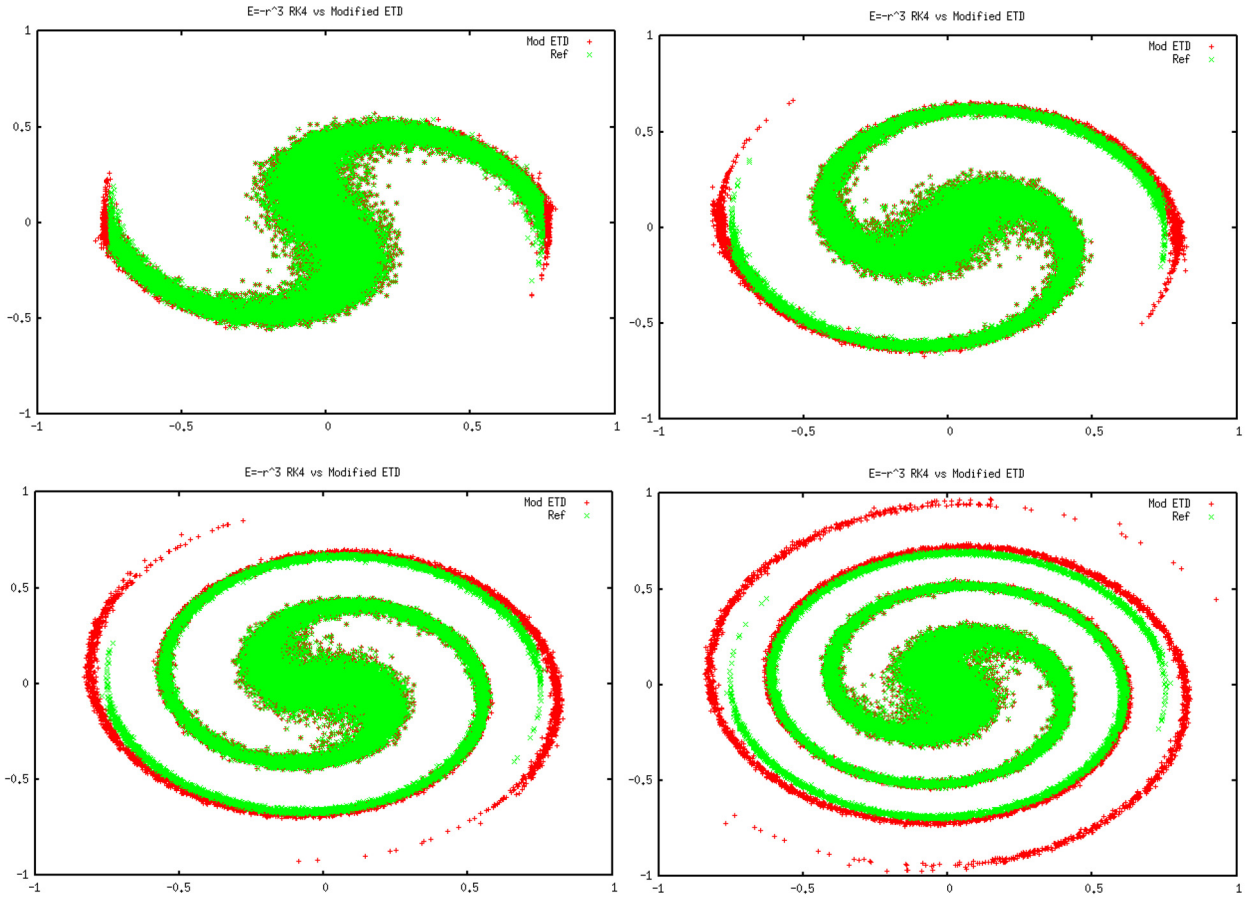
In this case, the characteristic equation or equivalently the Duffing equation (5) has periodic solutions in time (see the Appendix A). Thus, the time taken to make one rapid complete tour in the phase space depends only on the initial condition. It is actually constant in time and it corresponds to the period of the trajectory. The repartition of the periods with respect to the initial condition in (12) is given in Fig. 1. These values are computed with a fine RK4 solver and they are in accordance with formula (34) in the Appendix A. Notice that the periods are very close to  $2\pi\varepsilon$ , they belong to the  $(2\pi\varepsilon - 1.5\varepsilon^2, 2\pi\varepsilon)$  interval.

Now we compare the modified ETD scheme with a reference solution, obtained by solving (4) with an explicit 4th-order Runge–Kutta scheme with small time step. The solution in the phase space  $(R, V)$  at different large times, obtained with  $\varepsilon = 0.01$ , is illustrated in Fig. 2.

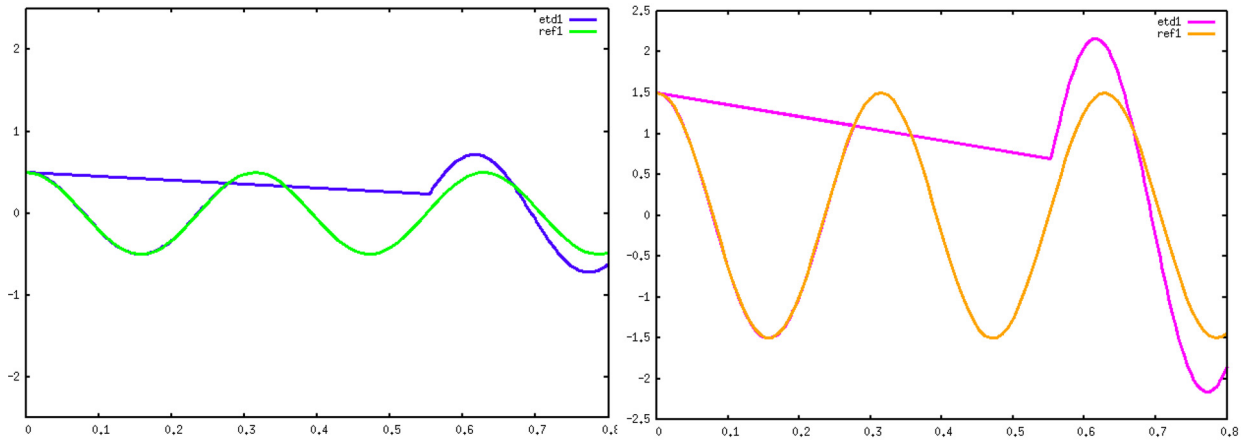
We notice that beyond  $t = 30$  the errors become significant. Next, we explain the reasons. Imagining a position (or velocity) trajectory in time being sinusoid-like, it is easy to see that a particle for which the period is underestimated (by the mean period) will drift inward in the phase space when applying the second step of the algorithm. When the period is overestimated, the particle will clearly drift outward. In addition, with the same approximation of the periods, larger is the amplitude of the sinusoid, bigger is the error made by the third step of the modified ETD scheme (see Fig. 3). Consequently, all the particles close to  $(0, 0)$  in the phase space are well treated by the algorithm, while the particles far from the center of the beam accumulate significant errors in time. This behavior was noticed in reference [1] in short-time simulation, in terms of errors for particles off or on the slow manifold.

2.3. Application to long-time simulation of the Vlasov–Poisson test case

In this section we consider the Vlasov–Poisson equation (2) provided with the initial condition (12). The calculus done in the appendix for computing periods is now difficult to achieve. Therefore, in this case, we compute the time taken to make one fast tour by using a fine RK4 solver. The Poisson equation is simply solved by means of the trapezoidal formula for the



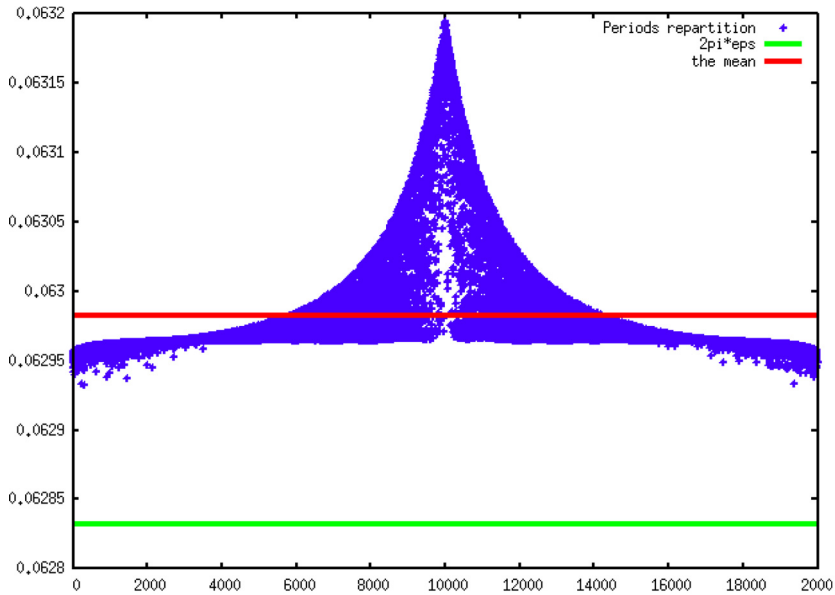
**Fig. 2.** The case of  $E_\varepsilon = -r^3$  with  $\varepsilon = 0.01$ . In green: the reference solution obtained with a Runge-Kutta algorithm with time step  $\Delta t = 2\pi\varepsilon/100$ . In red: the result of the modified ETD scheme with time step  $\Delta t = 0.5$ . The final times are, from left to right and from top to bottom,  $t = 10, 20, 30$ , and  $40$ . (For interpretation of the references to color in this figure, the reader is referred to the online version of this article.)



**Fig. 3.** In green: the amplitude of the exact solution is  $0.5$  and the period is  $2\pi\varepsilon$ . In blue: the solution obtained with the ETD Scheme with a period  $2\pi\varepsilon - 15\varepsilon^2$ . In orange: the amplitude of the exact solution is  $1.5$  and the period is  $2\pi\varepsilon$ . In purple the solution obtained with the ETD Scheme with a period  $2\pi\varepsilon - 15\varepsilon^2$ . (For interpretation of the references to color in this figure, the reader is referred to the online version of this article.)

integral in  $r$ , with 256 cells. As in the previous section, the fast time depends on the initial condition, but in addition, it changes slowly in time. The same happens for the amplitude  $\|\mathbf{y}(t)\|_2 = \sqrt{R(t)^2 + V(t)^2}$ , see Figs. 5 and 6.

The comparison between the modified ETD scheme and a reference solution obtained with explicit 4th-order Runge-Kutta scheme with small time step is summarized in Fig. 7. We see from Figs. 2 and 7 that the dynamics and/or the



**Fig. 4.** In blue: the computed periods with respect to the initial positions, in the Vlasov–Poisson case with  $\varepsilon = 0.01$ . In red: the mean of the periods, in green the  $2\pi\varepsilon$  value. (For interpretation of the references to color in this figure, the reader is referred to the online version of this article.)

filamentation in the Vlasov–Poisson case is slower than that in the case of  $E_\varepsilon$  given by (3). Certainly, this is because of bigger particle periods (compare Figs. 1 and 4).

Now, we can recall the discussion from the previous section about the consequences of using approximated period within the second and the third steps of the algorithm. However, in this Vlasov–Poisson case, the periods’ distribution is more complicated and thus, it seems more difficult to clearly state how the approximation influences long-time simulation. We only remark the following. The error done for particles at the extremities of the beam is less significant than in the previous Vlasov case. The reason is that the mean of the initial periods is very close to the periods of the particles far from the center of the beam (see Figs. 4 and 5). As for those close to the center (0, 0), we can see in Fig. 7 that particles slightly drifting both outward and inward are present, as expected from Fig. 4.

Therefore, in the next section, we will compute at each time step the period of each particle, in order to take into account this slow evolution of the periods distribution. We recall that the slow evolution of the amplitudes is considered when applying the first step of the algorithm.

### 3. The new time-stepping scheme

In this section, we change Algorithm 2.1 by improving the approximation the time taken by each particle to make one rapid complete tour. More precisely, within the first step, instead of solving the ODE (4) during some fixed fast time for all the particles, we will push each particle with its own fast time. Unfortunately, implementing this idea in Algorithm 2.1 will not work in the Vlasov–Poisson case. The reason is the following. Suppose that we have the particles  $(\mathbf{y}_i)_{i \in \{1, \dots, N_p\}}$  and their computed periods  $(T_i)_{i \in \{1, \dots, N_p\}}$  at time  $t_n$ . First, we finely solve the ODE with each particle as initial condition, during different times (the  $T_i$ s), paying attention to push enough in time (the maximum period) all the particles in order to calculate the correct self-consistent electric field. Then, applying the second step of Algorithm 2.1 with the corresponding  $N_i$  will lead to particles at different times  $t_n + N_i T_i$ . Now, it is impossible to apply the third step of the algorithm, because the particles arrived at the minimum of  $(t_n + N_i T_i)_{i \in \{1, \dots, N_p\}}$  can not be pushed with the right electric field, since all the particles are not available at that time. This problem is solved by changing the steps of the algorithm, as it is described below, in Algorithm 3.4.

Firstly, note that dynamical system (4) can be rewritten as

$$\mathbf{y}'(t) = \frac{1}{\varepsilon} M \mathbf{y}(t) + \mathbf{L}^\varepsilon(t, \mathbf{y}(t)) \tag{13}$$

where  $\mathbf{y}$  is defined by (8) and  $M$  and  $\mathbf{L}^\varepsilon$  are given by

$$M = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \quad \text{and} \quad \mathbf{L}^\varepsilon = \begin{pmatrix} 0 \\ E_\varepsilon \end{pmatrix} \tag{14}$$

Let  $T$  be a positive real number which is intended to be close to  $2\pi$  and let  $\mathbf{r}(\tau)$  be the matrix defined by

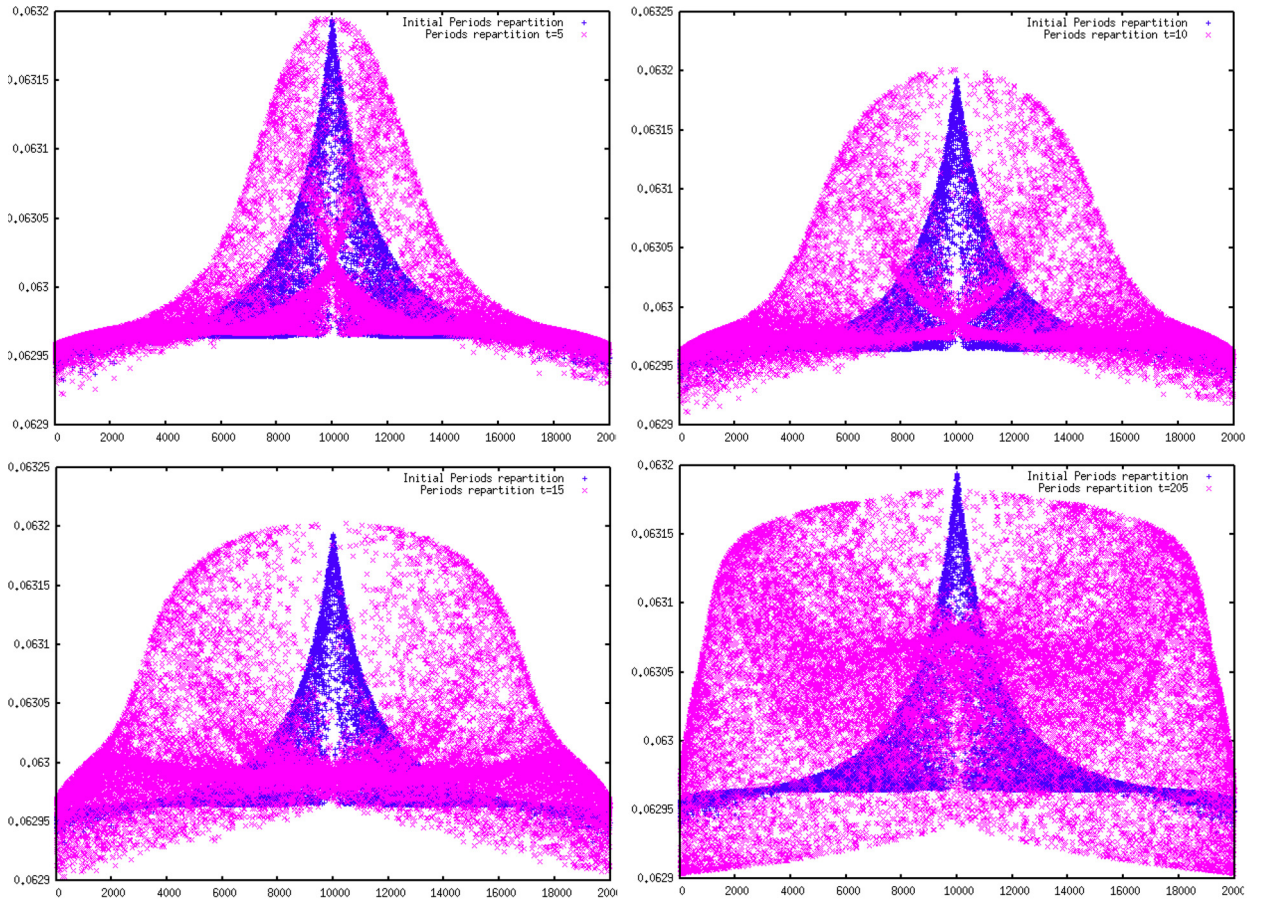


Fig. 5. The periods distribution with respect to the initial periods (in blue) at times  $t = 5, 10, 15,$  and  $20$ . The Vlasov–Poisson case with  $\varepsilon = 0.01$ . (For interpretation of the references to color in this figure, the reader is referred to the online version of this article.)

$$\mathbf{r}(\tau) = \begin{pmatrix} \cos(\frac{2\pi}{T} \tau) & \sin(\frac{2\pi}{T} \tau) \\ -\sin(\frac{2\pi}{T} \tau) & \cos(\frac{2\pi}{T} \tau) \end{pmatrix} \tag{15}$$

which is a  $T$ -periodic function. Then we compute

$$\begin{aligned} \frac{d}{d\tau} \left( \mathbf{r} \left( -\frac{\tau}{\varepsilon} \right) \mathbf{y}(\tau) \right) &= \frac{2\pi}{\varepsilon T} \mathbf{r} \left( -\frac{\tau}{\varepsilon} \right) M \mathbf{y}(\tau) + \mathbf{r} \left( -\frac{\tau}{\varepsilon} \right) \left( \frac{1}{\varepsilon} M \mathbf{y}(t) + \mathbf{L}^\varepsilon(\tau, \mathbf{y}(\tau)) \right) \\ &= \mathbf{r} \left( -\frac{\tau}{\varepsilon} \right) \mathbf{L}^\varepsilon(\tau, \mathbf{y}(\tau)) + \left( 1 - \frac{2\pi}{T} \right) \frac{1}{\varepsilon} \mathbf{r} \left( -\frac{\tau}{\varepsilon} \right) M \mathbf{y}(\tau) \\ &= \mathbf{r} \left( -\frac{\tau}{\varepsilon} \right) \boldsymbol{\beta}^\varepsilon(\tau, \mathbf{y}(\tau)) \end{aligned} \tag{16}$$

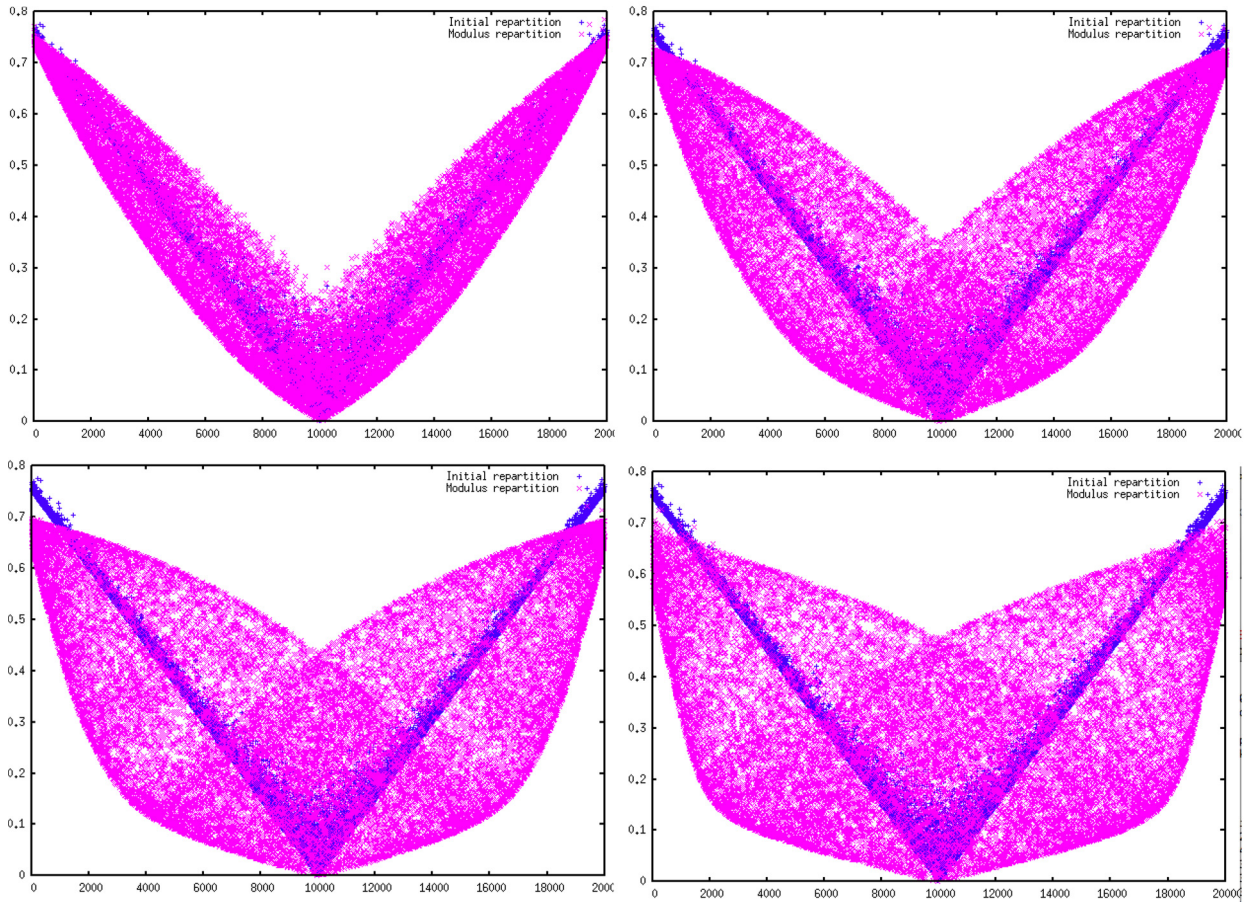
where

$$\boldsymbol{\beta}^\varepsilon(\tau, \mathbf{y}(\tau)) = \mathbf{L}^\varepsilon(\tau, \mathbf{y}(\tau)) + \left( 1 - \frac{2\pi}{T} \right) \frac{1}{\varepsilon} M \mathbf{y}(\tau) \tag{17}$$

Integrating (16) between  $s$  and  $t$  with  $s < t$  thus leads to

$$\mathbf{y}(t) = \mathbf{r} \left( \frac{t-s}{\varepsilon} \right) \mathbf{y}(s) + \mathbf{r} \left( \frac{t-s}{\varepsilon} \right) \int_s^t \mathbf{r} \left( \frac{s-\tau}{\varepsilon} \right) \boldsymbol{\beta}^\varepsilon(\tau, \mathbf{y}(\tau)) d\tau \tag{18}$$

Now we establish the time-stepping scheme. We write Eq. (18) 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  $t_n$ . Thus, for each particle of the beam, we are faced with the numerical computation of the integral from  $t_n$  to  $t_{n+1}$  involved in the right-hand side of (18). Subsequently, we



**Fig. 6.** The modulus  $\|(R(t), V(t))\|_2$  with respect to the initial modulus (in blue) at times  $t = 5, 10, 15,$  and  $20$ . The Vlasov–Poisson case with  $\varepsilon = 0.01$ . (For interpretation of the references to color in this figure, the reader is referred to the online version of this article.)

denote by  $\mathbf{y}_i$  the particle satisfying Eq. (13) provided with the initial condition  $\mathbf{y}_i^n$ . We denote by  $T_i^n$  the time to make the first rapid complete tour starting at  $t_n$ . This time is computed numerically.

Since we want to build a scheme with a time step  $\Delta t$  much larger than the fast oscillation, we first need to find the unique positive integers  $N_i^n$  and the unique reals  $o_i^n \in [0, T_i^n]$  such that

$$\Delta t = N_i^n T_i^n + o_i^n \tag{19}$$

The derivation of the scheme, Algorithm 3.4, is based on the following approximation.

**Approximation 3.1.** We have for any  $i \in \{1, \dots, N_p\}$

$$\int_t^{t+N_i^n T_i^n} \mathbf{r}_i^n \left( \frac{t-\tau}{\varepsilon} \right) \boldsymbol{\beta}^\varepsilon(\tau, \mathbf{y}_i(\tau)) d\tau \approx N_i^n \int_t^{t+T_i^n} \mathbf{r}_i^n \left( \frac{t-\tau}{\varepsilon} \right) \boldsymbol{\beta}^\varepsilon(\tau, \mathbf{y}_i(\tau)) d\tau \tag{20}$$

where  $t = t_n + o_i^n$  and  $\mathbf{r}_i^n$  corresponds to the matrix in (15) with  $T = T_i^n / \varepsilon$ .

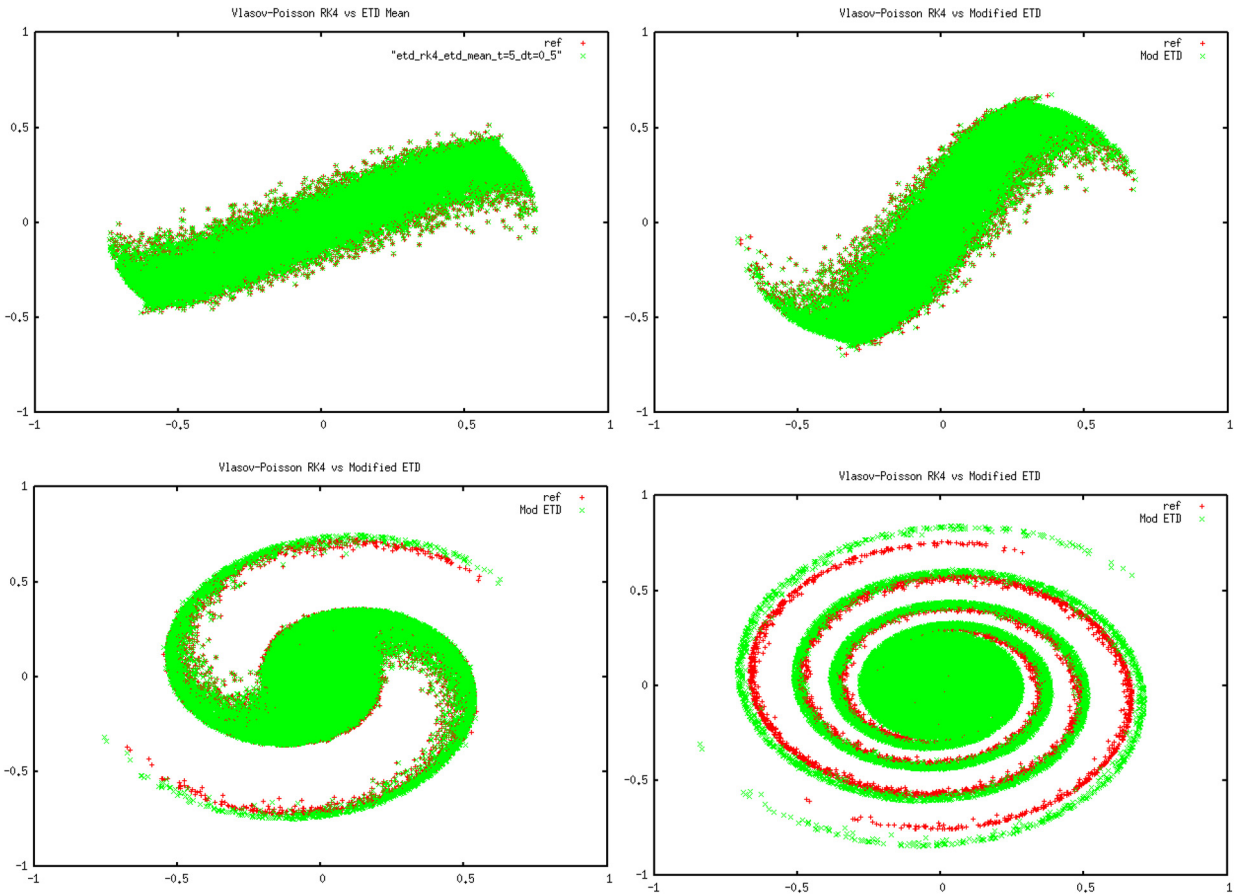
**Remark 3.2.** Approximation 3.1 is valid if we make the assumptions that the times for all particles to do the first rapid tour starting from  $t$  evolve slowly in time and that the particle and the electric field evaluated at the particle position are quasi-periodic in time (with a period close to the time needed to make the first rapid complete tour).

**Lemma 3.3.** Under Approximation 3.1, we obtain for any  $i \in \{1, \dots, N_p\}$  and any time  $t_n$

$$\mathbf{y}_i(t_{n+1}) \approx \mathbf{y}_i(t_n + o_i^n) + N_i^n (\mathbf{y}_i(t_n + o_i^n + T_i^n) - \mathbf{y}_i(t_n + o_i^n)) \tag{21}$$

where  $\mathbf{y}_i$  is the solution to Eq. (13) with a given initial condition  $\mathbf{y}_i(t_n) = \mathbf{y}_i^n$ .





**Fig. 7.** Vlasov–Poisson test case with  $\varepsilon = 0.01$ . In red: the reference solution obtained with a Runge–Kutta algorithm with time step  $\Delta t = 2\pi\varepsilon/100$ . In green: the result of the modified ETD scheme with time step  $\Delta t = 0.5$ . The final times are, from left to right and from top to bottom:  $t = 5, 10, 30$ , and  $60$ . (For interpretation of the references to color in this figure, the reader is referred to the online version of this article.)

**Proof.** Writing the variation-of-constants formula from  $s = t_n + o_i^n$  to  $t = t_{n+1}$  and recalling (19) we obtain the following exact scheme

$$\mathbf{y}_i(t_{n+1}) = \mathbf{y}_i(t_n + o_i^n) + \int_{t_n + o_i^n}^{t_{n+1}} \mathbf{r}_i^n \left( \frac{t_n + o_i^n - \tau}{\varepsilon} \right) \boldsymbol{\beta}^\varepsilon(\tau, \mathbf{y}_i(\tau)) d\tau \tag{22}$$

since  $\mathbf{r}_i^n(N_i^n T_i^n / \varepsilon)$  is the identity matrix. Then, under Approximation 3.1 we obtain:

$$\mathbf{y}_i(t_{n+1}) \approx \mathbf{y}_i(t_n + o_i^n) + N_i^n \int_{t_n + o_i^n}^{t_n + o_i^n + T_i^n} \mathbf{r}_i^n \left( \frac{t_n + o_i^n - \tau}{\varepsilon} \right) \boldsymbol{\beta}^\varepsilon(\tau, \mathbf{y}_i(\tau)) d\tau \tag{23}$$

Applying now formula (18) with  $s = t_n + o_i^n$ ,  $t = t_n + o_i^n + T_i^n$  and  $\mathbf{r} = \mathbf{r}_i^n$  yields

$$\mathbf{y}_i(t_n + o_i^n + T_i^n) - \mathbf{y}_i(t_n + o_i^n) = \int_{t_n + o_i^n}^{t_n + o_i^n + T_i^n} \mathbf{r}_i^n \left( \frac{t_n + o_i^n - \tau}{\varepsilon} \right) \boldsymbol{\beta}^\varepsilon(\tau, \mathbf{y}_i(\tau)) d\tau \tag{24}$$

since  $\mathbf{r}_i^n(T_i^n / \varepsilon)$  is the identity matrix. Thus, injecting (24) in (23) leads to (21).  $\square$

Using Lemma 3.3, we deduce the following algorithm to compute  $\mathbf{y}_i^{n+1}$  from  $\mathbf{y}_i^n$ .

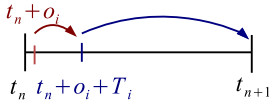
**Algorithm 3.4.** Assume that for each  $i \in \{1, \dots, N_p\}$ ,  $\mathbf{y}_i^n$ , the value of a solution of (13) at time  $t_n$ , is given.

1. Compute from  $(\mathbf{y}_i^n)_{i \in \{1, \dots, N_p\}}$  the periods  $(T_i^n)_{i \in \{1, \dots, N_p\}}$  at time  $t_n$ .
2. For each  $i \in \{1, \dots, N_p\}$  compute the unique positive integers  $N_i^n$  and the unique reals  $o_i^n \in [0, T_i^n)$  such that

$$\Delta t = N_i^n T_i^n + o_i^n \tag{25}$$

3. For each  $i \in \{1, \dots, N_p\}$  compute  $\mathbf{y}_i(t_n + o_i^n)$  and  $\mathbf{y}_i(t_n + o_i^n + T_i^n)$  by using a fine Runge–Kutta solver with initial condition  $\mathbf{y}_i^n$ .
4. For each  $i \in \{1, \dots, N_p\}$  compute an approximation of  $\mathbf{y}_i(t_{n+1})$  thanks to

$$\mathbf{y}_i^{n+1} = \mathbf{y}_i(t_n + o_i^n) + N_i(\mathbf{y}_i(t_n + o_i^n + T_i^n) - \mathbf{y}_i(t_n + o_i^n)) \tag{26}$$



**Remark 3.5.** As in [1], the way to compute the  $(T_i^n)_i$  from  $(\mathbf{y}_i^n)_i$  is the following. First, we solve the characteristics of Eq. (4) with a time step  $(\Delta t)_{\text{RK}} = 2\pi\varepsilon/100$ , for all initial conditions  $\mathbf{y}_i^n$  until all the particles reach their trajectory’s third extremum. The criterion for finding these extrema is the velocity’s change of sign. We then state that each particle’s period is the time interval between the first and the third extremum.

**Remark 3.6.** The time step  $(\Delta t)_{\text{RK}}$  chosen in the third step of the algorithm is  $2\pi\varepsilon/100$ . It is easy to see that the reals  $(t_n + o_i^n)_i$  and  $(t_n + o_i^n + T_i^n)_i$  are located on a few cells of length  $(\Delta t)_{\text{RK}}$ . Thus, we approximate the values  $\mathbf{y}_i(t_n + o_i^n)$  and  $\mathbf{y}_i(t_n + o_i^n + T_i^n)$  by quadratic interpolations.

#### 4. Numerical simulations

We now do long-time simulations with the *improved ETD scheme* (Algorithm 3.4) in the two cases of Vlasov equation. The results show that the new scheme works in long times while the modified ETD scheme becomes unstable. We can observe the idea underlined from the beginning of the paper: using all along the simulation accurate approximations of each particle period leads to very accurate solutions. Thus, we note that the numerical solutions in the first case (Section 4.1) are very good since very precise values for the periods are available and in addition they do not change in time. On the contrary, in the Vlasov–Poisson case the periods are numerically approximated at several levels and therefore the scheme does not perform as well as in the first case.

##### 4.1. Long-time simulation of the undamped and undriven Duffing equation

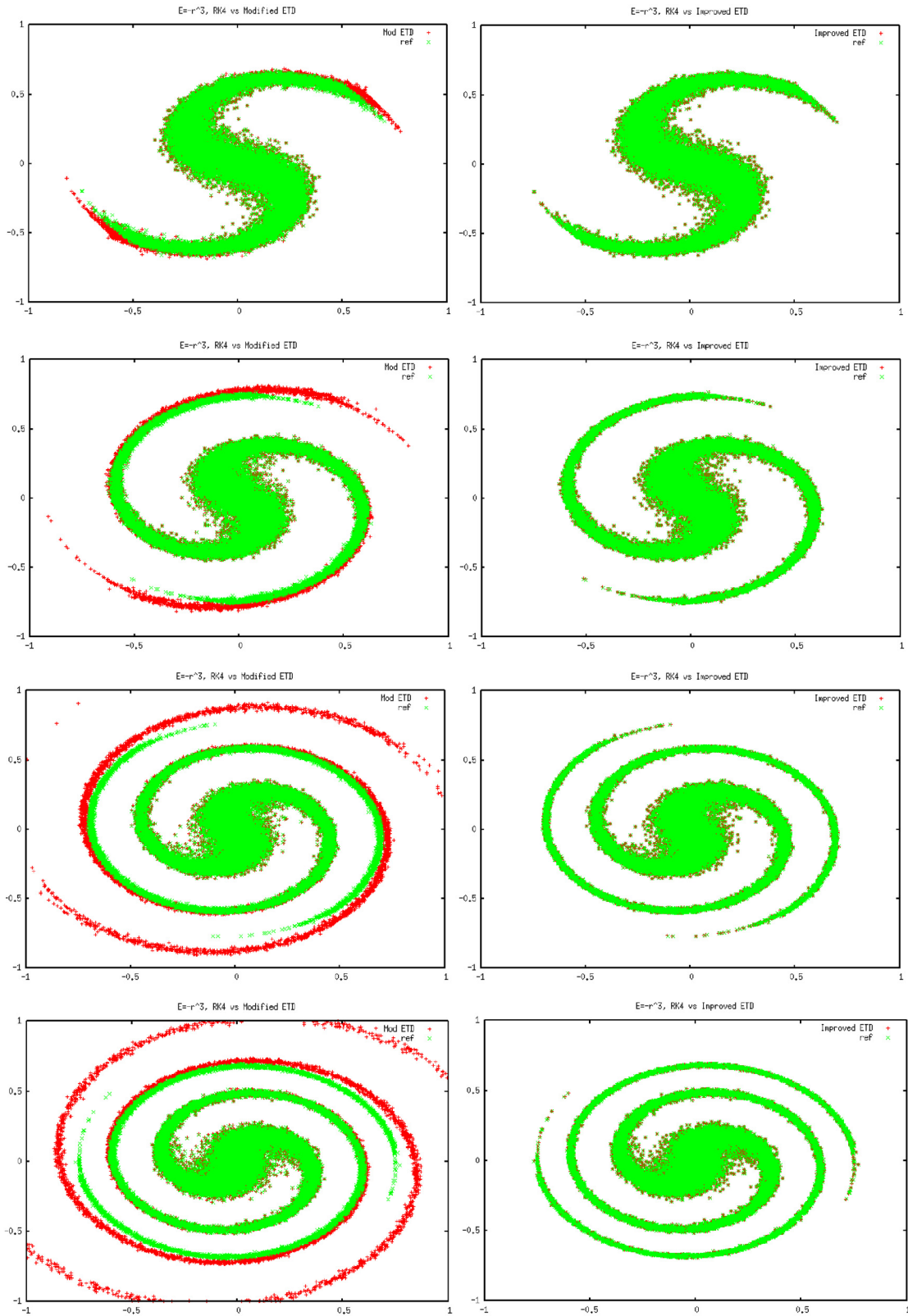
In this section we consider the Vlasov equation (1) provided with the electric field given by (3) and with the initial condition (12). We recall that we use 20000 macroparticles with equal weights. Note that the first step of Algorithm 3.4 is executed only initially, since the particle periods are constant in time.

The results obtained with  $\varepsilon = 0.001$  are summarized in Fig. 8 and with  $\varepsilon = 0.0001$  in Fig. 10 at right: we compare the outcome of the algorithm with a macroscopic time step  $\Delta t = 0.5$  to a reference solution computed with a small time step,  $2\pi\varepsilon/100$ . As already mentioned in Section 2.2 (see also the Appendix A), the particles moving according to (4) with given  $E_\varepsilon$ , have periodic trajectories and thus, the periods depend only on the initial conditions. We observe in Figs. 8 and 10 that both the filamentation phenomena and the fast rotation are well treated by the improved ETD scheme. Notice also that when  $\varepsilon = 0.0001$ , the integers  $N_i^n$  are about 800, meaning that the ETD scheme is about 800 times faster than the reference solution.

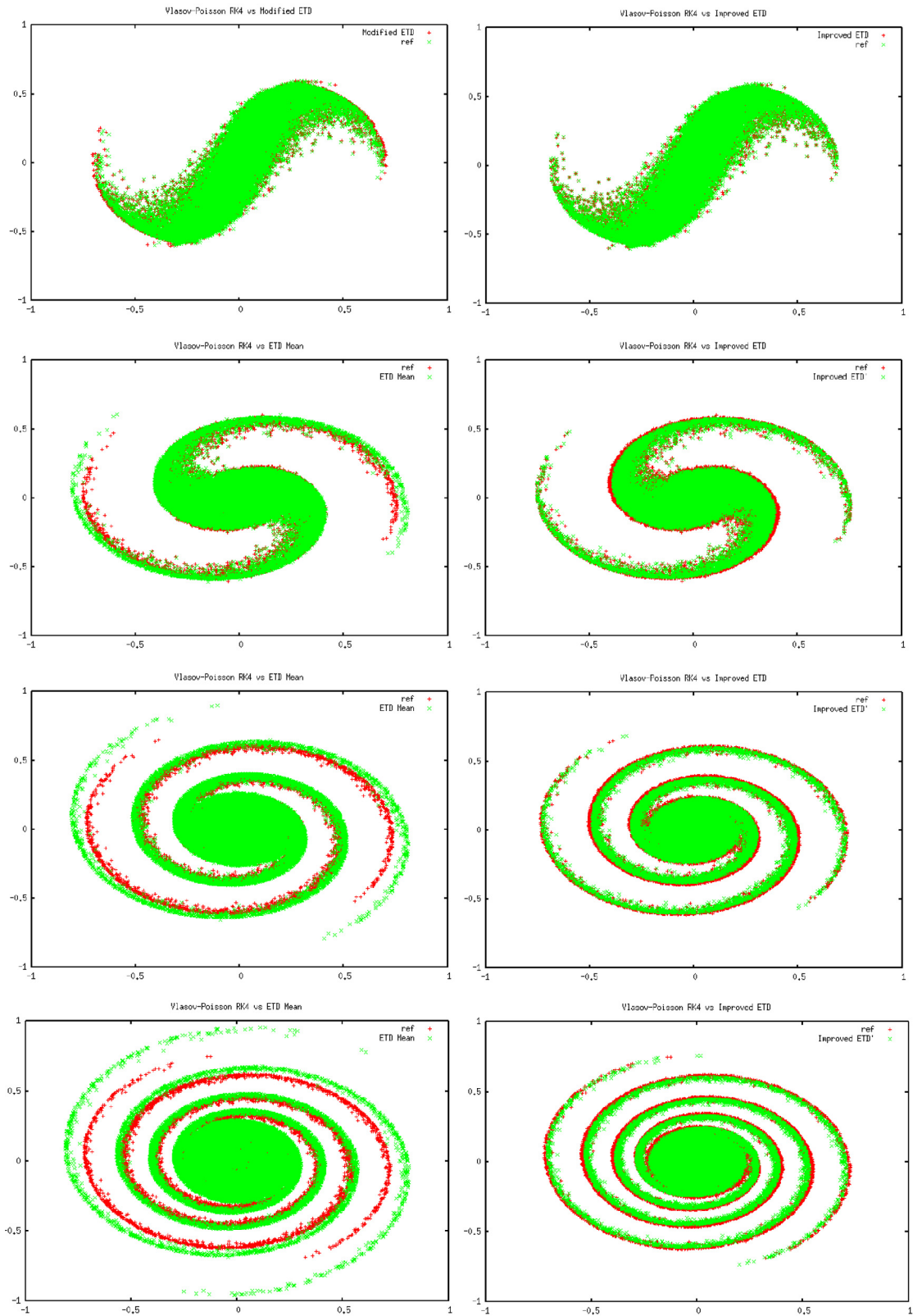
##### 4.2. Long-time simulation of the Vlasov–Poisson test case

In this section we consider the Vlasov–Poisson equation (2) provided with the initial condition (12). The Poisson equation is solved as mentioned in Section 2.3. The results obtained with  $\varepsilon = 0.001$  and the same time steps as above are summarized in Fig. 9. The case of  $\varepsilon = 0.0001$  is illustrated in Fig. 10 (on the left). Again, one can see the slower filamentation of the Vlasov–Poisson case in comparison with the given electric field case. The time needed to make one rapid complete tour evolves slowly in time (see Section 2.3). Therefore, the approximations made in Algorithm 3.4 lead to accurate results. In particular, the particle filaments are well treated by the improved ETD scheme (the old scheme is unstable at these times). Note again that the new algorithm is about 80 times faster (when  $\varepsilon = 0.001$ ) than the reference solution.

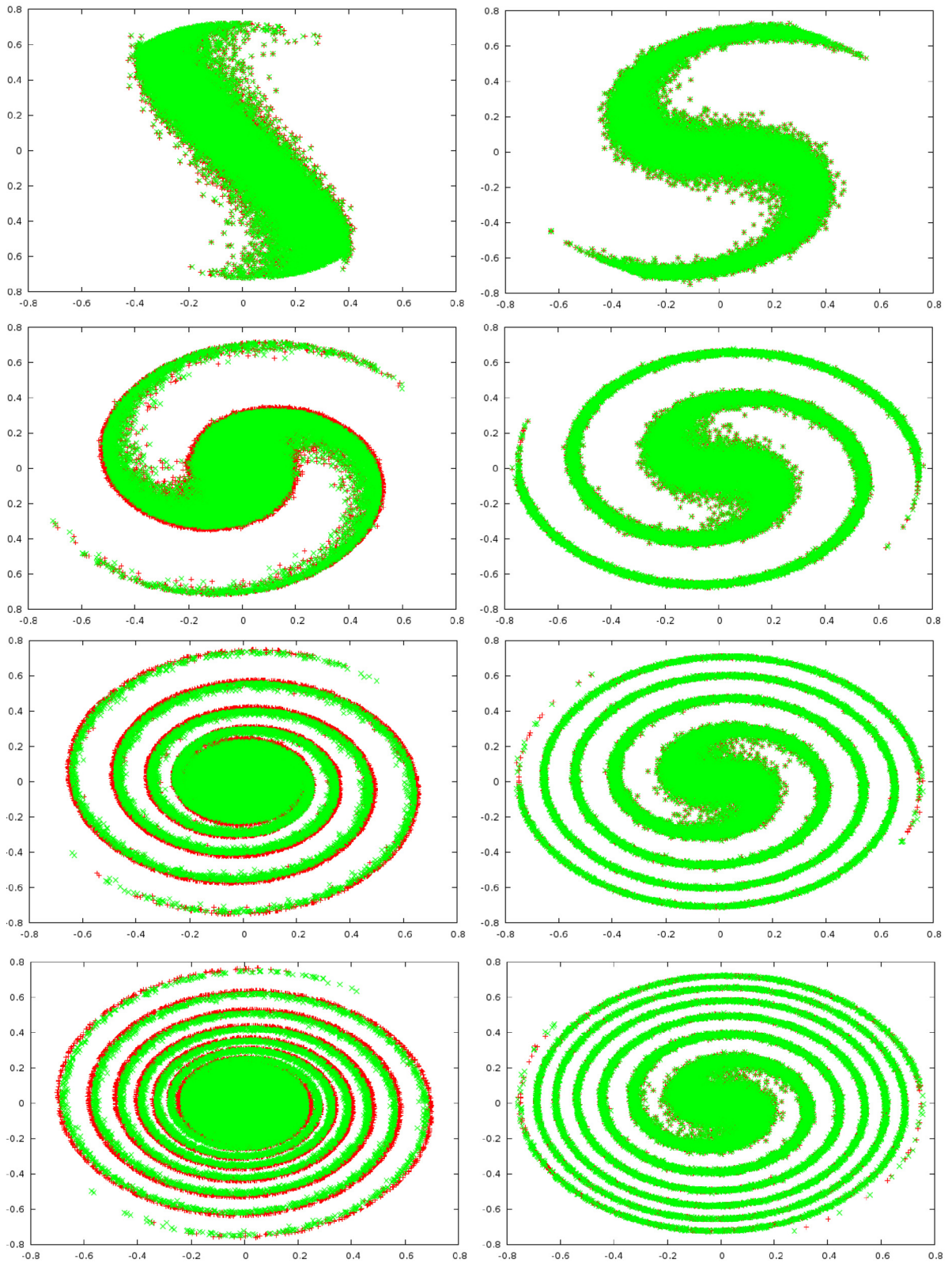
In conclusion, the scheme is able to produce quite accurate numerical solutions for stiff Vlasov-type equations for different small values of  $\varepsilon$  with the same computational cost, which is rather close to that of a reduced model.



**Fig. 8.** The case of  $E_\varepsilon = -r^3$  with  $\varepsilon = 0.001$ . In green: the reference solution. In red: the solution obtained with the modified ETD scheme (left) and with the improved ETD scheme (right). From top to bottom:  $t = 10$ ,  $t = 20$ ,  $t = 30$ , and  $t = 37$ . The time step used for the ETD schemes is  $\Delta t = 0.5$ . (For interpretation of the references to color in this figure, the reader is referred to the online version of this article.)



**Fig. 9.** Vlasov–Poisson test case with  $\varepsilon = 0.001$ . In red: the reference solution. In green: the solution obtained with the modified ETD scheme (left) and with the improved ETD scheme (right). From top to bottom:  $t = 15$ ,  $t = 30$ ,  $t = 45$ , and  $t = 60$ . The time step used for the ETD schemes is  $\Delta t = 0.5$ . (For interpretation of the references to color in this figure, the reader is referred to the online version of this article.)



**Fig. 10.** The cases of Vlasov–Poisson (left) and of Vlasov with  $E_\varepsilon = -r^3$  (right), when  $\varepsilon = 0.0001$ . In red: the reference solution. In green: the solution obtained with the improved ETD scheme. From top to bottom:  $t = 10$ ,  $t = 30$ ,  $t = 60$ , and  $t = 90$ . The time step for the ETD scheme is  $\Delta t = 0.5$ . (For interpretation of the references to color in this figure, the reader is referred to the online version of this article.)

## Appendix A. Calculus of periods for an undamped and undriven Duffing equation

Let  $\varepsilon > 0$  be some small parameter. The solution to the following Duffing-type equation

$$\begin{cases} R'' + \frac{1}{\varepsilon^2}R + \frac{1}{\varepsilon}R^3 = 0 \\ R(0) = r_0, \quad R'(0) = v_0/\varepsilon \end{cases} \quad (27)$$

can be written by using Jacobi elliptic functions. In this appendix, we obtain by an elementary calculus (see [7]) an approximated value of order 3 in  $\varepsilon$  for the period of the solution. In particular, we find out explicitly how the period  $T$  depend on the initial condition  $(r_0, v_0)$ .

The orbits of the solutions are level curves of the Hamiltonian function

$$\mathcal{H}(R, V) = \frac{R^2 + \varepsilon^2 V^2}{2\varepsilon^2} + \frac{R^4}{4\varepsilon}$$

where  $V$  stands for  $R'$ . As in Chapter V of reference [7], we denote the restoring force associated with the hard spring described by Eq. (27) by

$$g(R) = \frac{1}{\varepsilon^2}R + \frac{1}{\varepsilon}R^3$$

and by  $G(R) = \frac{R^2}{2\varepsilon^2} + \frac{R^4}{4\varepsilon}$  the potential energy. Since the function  $G$  has an absolute minimum in  $(0, 0)$  and no other critical point, all orbits are periodic. In addition, the closed orbit of a periodic solution is symmetric with respect to the  $R$ -axis and intersects it at two points,  $(a, 0)$  and  $(b, 0)$ , with  $a < b$ . Taking into account that the function  $g$  is odd, we have  $a = -b$  and, finally, we can determine an implicit formula for the period of the solution whose orbit intersect the  $R$ -axis in  $(b, 0)$ :

$$T = 4 \int_0^b \frac{1}{\sqrt{2h - 2G(R)}} dR \quad (28)$$

where  $h$  is the value of the Hamiltonian on the solution of system (27)

$$h = G(b) = \mathcal{H}(r_0, v_0) \quad (29)$$

Replacing  $h$  in (28), we obtain:

$$T = 4\varepsilon \int_0^1 \frac{1}{\sqrt{1 - R^2 + \frac{b^2\varepsilon}{2}(1 - R^4)}} dR \quad (30)$$

Then, solving Eq. (29) for  $b$  gives

$$b = \sqrt{\frac{1}{\varepsilon}(\sqrt{1 + 2\varepsilon(r_0^2 + v_0^2) + \varepsilon^2 r_0^4} - 1)} \quad (31)$$

which by Taylor expansion (recall  $0 < \varepsilon \ll 1$ ) results in

$$b^2\varepsilon = (r_0^2 + v_0^2)\varepsilon + \frac{r_0^4 - (r_0^2 + v_0^2)^2}{2}\varepsilon^2 + \mathcal{O}(\varepsilon^3) \quad (32)$$

Therefore, the parameter  $\alpha = \frac{b^2\varepsilon}{2}$  is small when  $\varepsilon \ll 1$ . Thus, by Taylor expansion to the second order of the function

$$\alpha \mapsto \frac{1}{\sqrt{1 - R^2 + \alpha(1 - R^4)}}$$

for each  $R$ , we obtain the following approximation of the period:

$$T = 2\pi\varepsilon - \frac{3\pi}{4}b^2\varepsilon^2 + \frac{57\pi}{128}b^4\varepsilon^3 + \mathcal{O}(\varepsilon^4) \quad (33)$$

By injecting (32) in (33), we finally obtain

$$\begin{aligned} T &= 2\pi\varepsilon - \frac{3\pi}{4}(r_0^2 + v_0^2)\varepsilon^2 \\ &\quad + \left( \frac{105\pi}{128}(r_0^2 + v_0^2)^2 - \frac{3\pi}{8}r_0^4 \right)\varepsilon^3 \\ &\quad + \mathcal{O}(\varepsilon^4) \end{aligned} \quad (34)$$

## References

- [1] E. Frénod, S.A. Hirstoaga, E. Sonnendrücker, An exponential integrator for a highly oscillatory Vlasov equation, *Discrete Contin. Dyn. Syst. Ser. S* 8 (1) (2015) 169–183.
- [2] E. Frénod, F. Salvarani, 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) (2009) 175–197.
- [3] C.K. Birdsall, A.B. Langdon, *Plasma Physics via Computer Simulation*, Institute of Physics, Bristol and Philadelphia, 1991.
- [4] M. Lutz, Application of Lie Transform Techniques for simulation of a charged particle beam, *Discrete Contin. Dyn. Syst. Ser. S* 8 (1) (2015) 185–221.
- [5] S.M. Cox, P.C. Matthews, Exponential time differencing for stiff systems, *J. Comput. Phys.* 176 (2002) 430–455.
- [6] M. Hochbruck, A. Ostermann, Exponential integrators, *Acta Numer.* 19 (2010) 209–286.
- [7] J.K. Hale, *Ordinary Differential Equations*, John Wiley, New York, 1969.