



**HAL**  
open science

# Parallel implementation of a relativistic semi-Lagrangian Vlasov-Maxwell solver

Mathieu Sarrat, Alain Ghizzo, Daniele del Sarto, Laurent Serrat

## ► To cite this version:

Mathieu Sarrat, Alain Ghizzo, Daniele del Sarto, Laurent Serrat. Parallel implementation of a relativistic semi-Lagrangian Vlasov-Maxwell solver. *The European Physical Journal D: Atomic, molecular, optical and plasma physics*, 2017, 71 (11), pp.271. 10.1140/epjd/e2017-80188-4 . hal-01791759

**HAL Id: hal-01791759**

**<https://hal.univ-lorraine.fr/hal-01791759v1>**

Submitted on 14 May 2018

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

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

# Parallel implementation of a relativistic semi-Lagrangian Vlasov-Maxwell solver

M. Sarrat<sup>1</sup>, A. Ghizzo<sup>1</sup>, D. Del Sarto<sup>1</sup>, L. Serrat<sup>1</sup>

August 23, 2017

1. Institut Jean Lamour, UMR 7198, Université de Lorraine, BP 70239 54506 Vandoeuvre les Nancy cedex

## Abstract

We describe the parallel implementation of a semi-lagrangian relativistic Vlasov ElectroMagnetic (VLEM) code for the numerical investigation of the dynamics of charged particle distribution in their self-consistent electromagnetic fields. This paper introduces the numerical solution of the Vlasov-Maxwell system in two spatial dimensions, and two or three momentum dimensions. Accuracy, stability, efficiency properties and the implementation of a new algorithm of charge conservation when solving Maxwell equations are discussed. The performances of the code are tested by studying the evolution of Weibel-type instabilities in the relativistic regime. Application to the coupling between Current Filamentation (CFI) and Two-Stream (TSI) instabilities is presented showing the importance of pair-wise vortex merging scenario in the saturation mechanism.

keywords: plasma physics, Vlasov equation, semi-lagrangian, splitting, relativistic regime, Weibel

## 1 Introduction

In a plasma, the interplay between the complex trajectories and the collective effects arising from the long-range nature of electromagnetic forces lead to a wide range of waves and instabilities. As a result, an enormous range of temporal and spatial scales involved. Although the fundamental laws that determine the behaviour of plasmas, such as Maxwell's equations and those of classical statistical mechanics, are well known, obtaining their solution under realistic conditions is a problem of extraordinary complexity. Most plasmas have nonlinear and kinetic effects of importance. The use of fundamental equations often retains the full nonlinear effects, and space charge and other collective effects can be included self-consistently by coupling charged particles to the field equations via the source terms.

At the most fundamental level, a collisionless plasma can be described by a Vlasov-Maxwell system for the distribution function within a six-dimensional (plus time) 3D3V phase space for each particle species. The kinetic simulation techniques used in plasma physics fall into two broad categories : Particle-In-Cell (PIC) models and Eulerian models. The most prominent kinetic approach is the PIC method pioneered by Dawson and others [1, 2, 3]. This method integrates the Vlasov equation in time by advancing marker particles, usually referred as super-particles, along a representative set of characteristics within the phase space. It basically involves a Lagrangian formulation : the Particle-In-Cell method can be regarded as a discretisation of the phase space in terms of a superposition of moving elements, each representing a « cloud » of physical particles called computational particles or super-particles, identified by the fact of being « close » one each other in the phase-space. PIC codes have proven to be useful in studying plasma dynamics even in 2D or 3D systems and complex geometries. In PIC codes, superparticle trajectories are computed from a field prescribed on a fixed grid whose typical size is of the order of the Debye length. This is indeed equivalent to compute the characteristic curves as given by the Vlasov equation. At the end of the time step, the charge of each particle is redistributed among the neighbouring grid points, allowing to solve the Maxwell equations. This method yields satisfying results with a relatively small number of particles. However, when a few particles per cell are used in such codes, a high level of numerical noise may appear and phase space structures become blurred, especially where the distribution function amplitude is small.

On the other hand, Eulerian Vlasov codes and Semi-Lagrangian (SL) Vlasov codes display an extremely low level of numerical noise. Then, they are able to describe wave-particle dynamics even in very low density phase-space regions (see for instance Refs. [4, 5, 6, 7, 8] for the study of Raman instabilities and Refs. [9, 10, 11] for two-dimensional plasmas). Unfortunately, these codes demand a stronger numerical effort than PIC codes because they require the discretisation of the whole phase space.

Another variant of the SL method, called the Cubic Interpolated Propagation (CIP) method, was introduced by Nakamura and Yabe in [12]. This method is based on a Hermite interpolation for which the gradients of the distribution function are also advanced along the characteristics. Another type of scheme for the Vlasov equation is the finite volume type method (or the flux balance method) which is based on the use of incoming and out-coming fluxes leading to mass conservation. First introduced by Fijalkow in [13] for the Vlasov equation, an improved version of the scheme called the Positive and Flux Conservative (PFC) method was proposed by Filbet et al in [14]. The PFC method is not only conservative but also preserves the positivity and the maximum value of the distribution function. In the general situation, the SL method does not conserve the total mass, but for linear advection with constant coefficients (i.e. independent of the advected quantities) the used of a centered approximation ensures the mass conservation. Moreover, methods using a smooth reconstruction (by Hermite or cubic spline) are less dissipative than the flux method, allowing a weak increase of the  $L^2$  norm. On the other hand, the PFC scheme is almost as accurate as the SL method using a cubic spline interpolation (see Filbet et al in ref. [15] for more details).

**A mathematical convergence of the SL scheme has been performed by Bermejo**

[42], and more recently by Falcone and Ferreti in Ref. [43]. The analysis of the scheme in term of accuracy and efficiency was made by Bartello and Thomas in [44]. More recently the convergence of the scheme for the one-dimensional Vlasov- Poisson system has been investigated by Besse in [45]. The SL method had already been investigated in weather and climate community by Staniforth and Côté in Ref. [46].

The basis for the SL model is the method first introduced by Cheng and Knorr in ref. [32], which introduces two new concepts : the method of time splitting, involving to separate the treatment of the Vlasov equation into several advections, and for each shift, the integration of the Vlasov equation along their characteristics. As the extremities of the characteristics do not coincide with the grid points, a reconstruction of the distribution function is required. This reconstruction is then made on a grid in the phase space using global cubic spline interpolations. The time splitting scheme can be efficiently parallelized by the use of a global transposition between each splitting step. Apart from this transposition that can be overlapped with computations, there is no need for communications between the processors. When several hundreds or more processes are targeted, the global transposition involves a large amount of data transfer and this feature becomes inefficient.

Recently, Crouseilles et al in [18] have proposed a new numerical scheme, locally performing the interpolation, specially designed to overcome this difficulty. Here we propose to use this technique in one-dimensional advections in the coordinate space without using the global transposition of the distribution function. To overcome the problem of global dependence inherent to the cubic spline interpolation, we have decomposed only the coordinate space into patches, each patch being devoting to one processor. The present 5D SL Vlasov code is parallelized via domain decomposition in the coordinate space, which involves the exchange of Hermite boundary conditions along adjacent processors.

This decomposition in the coordinate space was successfully used in gyrokinetic simulations for the study of Transport in Tokamaks in [16], leading to the development of the code GYSELA5D (for GYro SEmi LAgrangian code[17]). The code uses a “local” -i.e. on a patch- interpolation technique in  $r, \theta$ , global splines in  $\varphi$  and  $v_{\parallel}$  and the adiabatic invariance approximation in  $\mu$ . Here  $r$ ,  $\theta$  and  $\varphi$  denote the radial direction (in the cross section of the torus), the poloidal and the toroidal angles respectively, and  $v_{\parallel}$  is the velocity along the magnetic field. The code is fully parallelized and optimised to run on massive parallelized architecture. Today most of the Vlasov solvers have been developed for electrostatic plasmas (see Refs. [20, 21, 19]) and for electromagnetic ones are mostly restricted to non-relativistic systems (see [22, 23, 24, 25, 26, 30, 31]).

In this paper we present the first results of a two-dimensional parallelized SL Vlasov (VLEM) code in cartesian geometry, simulating the current filamentation instability (CFI) in the relativistic regime. The code is an extension of our previous works developed in [10, 29] to a fully parallelized hybrid OpenMP-MPI version. In this code, we use the splitting scheme and the integration of the Vlasov equation is made along the characteristics. We also use the local Hermite interpolation in the coordinate space and global (tensor) cubic spline interpolation in momentum space. The code exists in 1D3V, 2D2V and 2D3V geometries (the latter meaning two dimensions in the coordinate space and three-dimensional in the momentum space). The

changes proposed with respect to the previous Semi-Lagrangian version are twofold : the first is the introduction of a local Hermite interpolation scheme in the space domain (which uses only the continuity of the first derivation of  $f$  on patches in space), allowing to consider a domain decomposition in space. Secondly, we introduce a new charge conservation technique for solving the Maxwell equations.

The paper is organised as follows. In Sec. 2, we present the plasma geometry, the model equations and the main mathematical tools, namely the technique of time splitting, the global and local interpolation techniques. The new algorithm implementation for solving Maxwell equations preserving charge conservation is presented in Sec. 3. Sec. 4 is devoted to the global scheme of the VLEM2D2V code. Numerical tests relevant to the problem of the charge conservation are presented in Sec. 5. A more complex physical problem concerning electron beam instabilities is then investigated to check the efficiency of the local Hermite interpolation scheme. Performance of the scheme is discussed in Sec. 6 and conclusions are drawn in Sec. 7.

## 2 The Vlasov-Maxwell model

### 2.1 Model equations

The electron distribution function  $f(x, y, \mathbf{p}, t)$  obeys the relativistic Vlasov equation

$$\frac{\partial f}{\partial t} + \frac{\mathbf{p}}{m\gamma} \cdot \nabla_x f + e \left( \mathbf{E} + \frac{\mathbf{p} \times \mathbf{B}}{m\gamma} \right) \cdot \nabla_p f = 0 \quad (1)$$

where the Lorentz factor is given by

$$\gamma = \sqrt{1 + \frac{\mathbf{p}^2}{m^2 c^2}}. \quad (2)$$

The electromagnetic field  $(\mathbf{E}, \mathbf{B})$  obeys Maxwell equations :

$$\frac{\partial \mathbf{E}}{\partial t} - c^2 \text{rot} \mathbf{B} + \frac{\mathbf{J}}{\varepsilon_0} = 0, \quad (3)$$

$$\frac{\partial \mathbf{B}}{\partial t} + \text{rot} \mathbf{E} = 0, \quad (4)$$

$$\text{div} \mathbf{E} = \frac{\rho}{\varepsilon_0}, \quad (5)$$

$$\text{div} \mathbf{B} = 0, \quad (6)$$

where the electron current density  $\mathbf{J}$  and the charge density  $\rho$  ( $n_i$  being the fixed background ion density and  $e < 0$  the elementary charge of an electron) are given by :

$$\mathbf{J} = \iiint \frac{\mathbf{p}}{m\gamma} f d^3 \mathbf{p} \quad (7)$$

$$\rho = e \iiint f d^3\mathbf{p} - en_i \quad (8)$$

$\mathbf{J}$  and  $\rho$  verify together the continuity equation

$$\frac{\partial \rho}{\partial t} + \text{div} \mathbf{J} = 0 \quad (9)$$

which expresses the local charge conservation. This is automatically satisfied provided the Poisson equation (5) is initially verified. In that case, the Poisson law is satisfied for all times. In our numerical experiment, the physical quantities are normalised as follows : time  $t$ , space coordinate  $\mathbf{x} = (x, y)$  and momentum  $\mathbf{p} = (p_x, p_y, p_z)$  are respectively normalised to the inverse plasma frequency  $\omega_p^{-1}$ ,  $c\omega_p^{-1}$  and  $mc$ ;  $m$  being the electron rest mass and  $c$  the light velocity. The electric  $\mathbf{E}$  and magnetic  $\mathbf{B}$  fields are respectively normalised to  $m\omega_p c/e$  and  $m\omega_p/e$ . In a two-dimensional  $(x, y)$  problem, notice that components  $E_x$ ,  $E_y$  and  $B_z$  are decoupled from the components  $B_x$ ,  $B_y$  and  $E_z$ . Finally, we restrict our system to periodic boundary conditions in both  $x$  and  $y$  directions and we assume that the distribution function is zero at the boundaries of the momentum box.

The system presents a significant computational challenge, even with a relatively poorly sampled phase space. The Vlasov equation (1) is integrated in the original phase space by applying a splitting scheme (a method of fractional step) which consists in treating the convective term (the particle free streaming) and the acceleration-plus-rotation in  $\mathbf{p}$  term separately, as previously introduced by Knorr and Chen in Ref. [32] for electrostatic problems. While the scheme of time splitting fits with electrostatic problems, this technique nevertheless presents shortcomings that prevent its application for the treatment in the momentum space when relativistic effects have been taken into account. In order to avoid these difficulties it becomes necessary to use direct multidimensional (B-spline) full advection schemes in momentum that treat features of particle dynamics more accurately. In particular, global rotation in  $\mathbf{p}$  preserves the mass conservation of  $f$  since the norm of  $f$  is conserved during the rotation. Such a model had been successfully used in Refs. [27] and [28] for the study of laser-plasma interaction in the relativistic regime and for a one-dimensional system.

## 2.2 The Semi-Lagrangian method

The semi-Lagrangian method is based on the advective form of Vlasov equation (1). The distribution function solution of the Vlasov equation is constant along its characteristics. So, assuming  $f$  is known at time  $t_n = n\Delta t$ , the solution at time  $t_{n+1}$  is given by

$$f(\mathbf{x}, \mathbf{p}, t_{n+1}) = f(\mathbf{X}(t_n; \mathbf{x}, \mathbf{p}), \mathbf{P}(t_n; \mathbf{x}, \mathbf{p}), t_n) \quad (10)$$

where  $\mathbf{X}(t_n; \mathbf{x}, \mathbf{p})$  and  $\mathbf{P}(t_n; \mathbf{x}, \mathbf{p})$  stand for the solution of the differential system defining the characteristic curves which read

$$\frac{d\mathbf{X}}{dt} = \frac{\mathbf{P}(t)}{m\gamma(\mathbf{P}(t))} \quad (11)$$

and

$$\frac{d\mathbf{P}}{dt} = \mathbf{F}(\mathbf{X}(t), t) \quad (12)$$

where  $\mathbf{F} = e\mathbf{E} + \mathbf{P} \times \mathbf{B}/(m\gamma)$  and  $\gamma = \sqrt{1 + P^2(t)/(m^2c^2)}$  is the Lorentz factor. Thus, in the SL method,  $f$  is approximated using each grid point of the computational domain  $(\mathbf{x}, \mathbf{p})$  and is updated at each time step from its value at the origin of the characteristics  $\mathbf{X}, \mathbf{P}$ . The distribution function is then computed from the datas known on the grid by using a high order interpolation method.

### 2.3 The time split Semi-Lagrangian scheme for VLEM2D3V

Cubic spline interpolation seems to be a good compromise between accuracy (very small diffusivity in comparison to flux methods) and simplicity. Then, our strategy consists in getting a parallel version of the code, with numerical results as close as possible to a sequential version. We only decompose the spatial domain into patches, each patch being devoted to one processor. Here, no domain decomposition has been introduced in the momentum space. This technique is somewhat similar to the scheme of parallelization generally used in PIC codes, where only the spatial domain is decomposed in cells, each cell containing a number of super-particles. Such a strategy allows also a straightforward parallelization of the Maxwell equations and a possible extension in a 3D spatial system without difficulty. Each patch computes not only its own local cubic spline coefficients but also the 3D advection in momentum for the grid points which are spatially located on the patch. Inter-processor communications are reduced and communications are only performed between adjacent processors. Although in principle the use of local spline is possible in the momentum space in 3D, it would involve a relatively large amount of data exchange between the 27 neighbouring processors. Moreover, if a decomposition domain was adopted in the momentum space, since strong relativistic heating (as met in the Weibel-type instabilities) or strong particle acceleration processes (as met in the magnetic reconnection) are expected, the use of a local advection would require to transpose the distribution to follow such a particle heating in momenta over several neighbouring processors. No condition on the time step is required to control the shifts of the distribution function along the characteristics (except the standard Courant-Friedrichs-Lewy (CFL) condition imposed by the resolution of the Maxwell equations).

Starting from the Vlasov equation (1), we split it into one 3D advection in  $\mathbf{p}$  (or 2D for the considered case of a 2D2V plasma),

$$\frac{\partial f}{\partial t} + e \left( \mathbf{E} + \frac{\mathbf{p} \times \mathbf{B}}{m\gamma} \right) \cdot \nabla_{\mathbf{p}} f = 0, \quad (13)$$

and into two 1D advectons, namely,

$$\frac{\partial f}{\partial t} + \frac{p_x}{m\gamma} \frac{\partial f}{\partial x} = 0 \quad (14)$$

and

$$\frac{\partial f}{\partial t} + \frac{p_y}{m\gamma} \frac{\partial f}{\partial y} = 0. \quad (15)$$

It must be pointed out that in Eq.(13) the advection field depends on the advection variable  $\mathbf{p}$  via the Lorentz factor (2), forbidding a splitting of this equation into smaller ones because the three momentum variables are coupled by  $\gamma$ . Taken separately, steps (13), (14) and (15) can be exactly integrated along their characteristics.

When the electric field is zero, the characteristics of (13) can be exactly determined since a particle rotates at constant speed around the magnetic field line ( $\gamma$  is conserved). By considering the unit vector  $\mathbf{b} = \mathbf{B}/B$  associated with the magnetic field computed at time  $t_n = \Delta t$ , the rotation around  $\mathbf{B}$  allows us to determine  $\mathbf{P}^n = \mathfrak{R}_b \mathbf{P}^{n+1}$ , where  $\mathbf{P}^n$  is the momentum before the rotation. The rotation matrix is given by (using  $\varphi = \omega_c \Delta t$ ):

$$\mathfrak{R}_b = \begin{pmatrix} b_x^2 + (1 - b_x^2) \cos \varphi & b_x b_y (1 - \cos \varphi) - b_z \sin \varphi & b_x b_z (1 - \cos \varphi) + b_y \sin \varphi \\ b_x b_y (1 - \cos \varphi) + b_z \sin \varphi & b_y^2 + (1 - b_y^2) \cos \varphi & b_y b_z (1 - \cos \varphi) - b_x \sin \varphi \\ b_x b_z (1 - \cos \varphi) - b_y \sin \varphi & b_y b_z (1 - \cos \varphi) + b_x \sin \varphi & b_z^2 + (1 - b_z^2) \cos \varphi \end{pmatrix} \quad (16)$$

Taking into account the electric field components, the origin in momentum of a characteristic is given by :

$$P_x^n = -\frac{e\Delta t E_x}{2} + P_x^* \mathfrak{R}_{xx} + P_y^* \mathfrak{R}_{xy} + P_z^* \mathfrak{R}_{xz}, \quad (17)$$

$$P_y^n = -\frac{e\Delta t E_y}{2} + P_x^* \mathfrak{R}_{yx} + P_y^* \mathfrak{R}_{yy} + P_z^* \mathfrak{R}_{yz} \quad (18)$$

and

$$P_z^n = -\frac{e\Delta t E_z}{2} + P_x^* \mathfrak{R}_{zx} + P_y^* \mathfrak{R}_{zy} + P_z^* \mathfrak{R}_{zz}, \quad (19)$$

where

$$\mathbf{P}^* = \mathbf{P}^{n+1} - \frac{e\Delta t}{2} \mathbf{E}. \quad (20)$$

The advection due to the electric field is symmetrised with respect to the rotation.

A formal solution of Eqs.(13), (14) and (15) is given by the following steps :

(i) Determine the field component ( $E, B$ ) using Maxwell equations (3) to (6) and the corresponding currents (see Sec. 4).

Integrate Eq. (13) at time  $t_n$  over a global time step  $\Delta t$ , using Eqs. (17) - (19):

$$f^P(x, y, \mathbf{p}) = f^n(x, y, \mathbf{P}^n(t_n; \mathbf{x}, \mathbf{p})) \quad (21)$$

Eq. (21) allows to exactly conserve the charge, since the advection is made globally. Another advantage of the method is the possibility to treat strong advection without limitation on the time step and due to the communication between processors, since the operation is made locally on the same processor.

From a numerical point of view, the advection can be parallelized using OpenMP. Once we have followed the characteristic curves backward, we have to evaluate  $f$  at the end points of the characteristic curves which do not generally coincide with the mesh points, where  $f$  is known instead. In Eq.(21), the distribution function is calculated by a tensor product of B-spline in the framework of the interpolation scheme.



(ii) The shifts in  $x$  and  $y$  are performed using Eqs (14) and (15) by alternating the advectons (using an  $\frac{X}{2}Y\frac{X}{2}$  operator sequence). Thus we integrate Eqs. (14) and (15) along their characteristics given by

$$f^*(x, y, \mathbf{p}) = f^P \left( x - \frac{p_x}{m\gamma} \frac{\Delta t}{2}, y, \mathbf{p} \right), \quad (22)$$

$$f^{**}(x, y, \mathbf{p}) = f^* \left( x, y - \frac{p_y}{m\gamma} \Delta t, \mathbf{p} \right), \quad (23)$$

and

$$f^{n+1}(x, y, \mathbf{p}) = f^{**} \left( x - \frac{p_x}{m\gamma} \frac{\Delta t}{2}, y, \mathbf{p} \right). \quad (24)$$

The whole scheme is of second order accurate in time (see Ref.[11]).

Summarising, denoting the standard advection in  $x, y$ , over a time step  $\Delta t$  by the operators  $X$  and  $Y$  ( $\frac{X}{2}$  corresponding indeed to a shift in  $x$  over half a time step) and the shift in momentum by  $P$ , the global scheme can be condensed in the sequence  $\frac{X}{2}Y\frac{X}{2}P\frac{Y}{2}\frac{X}{2}$ . It must be pointed out that, besides the first sequence  $\frac{X}{2}Y\frac{X}{2}$ , the scheme is equivalent to  $P\frac{X}{2}Y\frac{X}{2}$ . Moreover the extension of the numerical scheme to a 3D system is straightforward just by making a third spatial advection (in  $z$ ) by modifying the sequence of shifts in  $\frac{X}{2}Y\frac{X}{2}Z\frac{Y}{2}\frac{X}{2}$ . Here the domain decomposition is made in space using MPI. Such a decomposition can also be used for solving the Maxwell equations.

**We have used global B-spline for the treatment of the 3D advection for  $\mathbf{p}$  ( see Refs. [8, 29] for more details).** A shared memory approach is used (using OpenMP) : each worker thread follows the trajectory of a set of grid points whose indices are in a predefined sub range. Here a fine grained approach has been adopted using OpenMP parallel for directive. The OpenMP parallel for directive or parallel\_reduce algorithms can be used to automatically distribute the do loop workload across the threads. In this case, the  $\mathbf{p}$ -advection part of the code remains local all the time (as in a sequential form) with the exception of the work-sharing parallel sections related to the loops). A first version of the 6D Vlasov code, using only OpenMP and this 3D version of the B-spline interpolation for 3D advectons has been already realised by Coulaud et al in Ref. [29].

## 2.4 Local spline interpolation

Even if the cubic spline approach is commonly used in order to solve the Vlasov equation, it remains a global method since the computation requires the value of the distribution function on the whole domain. This can become an inconvenient from a parallelization point of view. In this section, we present the interpolation technique based on a local one-dimensional cubic spline method (see N. Crouseilles et al for more details in Ref. [18]). This scheme avoids the globality and is based on a domain decomposition. Thus we decompose the configuration space, along the  $x$  and  $y$  directions, in several cells, each cell being devoted to one processor (using MPI). The strategy is based on adapted boundary conditions which make possible a reconstructed solution on the global space domain even on the cell boundaries.

The programming structure is thus simplified and load-balancing of all the processors is observed as the result of the Eulerian approach where data stay in place (with respect to the Lagrangian approach). One cell computes its own local cubic spline coefficients by solving linear systems and Hermite boundary conditions are imposed on all the cells to reconstruct a smooth numerical solution in the whole configuration space.

Our strategy consists in getting a fully parallelized hybrid (MPI/ OpenMP) version of the code. This is possible thanks to the splitting scheme in space which allows the use of only one- dimensional shift of  $f$  in space.

Each spatial direction, say  $x$  to simplify, is decomposed into several sub-domains denoted by  $[x_0, x_N]$ , each sub-domain being assigned to a given processor.

The projection  $S(x)$  of  $f$  onto the cubic spline basis can be written in the following form :

$$f(x) \simeq S(x) = \sum_{\nu=1}^{N+1} \eta_{\nu} B_{\nu}(x) \quad (25)$$

where  $B_{\nu}(x)$  is a cubic B-spline coefficient (see Ref. [18] for more details). The interpolation spline  $s$  is uniquely determined by  $N + 1$  interpolation conditions

$$f(x_i) = S(x_i) \quad \text{for } i = 0, \dots, N \quad (26)$$

and by the Hermite boundary conditions

$$f'(x_0) = S'(x_0) \quad \text{and} \quad f'(x_N) = S'(x_N) \quad (27)$$

at both ends of the chosen interval.

Denoting by  $\Delta x = (x_N - x_0)/(N + 1)$  the spatial grid, the only cubic B-splines not vanishing at point  $x_i$  are  $B_{i\pm 1}(x_i) = 1/6$  and  $B_i(x_i) = 2/3$ . Hence Eq. (26) gives

$$f(x_i) = \frac{1}{6} (\eta_{i-1} + 4\eta_i + \eta_{i+1}) \quad (28)$$

for  $i$  varying from 0 to  $N$ . We recover the standard tridiagonal matrix, except for the first and last lines. The Hermite boundary conditions (27) become

$$f'(x_0) \simeq S'(x_0) = \frac{-\eta_{-1}}{2\Delta x} + \frac{\eta_1}{2\Delta x} \quad (29)$$

and

$$f'(x_N) \simeq S'(x_N) = \frac{-\eta_{N-1}}{2\Delta x} + \frac{\eta_{N+1}}{2\Delta x}. \quad (30)$$

A little algebra allows us to determine the coefficients  $\eta_{\nu}$  in the form of an array of type  $\boldsymbol{\eta} = (\eta_{-1}, \eta_0, \dots, \eta_{N+1})^T$ , which is the solution of the  $(N + 3) \times (N + 3)$  system of type  $[A] \boldsymbol{\eta} = \mathbf{F}$  where  $\mathbf{F} = (f'(x_0), f(x_0), \dots, f(x_N), (f'(x_N)))^T$  and where the matrix  $[A]$  is given by

$$[A] = \frac{1}{6} \begin{pmatrix} \frac{-3}{\Delta x} & 0 & \frac{3}{\Delta x} & 0 & \cdots & 0 \\ 1 & 4 & 1 & 0 & \ddots & \vdots \\ 0 & 1 & 4 & 1 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & \ddots & 0 \\ \vdots & \cdots & 0 & 1 & 4 & 1 \\ 0 & \cdots & 0 & -\frac{3}{\Delta x} & 0 & \frac{3}{\Delta x} \end{pmatrix} \quad (31)$$

The LU decomposition of the matrix  $[A]$  is performed once and for all. At each time step, a spline interpolation needs to be computed solving  $[L][U]\boldsymbol{\eta} = \mathbf{F}$  into two steps. First, the solution of  $[L]\boldsymbol{\varphi} = \mathbf{F}$  and then the solution of  $[U]\boldsymbol{\eta} = \boldsymbol{\varphi}$ . Several remarks must be pointed out.

(i) Some communications between processors have to be managed in a suitable way. Indeed, the method can introduce a limitation on the time step because a shifted piece of  $f$  is able to reach another subdomain during the advection. The corresponding data must be forwarded to the processor which controls the subdomain where the piece of  $f$  is advected.

In general, in this scheme a condition on the time step can usually allow us to control the shifts so that the communications are performed only between adjacent processors. It can become a strong constraint when the scheme is applied, for instance, for the treatment of velocities due to the strong level of particle acceleration in the relativistic regime. For the different shifts in space, the situation is somewhat different since the particle velocities are bounded by the relativistic constraints ( $v_x/c = p_x/(m\gamma c) < 1$ ) allowing thus to restrict communications only to the two first neighbouring processors in  $x$ .

(ii) Because we want the local determination of cubic spline coefficients to recover in the best way an usual solution on the global domain, it is necessary to approximate the derivatives of  $f$  in an accurate way on the global domain. The derivatives of  $f$  (here noted  $f'(x_0)$  and  $f'(x_N)$ ) connect adjacent sub-domains and the accuracy of the local spline scheme is thus very sensitive to those data. In particular low-order finite difference approximation may lead to strong oscillations at the interface region between two adjacent processors, and these oscillations may propagate to the whole domain. On the contrary, the approximation of the derivatives using cubic splines, if adequately made, allows to obtain a robust scheme.

(iii) It has been shown in Ref. [18] that it was possible to build an accurate approximation of the derivatives  $f'(x_0)$  and  $f'(x_N)$  using Eqs. (28), (29) and (30) leading to

$$S'(x_i) \simeq \sum_{j=-10}^{+10} \tilde{\omega}_j f_{i+j} = \sum_{j=-10}^{-1} \tilde{\omega}_j^- f_{i+j} + \sum_{j=1}^{+10} \tilde{\omega}_j^+ f_{i+j} \quad (32)$$

where  $\tilde{\omega}_0 = 0$  and  $\tilde{\omega}_j^+ = -\tilde{\omega}_j^-$ . The coefficients  $\tilde{\omega}_j$  are given in table I.

$j$	$\tilde{\omega}_j^-$
-10	$0.22143 \times 10^{-5}$
-9	$-1.77144 \times 10^{-5}$
-8	$7.9715 \times 10^{-5}$
-7	$-3.01146 \times 10^{-4}$
-6	$1.11379 \times 10^{-3}$
-5	$-4.1452 \times 10^{-3}$
-4	0.01546474
-3	-0.05771377
-2	0.21539034
-1	-0.80384758

Table 1: Values of the different coefficients “local”  $\tilde{\omega}_j$  calculations in the  $x$  and  $y$  directions.

### 3 Maxwell equations

#### 3.1 The standard Yee algorithm

We use the Yee algorithm [33] in order to solve Maxwell equations and we limit the treatment to the components  $E_x$ ,  $E_y$  and  $B_z$ . In the case of a 2D system, the  $(E_x, E_y, B_z)$  and  $(E_z, B_x, B_y)$  components are decoupled, so the second set of the fields can be integrated in an analogous way. Thus we focus here on the treatment of the  $(E_x, E_y, B_z)$  field components. In this algorithm, the continuous derivatives in space and time are approximated by second-order accurate, two-point centred difference forms. A leapfrog integration in time is used to update the fields. The resulting set of equations is :

$$\frac{\partial E_x}{\partial t} = c^2 \frac{\partial B_z}{\partial y} - \frac{J_x}{\varepsilon_0} \quad (33)$$

$$\frac{\partial E_y}{\partial t} = -c^2 \frac{\partial B_z}{\partial x} - \frac{J_y}{\varepsilon_0} \quad (34)$$

$$\frac{\partial B_z}{\partial t} = \frac{\partial E_x}{\partial y} - \frac{\partial E_y}{\partial x} \quad (35)$$

We now consider the discretization of Eqs. (33) to (35).

The  $E_x$  component is located at half  $x$  and integer  $y$  grid points, i.e.  $(i + \frac{1}{2}, j)$  while the  $E_y$  component is located at  $(i, j + \frac{1}{2})$ . The magnetic field component  $B_z$  is located at  $(i + \frac{1}{2}, j + \frac{1}{2})$ . These grid points are chosen to accommodate the interleaved leapfrog algorithm that makes up the Yee method. The discretized versions of the Eqs. (33), (34) and (35) are :

$$E_{x, i+\frac{1}{2}, j}^{n+1} = E_{x, i+\frac{1}{2}, j}^n + \frac{c^2 \Delta t}{\Delta y} \left( B_{z, i+\frac{1}{2}, j+\frac{1}{2}}^{n+\frac{1}{2}} - B_{z, i+\frac{1}{2}, j-\frac{1}{2}}^{n+\frac{1}{2}} \right) - \frac{\Delta t}{\varepsilon_0} J_{x, i+\frac{1}{2}, j}^{n+\frac{1}{2}} \quad (36)$$

$$E_{y, i, j+\frac{1}{2}}^{n+1} = E_{y, i, j+\frac{1}{2}}^n - \frac{c^2 \Delta t}{\Delta x} \left( B_{z, i+\frac{1}{2}, j+\frac{1}{2}}^{n+\frac{1}{2}} - B_{z, i-\frac{1}{2}, j+\frac{1}{2}}^{n+\frac{1}{2}} \right) - \frac{\Delta t}{\varepsilon_0} J_{y, i, j+\frac{1}{2}}^{n+\frac{1}{2}} \quad (37)$$

$$B_{z\ i+\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}} = B_{z\ i+\frac{1}{2},j+\frac{1}{2}}^{n-\frac{1}{2}} + \frac{\Delta t}{\Delta y} \left( E_{x\ i+\frac{1}{2},j+1}^n - E_{x\ i+\frac{1}{2},j}^n \right) - \frac{\Delta t}{\Delta x} \left( E_{y\ i+1,j+\frac{1}{2}}^n - E_{y\ i,j+\frac{1}{2}}^n \right) \quad (38)$$

which exhibit the following Courant-Friedrichs-Lewy (CFL) condition on the time step

$$c^2 \Delta t^2 (\Delta x^{-2} + \Delta y^{-2}) \leq 1 \quad (39)$$

The discrete Poisson law, at time  $t_n = n\Delta t$  writes :

$$\frac{\rho_{ij}^n}{\varepsilon_0} = \frac{E_{x\ i+\frac{1}{2},j}^n - E_{x\ i-\frac{1}{2},j}^n}{\Delta x} + \frac{E_{y\ i,j+\frac{1}{2}}^n - E_{y\ i,j-\frac{1}{2}}^n}{\Delta y} \quad (40)$$

Taking the discrete derivative  $\partial_x$  of Eq. (36) plus the discrete derivative  $\partial_y$  of (37) leads to

$$\begin{aligned} & \frac{E_{x\ i+\frac{1}{2},j}^{n+1} - E_{x\ i-\frac{1}{2},j}^{n+1}}{\Delta x \Delta t} + \frac{E_{y\ i,j+\frac{1}{2}}^{n+1} - E_{y\ i,j-\frac{1}{2}}^{n+1}}{\Delta y \Delta t} - \frac{E_{x\ i+\frac{1}{2},j}^n - E_{x\ i-\frac{1}{2},j}^n}{\Delta x \Delta t} - \frac{E_{y\ i,j+\frac{1}{2}}^n - E_{y\ i,j-\frac{1}{2}}^n}{\Delta y \Delta t} = \\ & - \frac{J_{x\ i+\frac{1}{2},j}^{n+\frac{1}{2}} - J_{x\ i-\frac{1}{2},j}^{n+\frac{1}{2}}}{\varepsilon_0 \Delta x} - \frac{J_{y\ i,j+\frac{1}{2}}^{n+\frac{1}{2}} - J_{y\ i,j-\frac{1}{2}}^{n+\frac{1}{2}}}{\varepsilon_0 \Delta y} \end{aligned} \quad (41)$$

Thus if we replace the components of the fields at time  $t_n$  by  $\rho_{ij}^n$ , we see that the Poisson condition (60) is also satisfied at later time  $t_{n+1}$ , provided that :

$$\frac{\rho_{ij}^{n+1} - \rho_{ij}^n}{\Delta t} + \frac{J_{x\ i+\frac{1}{2},j}^{n+\frac{1}{2}} - J_{x\ i-\frac{1}{2},j}^{n+\frac{1}{2}}}{\varepsilon_0 \Delta x} + \frac{J_{y\ i,j+\frac{1}{2}}^{n+\frac{1}{2}} - J_{y\ i,j-\frac{1}{2}}^{n+\frac{1}{2}}}{\varepsilon_0 \Delta y} = 0 \quad (42)$$

which is nothing else than the discrete expression of the charge conservation law (9).

### 3.2 Charge conserving scheme for the Maxwell solver

It is well known that the numerical solution of the Vlasov-Maxwell system leads to charge and current densities which do not verify the discrete charge conservation.

We present a new charge conserving scheme. We will give a way to compute the current, which leads to charge conserving numerical scheme. The scheme is formulated for shifted distribution values moving less than one cell  $\Delta x$  per time step, an assumption verified since the velocity of the considered particle is found less than the light velocity, as required by the relativistic model. The scheme can be interpreted as an extension of the Esirkepov's method [34] for the backward (or even forward) SL scheme. In PIC codes, due to the linearity of the conservation equation (42), Esirkepov introduces a current density  $e \frac{d\mathbf{x}}{dt} \mathbf{W}$  associated to the motion of a super-particle. However in the SL approach, the spline interpolation function describes the shifted particle distribution function by introducing an interpolation function

a role equivalent which plays an equivalent role to the form factor used in PIC codes. The spline interpolation function  $S(x, y)$  estimated, at point  $(x, y)$ , is defined by Eqs. (??) and (25). In Esirkepov method the scheme is formulated for super-particles moving less than one cell per time step where the form-factor is also the charge density of the super-particles.

Thus, using the definition of the variation of the current density given by the quantity  $e \overline{\frac{dx}{dt} \mathbf{W}}$ , we can write :

$$J_{x\ i+\frac{1}{2},j}^{n+\frac{1}{2}} - J_{x\ i-\frac{1}{2},j}^{n+\frac{1}{2}} = -e \int d^2 p v_x W_x = -e \overline{\frac{dx}{dt} W_x} \quad (43)$$

$$J_{y\ i,j+\frac{1}{2}}^{n+\frac{1}{2}} - J_{y\ i,j-\frac{1}{2}}^{n+\frac{1}{2}} = -e \int d^2 p v_y W_y = -e \overline{\frac{dy}{dt} W_y} \quad (44)$$

According to the continuity equation (42), we can write :

$$W_x + W_y = S(x^{n+1}, y^{n+1}) - S(x^n, y^n) = S(x + \delta x, y + \delta y) - S(x, y) \quad (45)$$

Here  $(\delta x, \delta y)$  denotes the two-dimensional shift of the “particle”, located on the grid point in the backward SL model and  $S$  is the Hermite interpolation function defined in the previous section. In Esirkepov method, the super-particle trajectory in a time step is treated from a statistical point of view by calculating the expression of the quantities  $W_x$  and  $W_y$  as linear combinations of the different shifts in  $\delta x$  and  $\delta y$  (or, equivalently, as a function of quantities  $S(x + \delta x, y + \delta y)$ ,  $S(x + \delta x, y)$ ,  $S(x, y + \delta y)$  and  $S(x, y)$ ).

Starting from the characteristics of the Vlasov equation, it is possible to decompose the solution (45) in the following form :

$$W_x = S\left(x + \frac{\delta x}{2}, y\right) - S(x, y) + S(x + \delta x, y + \delta y) - S\left(x + \frac{\delta x}{2}, y + \delta y\right) \quad (46)$$

$$W_y = S\left(x + \frac{\delta x}{2}, y + \delta y\right) - S\left(x + \frac{\delta x}{2}, y\right) \quad (47)$$

Note that by summation of Eq. (46) to (47) we recover (45). In (46) the quantity  $S(x + \delta x/2, y) - S(x, y)$  therefore corresponds to the first shift in  $x$  over half a time step  $\Delta t/2$  (after the sequence  $\frac{X}{2}$ ), while the quantity  $S(x + \delta x, y + \delta y) - S(x + \delta x/2, y + \delta y)$  is related to the second shift in  $x$  (i.e. over half a time step  $\Delta t/2$  after the global sequence  $\frac{X}{2} Y \frac{X}{2}$ ). In Eq. (47) the second member corresponds now to the shift of the “particle” in  $y$  (after the sequence  $\frac{X}{2} Y$ ). It is then possible to accurately evaluate the current corrections needed to ensure the charge conservation, using the correction of the current fluxes. We find :

$$\begin{aligned} e \overline{\frac{dx}{dt} W_x} &= e \int d^2 p \frac{v_x}{2} \left[ S\left(x + \frac{\delta x}{2}, y\right) - S(x, y) + S(x + \delta x, y + \delta y) - S\left(x + \frac{\delta x}{2}, y + \delta y\right) \right] \\ &= \frac{1}{2} (J_{x\ i,j}^* - J_{x\ i,j}^n + J_{x\ i,j}^{n+1} - J_{x\ i,j}^{**}) \end{aligned} \quad (48)$$

$$e \frac{dy}{dt} W_y = e \int d^2 p v_y \left[ S \left( x + \frac{\delta x}{2}, y + \delta y \right) - S \left( x + \frac{\delta x}{2}, y \right) \right] = J_{y i, j}^{**} - J_{y i, j}^* \quad (49)$$

In Eq. (48) the integration is made on the characteristics  $x - v_x t/2 = \text{const}$  while in (49) we have used the characteristics  $y - v_y t = \text{const}$ . Thus, from relation (43), the current  $J_{x i+\frac{1}{2}, j}^{n+\frac{1}{2}}$  to be injected in the Faraday law, written in its discrete form (36), is  $J_{x i+\frac{1}{2}, j}^{n+\frac{1}{2}} = J_{x i-\frac{1}{2}, j}^{n+\frac{1}{2}} - e \frac{dx}{dt} W_x$ . This can be approximated by

$$J_{x i+\frac{1}{2}, j}^{n+\frac{1}{2}} = \frac{1}{4} (J_{x i, j}^n + J_{x i-1, j}^n + J_{x i, j}^{n+1} + J_{x i-1, j}^{n+1}) - \frac{1}{2} (J_{x i, j}^* - J_{x i, j}^n + J_{x i, j}^{n+1} - J_{x i, j}^{**}) \quad (50)$$

In the same manner, the  $y$ - component of the current, say  $J_{y i, j+\frac{1}{2}}^{n+\frac{1}{2}}$ , is calculated from Eqs. (44), (47) and (49), leading to

$$J_{y i, j+\frac{1}{2}}^{n+\frac{1}{2}} = \frac{1}{4} (J_{y i, j}^n + J_{y i, j-1}^n + J_{y i, j}^n + J_{y i, j-1}^n) - (J_{y i, j}^{**} - J_{y i, j}^*) \quad (51)$$

Note that the numerical scheme is equivalent to that proposed in [35] for the numerical resolution of the Vlasov equation using a conservative scheme based on the use of the flux of  $f$  across a control cell.

## 4 The Semi-Lagrangian Vlasovian code : VLEM

### 4.1 The (old) SL global approach: the reference code

We start with the “global” SL Vlasov-Maxwell scheme, here named the “global scheme”. The system is solved by making use of the standard SL algorithm [10, 11] and the reduced 2D2V Vlasov equation reads as

$$\frac{\partial f}{\partial t} + \frac{p_x}{m\gamma} \frac{\partial f}{\partial x} + \frac{p_y}{m\gamma} \frac{\partial f}{\partial y} + e \left( E_x + \frac{p_y B_z}{m\gamma} \right) \frac{\partial f}{\partial p_x} + e \left( E_y - \frac{p_x B_z}{m\gamma} \right) \frac{\partial f}{\partial p_y} = 0, \quad (52)$$

self-consistently coupled to the Maxwell equations (36), (37) and (38). The global Vlasov scheme is solved numerically using the time splitting scheme. Full tridiagonal solvers (required for computing 1D cubic splines or B-splines coefficients) are well-known to have not a very good scalability when performed on multiple processors. A 1D band decomposition over  $x$  or  $p_x$  direction is then performed, even though this would involve global MPI communications through the global transposition of  $f$ . For instance, in the case of the 1D  $x - p_x$  system, the spatial variable is kept fixed while the other variable  $p_x$  is decomposed and the spline interpolation is performed in a global way, without using the parallel version of the cubic spline interpolation technique, previously presented in Sec. 2.5.

In the case of the 2D geometry the field components are shifted in time because the current densities are only known at time  $t_n$ :  $E_x$  and  $E_y$  are discretized as  $E_{x i+\frac{1}{2}, j}^{n+\frac{1}{2}}$  and  $E_{y i, j+\frac{1}{2}}^{n+\frac{1}{2}}$  while

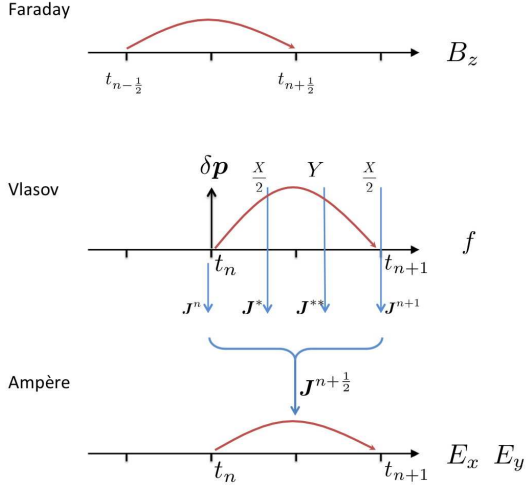


Figure 1: Numerical scheme used in the VLEM2D3V solver in the local version (with local Hermite interpolation). The computation of the different current densities (after the corresponding shifts) are indicated by blue arrows (middle panel). The top panel corresponds to the computation of the magnetic field, while the bottom panel is related to the calculation for the electric field. The Poisson equation is only used at the initial time.

$B_z$  corresponds to  $B_{z\ i+\frac{1}{2},j+\frac{1}{2}}^{n+1}$ . We denote the shifts of the distribution function  $f$ , over half a time step, along the direction  $x$  and  $y$  by the operators  $\frac{X(f)}{2}$  and  $\frac{Y(f)}{2}$  respectively, and the shift in the momentum space over a time step by  $P(f)$ . The integration of the Vlasov equation is given by the sequence of shifts (for a time step)  $\frac{X(f)}{2} \frac{Y(f)}{2} \mathbf{P}(f^t) \frac{Y(f)}{2} \frac{X(f)}{2}$  and a transposition of the global matrix is required before and after the shift in  $\mathbf{p}$ . Here  $f^t$  is the transposed distribution function obtained by application of the operator  $T$  on  $f$ . This symmetrization technique can also be applied in the case of the global 1D2V Vlasov solver. After this, the system is transposed before applying the shift in the  $p_x$  direction. MPI-based communications are only needed when calculating the charge density and the current density. Because the model is built on the continuity of the two first derivatives in space, it can be used as a Reference solution to check the accuracy of the VLEM2D2V solver.

## 4.2 The (new) local version of the SL Vlasov-Maxwell solver: VLEM

The VLEM2D2V (or its extension to VLEM2D3V) code is parallelized by using a regular spatial domain decomposition in the  $x$  and  $y$  directions. Each processor involved in the simulation is in charge of one subdomain. The information shared between the threads of neighbouring sub-domains combines OpenMP with MPI. Thus, in the local version, the model parallelizes computed-bound loops with OpenMP, and inter-processor communications are handled by MPI. Specifically, the partitioning of a given subdomain is done so that the sizes of the sub-domains span a significant range of  $16^2$  grid points in spatial grid (because at least



10 points are necessary to calculate the derivatives of  $f$  in an accurate way).

The Vlasov equation (52) (or its equivalent 2D3V version) is solved following the different sequences shown in Fig. 1. We list the different steps below.

- Step 0 : At the initial time, the field components  $E_x^*$  and  $E_y^*$  are solved using the Poisson equation and then we compute the quantities :

$$E_{x\ i+\frac{1}{2},j}^n = \frac{1}{2} (E_{x\ i,j}^* + E_{x\ i+1,j}^*) \quad (53)$$

$$E_{y\ i,j+\frac{1}{2}}^n = \frac{1}{2} (E_{y\ i,j}^* + E_{y\ i,j+1}^*) \quad (54)$$

- Step 1 : Compute the  $z$ -component of the magnetic field  $B_{z\ i+\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}}$  from Eq. (38). Since we need to know the electromagnetic field at the grid points  $(i, j)$  to compute the 2D advection in momentum, we approximate the magnetic field by the quantity  $B_{z\ i,j}^* = (B_{z\ i,j}^{n-\frac{1}{2}} + B_{z\ i,j}^{n+\frac{1}{2}}) / 2$  where

$$B_{z\ i,j}^{n+\frac{1}{2}} = \frac{1}{4} \left( B_{z\ i+\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}} + B_{z\ i-\frac{1}{2},j+\frac{1}{2}}^{n+\frac{1}{2}} + B_{z\ i+\frac{1}{2},j-\frac{1}{2}}^{n+\frac{1}{2}} + B_{z\ i-\frac{1}{2},j-\frac{1}{2}}^{n+\frac{1}{2}} \right), \quad (55)$$

while the  $x$  and  $y$  components of the electric field are computed using

$$E_{x\ i,j}^* = \frac{1}{2} \left( E_{x\ i+\frac{1}{2},j}^n + E_{x\ i-\frac{1}{2},j}^n \right) \quad (56)$$

and

$$E_{y\ i,j}^* = \frac{1}{2} \left( E_{y\ i,j+\frac{1}{2}}^n + E_{y\ i,j-\frac{1}{2}}^n \right). \quad (57)$$

Finally we compute the current density  $\mathbf{J}_{i,j}^n$  from the data of  $f$  using (7).

- Step 2 : Perform the 2D (or 3D) rotation-plus-advection in  $\mathbf{p}$  over a time step  $\Delta t$  using Eq. (21).
- Step 3 : Perform the shift over half a time step  $\frac{\Delta t}{2}$  of  $f$  in the  $x$  direction according to (22) (denoted in Fig. 1 by the operator  $\frac{X}{2}$ ) and compute the corresponding current density  $J_{x\ i,j}^* = e \int d^2 p v_x S(x + \delta x/2, y)$  by integrating the shifted distribution function  $f^*(x, y, \mathbf{p})$  using (7).
- Step 4 : Perform the shift in  $y$  over a global time step  $\Delta t$  using (23) and compute the current density  $J_{y\ i,j}^{**} = e \int d^2 p v_y [S(x + \delta x/2, y + \delta y)]$  by integrating the shifted distribution function  $f^{**}(x, y, \mathbf{p})$
- Step 5 : Repeat the step 3 to obtain  $f_{i,j}^{n+1}$  according to (24) and compute the current density  $J_{x\ i,j}^{n+1}$ .

- Step 6 : Compute the electric field components using the Maxwell's equations (36) and (37) by using the “exact” current density components  $J_{x\ i+\frac{1}{2},j}^{n+\frac{1}{2}}$  and  $J_{y\ i,j+\frac{1}{2}}^{n+\frac{1}{2}}$  from relations (50) and (51).

As indicated in Fig. 1, the current densities  $\mathbf{J}^*$  and  $\mathbf{J}^{**}$  are computed just after the sequences of shifts  $\frac{X}{2}$  and  $\frac{X}{2}Y$  respectively.

It must be pointed out that, in the 1D case, the electrostatic component of the electric field can be exactly determined, using the Poisson equation, at the end of the sequence  $\frac{X}{2}\frac{Y}{2}$ . Such a model provides then the reference code with respect to the charge conservation (since it includes the exact treatment of the Poisson equation).

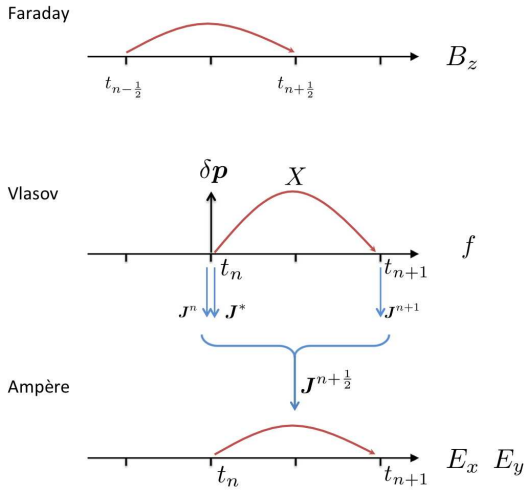


Figure 2: The same numerical scheme as in Fig. 1 is used now in the case of the VLEM1D3V version. The corrected current densities are indicated again by blue arrows on the resolution scheme of  $f$  (middle panel). The Poisson equation is only used at the beginning.

## 5 Numerical tests

In this section we apply the local semi-lagrangian scheme on a series of numerical tests and we compare the resulting solutions with a reference solutions. Two examples will be considered here. In the first example, the VLEM1D3V code is run first by using the Poisson equation (with the numerical scheme shown in Fig. 3- The reference model-) and then, for comparison, by solving the Ampere equation with the correction of the current density (the scheme being presented in Fig. 2). The second example is carried out to check the accuracy of the local spline in VLEM2D2V and its ability to treat a more complex situation when the coupling between CFI and TSI instabilities takes place. In the latter case, the comparisons are made by referring to the global Maxwell-Vlasov solver.

## 5.1 The check of the charge conservation scheme

We now consider the example of an instability met in a relativistic unmagnetized beam-plasma system, described by a 3D (in  $\mathbf{p}$ ) drifting Maxwell distribution. In this analysis a particular emphasis is on the generation of strong magnetic fields. The current filamentation instability<sup>1</sup> (CFI) is a basic plasma process which generates strong and long-living magnetic fields at the expenses of momentum anisotropy, as first shown by Fried in [37]. The initially unmagnetized plasma transforms the initial kinetic energy into electromagnetic energy associated with the amplification of a quasi-static magnetic field. CFI is thus a good example to test the accuracy of our Vlasov solver.

This first test case simulates CFI and we focus here on the charge conservation when Maxwell’s equations are solved numerically. In the one-dimensional form of the Vlasov-Maxwell system, the Poisson equation can be solved at each time step, which ensures the conservation of the charge in its discrete form. Secondly we have obtained in the previous section an explicit expression for the current which satisfies the discretized continuity equation of the Yee scheme for any split Vlasov solver in the advective formulation. A direct comparison between the two techniques is therefore possible. To this purpose, we use first a 1D local version of VLEM, named VLEM1D3V, whose corresponding algorithm is shown in Fig. 2, and, secondly, a model (shown in Fig. 3) called “Full Poisson” because the Poisson equation is solved at each time step with a different time-splitting procedure.

We start with a simulation carried out with VLEM1D3V for the study of CFI in the weak relativistic regime. In both numerical solvers the system is initialised by the identical symmetrical counter-propagating beams. The initial distribution function is given by:

$$f(x, \mathbf{p}, t = 0) = n_{01} F_{max}(p_x, p_y + p_{01}, p_z) + n_{02} F_{max}(p_x, p_y + p_{02}, p_z) \quad (58)$$

where the (normalised) Maxwellian  $F_{max} = \frac{1}{(2\pi)^{3/2} \beta_{th}^3} \exp\left(-\frac{p_x^2 + p_y^2 + p_z^2}{2m^2 c^2 \beta_{th}^2}\right)$  and  $\beta_{th} = \frac{k_B T}{mc^2}$  is the thermal velocity normalised to  $c$ . The initial condition is given by a bi-maxwellian distribution with beam momenta located at  $\mathbf{v}_{01} = -0.6c\mathbf{e}_y$  and  $\mathbf{v}_{02} = +0.6c\mathbf{e}_y$ . We also

---

<sup>1</sup>We point out that, with CFI we refer here to Fried’s interpretation [37] of Weibel temperature anisotropy-driven instability [47] in the limit of two counter-propagating cold electron beams, represented in a kinetic description by two Dirac distributions. For the sake of completeness we note however that it is only with Ref. [48] that Fried’s modelling [37] has been first related to the “current filamentation” process, observed together with the generation of strong quasi-static magnetic fields in numerical simulations of laser-plasma interactions. The analysis provided in Ref. [48] led to the re-interpretation of the Fried mechanism in a fluid modelling, namely in the cold Electron-MHD regime. In this framework, the possibility to generate strong magnetic fields via acceleration of electron beams by intense laser radiation of an overdense plasma had been already highlighted by [49]. The corresponding fluid dispersion relation of the “cold” Weibel-Fried’s [36] instability has been first studied in Ref. [50]. Even if in the early literature Fried’s version of Weibel instability was not distinguished from the “warm” Weibel’s instability of [47], since the end of 1990s some specific terming has been introduced to point out the distinction. In particular we recall here the terming “electromagnetic counter streaming instability” of Ref. [51] and the “streaming Weibel instability” [52]. The terming and acronym we have adopted, which highlights the current filamentation process first evidenced in [48], seems to have been first introduced in Ref. [53].

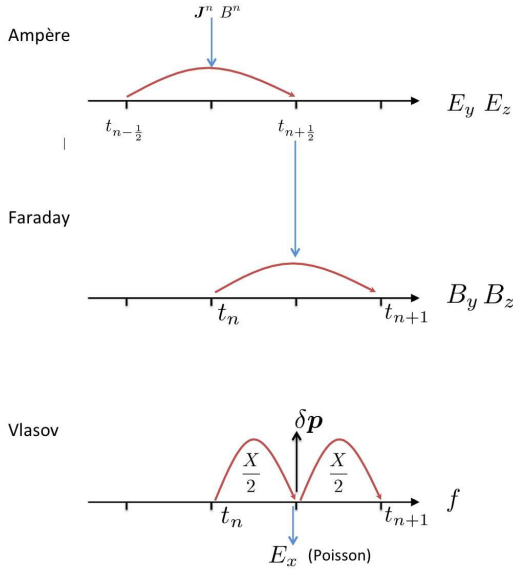


Figure 3: Numerical scheme including the resolution of the Poisson equation at each time step (“Full-Poisson model”), at the end of the first shift in  $x$  over half a time step,  $\Delta t/2$ . Such a technique exactly preserves the charge conservation since the Poisson equation is used. This model is used as a reference model to check the numerical conservation of the charge.

assume  $n_{01} = n_{02} = n_0/2$ . The distribution function is perturbed by purely transverse magnetic fluctuations introduced on the  $z$ -component in the form of a sine function of type  $B_z(x) = \delta B_0 \sin(k_0 x + \varphi)$ , where  $e\delta B_0/(m\omega_p) = 10^{-5}$  and where  $\varphi$  is an arbitrary phase. The phase space sampling is chosen so that  $N_x N_{p_x} N_{p_y} N_{p_z} = 512 \times 80^3$ . The time step is  $\Delta t \omega_p = 0.005$ . The plasma parameters are respectively  $T_{plasma} = T_{beam} = 6keV$ . The numerical space domain is  $L_x \omega_p/c \simeq 3.115$  (leading to a grid spacing of  $\Delta x \omega_p/c \simeq 0.0061$ ). With this choice the largest wavelengths are typically the most unstable and we can initially excite only the largest wavelength admitted in the numerical domain, corresponding here to  $k_0 c/\omega_p \simeq 2.017$  (which corresponds indeed to the fundamental mode  $2\pi/L_x$ ). For each direction, the momentum domain is  $|p_{x,y,z}| \leq 2mc$ . The numerical scheme, in its 1D local version, is shown in Fig.2. We have compared the algorithm to the “reference” one, shown in Fig. 3, where the electrostatic field  $E_x$  is exactly calculated from the Poisson equation at the end of the first shift in  $x$  (corresponding to the operator  $\frac{X}{2}$ ). In that case, the charge conservation is preserved. In the local scheme, shown in Fig. 2, the current density is computed using Eq. (43) which, in its 1D version, leads to the corrected current density:

$$J_{x\ i+1/2}^{n+1/2} = \frac{1}{4} (J_{x\ i}^n + J_{x\ i-1}^n + J_{x\ i}^{n+1} + J_{x\ i-1}^{n+1}) - (J_{x\ i}^{n+1} - J_{x\ i}^*) \quad (59)$$

Fig. 4 shows the time evolution of the magnetic energy  $\epsilon_{m,z} = \frac{1}{2\mu_0 L_x} \int B_z^2 dx$  in a logarithmic scale obtained from the “Full Poisson” model (red curve) and from VLEM1D3V (in blue). We recover the growth of the magnetic energy. The growth rate was found close to  $\frac{\Gamma_{CFI}}{\omega_p} \simeq 0.33$ . On the top panel of Fig. 5, we have plotted the time evolution of the electric energy along

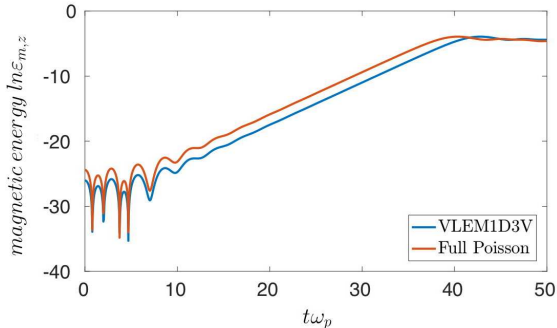


Figure 4: The magnetic energy  $\epsilon_{m,z}$ , in a logarithmic scale, versus time : the Full-Poisson model solution is plotted in red while the VLEM1D3V solution is plotted in blue.

the  $y$  component  $\epsilon_{e,y} = \frac{1}{2L_x} \int \epsilon_0 E_y^2 dx$  and of the electric energy along  $x$  (the electrostatic part), i.e.  $\epsilon_{e,x} = \frac{1}{2L_x} \int \epsilon_0 E_x^2 dx$ , in a logarithmic scale. Note that in Figs 4 and 5 (and then in Fig. 6) a slight difference has been introduced in the initial perturbation in order that the curves are not exactly superimposed, for an easier comparison.

Different energy evolutions have been given both for the local Vlasov solver (in blue) and when we have used the exact treatment by solving the Poisson equation (in red). During the linear phase, both  $\epsilon_{e,y}$  and  $\epsilon_{m,z}$  grow with the same growth rate  $\Gamma_{CFI}/\omega_p$ , while the growth rate of the quantity  $\epsilon_{e,x}$  is double, i.e.  $\Gamma/\omega_p \simeq 0.66$ . Note that the different components of the energy start up from the round-off error level. We have plotted time evolution of the relative charge variation (or mass variation) in Fig. 6 (top panel) and the total electric energy on bottom panel of Fig. 6. We observe in Fig. 6 the very well conservation of the charge density close to  $10^{-5}$ . The conservation of the total energy is close to 1%. Similar results are shown in Fig. 7 for the  $L^2$ -norm, when the saturation mechanism is reached. We see that the  $L^2$ -norm is conserved at 2%, a result which is not too bad if we consider that the regime is strongly nonlinear (note that without the correction in the current, the density charge is not conserved and diverges in several time steps in this example). Secondly the increase in energy remains here weak in comparison to the test case where the flux correction is not included in the model. In this case the increase in energy is exponential and can reach a level of 50%, in this example with the same numerical parameters.

## 5.2 CFI in a 2D spatial system

In this section we present a comparison between the two 2D2V Vlasov solvers : the reference code (using global splines) and VLEM. The next example tests the ability of the code to reproduce results consistent with the expected results of coupling of the electrostatic two-stream instability (TSI) and CFI. As above, a bi-maxwellian given by Eq. (58) is considered. Here, however, we consider a 2D distribution in  $\mathbf{p}$  (the variable  $p_z$  being omitted).

For this initial state, results are computed using both global (the reference code) and local

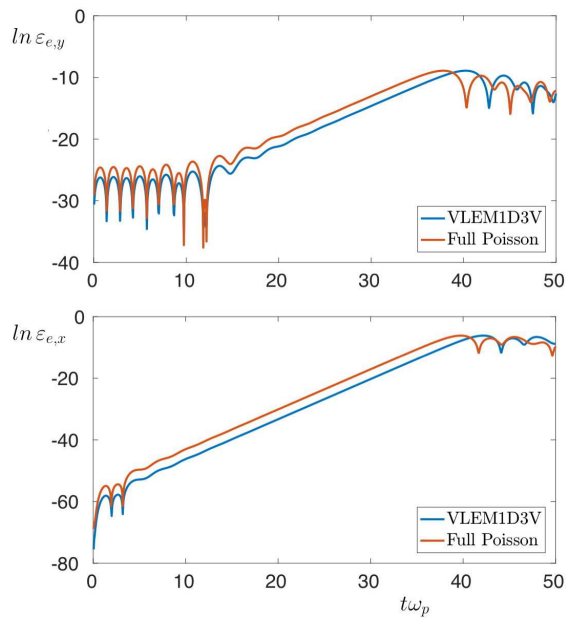


Figure 5: Top panel:  $\epsilon_{e,y}$  component of the electric energy  $\epsilon_e$ , versus time, in a logarithmic scale, showing the exponential growth just before the saturation. Bottom panel: Plot of  $\epsilon_{e,x}$ , in a logarithmic scale, versus time showing the transfer of energy in CFI to the electrostatic part. Results obtained from the VLEM1D3V solver are shown in blue, the simulation of reference solving the Poisson law is shown in red.

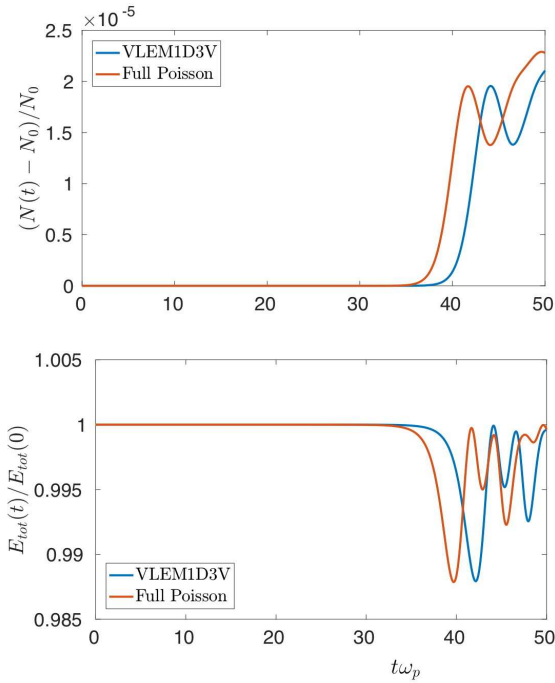


Figure 6: Measure of the relative charge variation versus time. The result obtained from VLEM1D3V is plotted in blue while the reference result is shown in red.

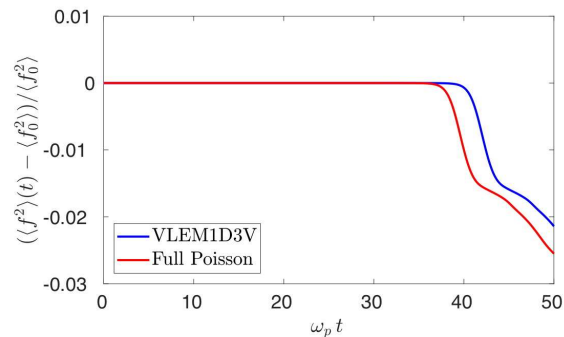


Figure 7: The corresponding relative variation of the  $L^2$  norm versus time for the problem shown in Figs. 4, 5 and 6. The result obtained from VLEM1D3V is plotted in blue while the reference result is shown in red.

SL Vlasov (VLEM) solvers in order to demonstrate the efficiency of the new fully parallelized version of VLEM. This time, the charge correction is not implemented in the local Vlasov model, but the numerical parameters were adapted to obtain a comparable conservation in mass and energy in both solvers.

Numerical simulations have been performed with a phase space sampling corresponding to  $N_x N_y N_{p_x} N_{p_y} = 256^2 \times 128^2$ , while the typical time step is  $\Delta t \omega_p = 0.008$ . We choose here  $p_{01} = 0.9mc$  and  $p_{02} = -0.9mc$  which corresponds to a Lorentz factor of  $\gamma_{01} = \sqrt{1 + p_{01}^2 / (m^2 c^2)} \simeq 1.345$  or, equivalently, to a velocity  $\beta_{01} = p_{01} / (mc \gamma_{01}) \simeq 0.668$  and  $\beta_{02} = -0.668$ . We consider a symmetric configuration, that is  $n_{01} = n_{02} = n_0/2$  and the same plasma and beam temperatures both in longitudinal (in  $p_x$ ) and transverse (in  $p_y$ ) directions, i.e.  $T_x = T_y = 6keV$ . The plasma box has  $L_x = 12\pi c \omega_p^{-1}$  and  $L_y = 12\pi c \omega_p^{-1}$ . A perturbation in density of type  $1 + \sum_{n=1}^{10} \varepsilon \cos(2\pi n_y / L_y)$  has been added to the perturbation on the magnetic field  $\sum_{n=1}^{10} \delta B_0 \sin(2\pi n_x / L_x)$ , with  $\frac{e \delta B_0}{m \omega_p} \simeq 10^{-5}$  and  $\varepsilon = 10^{-3}$ . In the case of the local spline version of the code, we have used  $N_{threads} = 4$  (OpenMP) threads by processor and the total number of processors is  $N_{procs} = 256$  ( $\sqrt{N_{procs}}$  by spatial direction), i.e. 1024 ( $N_{threads}$  times  $N_{procs}$ ) cores.

For the short time scale considered here, ion motion can be neglected so that we only need to consider electron- induced instabilities. These instabilities are clearly identified as CFI, TSI and the oblique instability OI (see Refs. [36, 41]). It is well known that TSI modes are exactly longitudinal ( $k_x = 0$ ), while CFI modes are exactly perpendicular ( $k_y = 0$ ) with respect to the beam direction. If perturbations both parallel and perpendicular to the beam flow become unstable, the filamentation-type instability is referred as oblique. The existence of three distinct instability classes is clearly shown in Fig. 8 where we have plotted the analytic solution of the kinetic dispersion relation using Eq. (A.1) of the appendix.

Indeed, as previously indicated in [36], the three instability classes do not share the same sensitivity to the beam temperature and to the ratio of the beam to the plasma densities. Usually TSI is seen to prevail in presence of a low density ratio and a high beam temperature, while CFI is dominant at higher values of the density (here assumes to be equal for the two beams). High enough beam temperature (close to 1 MeV) always ends up to favour oblique modes. This is illustrated in Fig. 8, where the cross corresponds to a (“triple”) point taken in a region where CFI, TSI and OI coexist with comparable growth-rates. Note that the oblique mode in Fig. 8 corresponds to a new lobe, i.e. a new region of instability in the  $(k_x, k_y)$  plane.

A two-stream governed regime will usually generate phase space holes (Bernstein-Greene-Kruskal - BGK structures) perpendicular to the flow, and may excite an inverse-cascade process leading to the vortex-merging, a typical nonlinear phenomenon. CFI-dominated regime usually prompts filaments or magnetic trapping structures. Oblique modes have rather a hybrid electrostatic -electromagnetic nature usually associated to the “tilted” filaments. Here the emphasis is put on the CFI- TSI coupling. Such a coupling is usually very difficult to observe numerically, since the dominant modes (with higher growth rates) are pure CFI modes (with  $k_y = 0$ ) and OI modes (with  $k_x \neq 0$  and  $k_y \neq 0$ ) localised in a different region of the wave number space and finally TSI modes (with  $k_x = 0$ ). The coupling TSI/CFI takes place with a small growth rate, which necessitates a highly accurate numerical scheme, in



which the start up of the instability is produced by a small perturbation on the corresponding mode.

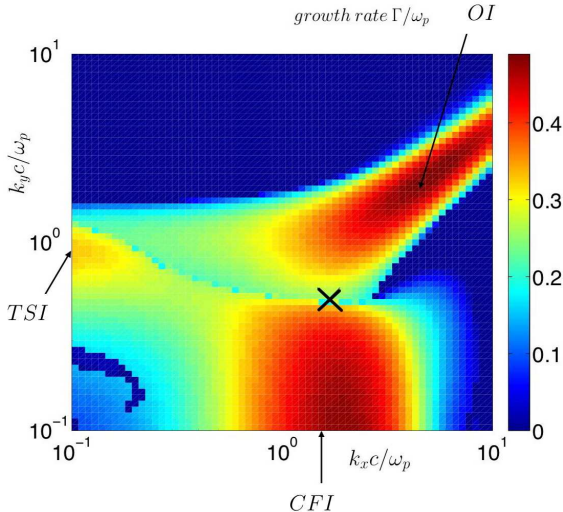


Figure 8: Growth rate  $\frac{\Gamma}{\omega_p}$  versus  $\mathbf{k}$  obtained from the kinetic dispersion relation showing the three different types of instability: CFI, TSI and the OI modes. The cross indicates the choice of physical parameters used in simulation to start up the coupling between the three classes ,of instabilities OI, CFI and TSI

The changes in the characteristics of the CFI dominant mode, when the coupling with TSI is established, are investigated in more detail. As illustrated in Fig. 8, the identification in both simulations (global and local cases) of the same resulting dominant mode is shown, at least for this set of physical parameters. Both Vlasov schemes proved to work very well in the case where the perturbation  $\delta B_0$  is very weak and located in the region where CFI or the oblique instability are not dominant. One of the most famous advantages of the SL method is its noiseless character, due to the very low diffusivity (in comparison to particle codes as Particles-In-Cell (PIC) codes), which allows to study the onset of relativistic instabilities from the level of computer round-off errors  $\sim 10^{-15}$ . Such a noiseless character is preserved by the local spline method, although the continuity of the second spatial derivative is not guaranteed. Thus, the unavoidable growth of the most unstable modes of TSI, CFI and OI take place in a simultaneous way. The cross in Fig. 8 indicates the choice of physical parameters in a region where the growth rate is small. We can thus study in detail this coupling, without observing the occurrence of the CFI and TSI alone, which are characterised by stronger growth-rates.

A first numerical simulation is performed using the global spline interpolation version for  $p_{x,max} = \pm 6mc$  and  $p_{y,max} = \pm 6mc$ . Fig. 9 shows, on top panel and in a logarithmic scale, the time evolution of the electric energy components  $\epsilon_{e,x}$  and  $\epsilon_{e,y}$  and the magnetic energy along the  $z$  direction  $\epsilon_{m,z}$ . We show here only the results obtained with the global version of the code since no difference is observed in VLEM2D2V

On the bottom panel, in Fig. 9, we have plotted the time evolution of the different contri-

contributions of the energies:  $\epsilon_{m,z}$ , the total electric energy  $\epsilon_e = \epsilon_{e,x} + \epsilon_{e,y}$ , the relativistic kinetic energy  $\epsilon_{kin}$  and their sum  $\epsilon_{total} = \epsilon_{kin} + \epsilon_{m,z} + \epsilon_e$ . We see that the magnetic energy, after a transient period, increases linearly in time, in the logarithmic scale, for  $5 \leq t\omega_p \leq 20$ , at a growth rate close to  $\frac{\Gamma_{num}}{\omega_p} \simeq 0.41$ . We observe that the growth of the magnetic energy is accompanied by a strong increase of the (quasi) electrostatic activity.

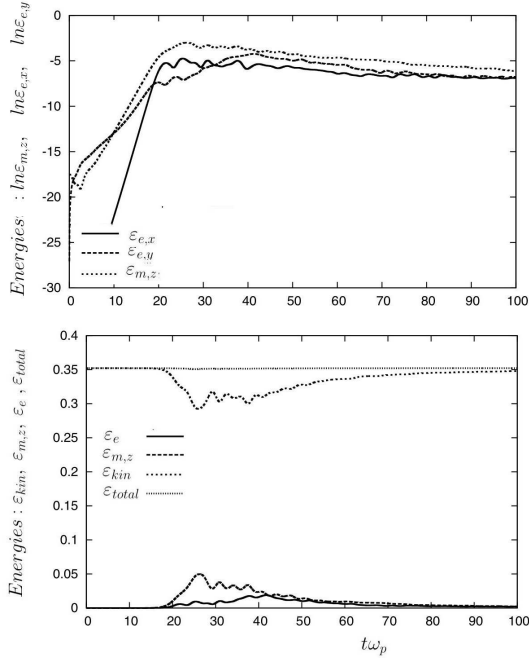


Figure 9: Top panel:  $\epsilon_{e,x}$ ,  $\epsilon_{e,y}$  components of the electric energy  $\epsilon_e$ , the magnetic energy  $\epsilon_{m,z}$  versus time, in a logarithmic scale. Bottom panel: measure of the energy transfer by taking into account the total relativistic kinetic energy  $\epsilon_{kin}$  and the total energy  $\epsilon_{total} = \epsilon_{kin} + \epsilon_e + \epsilon_{m,z}$ . Note that, after saturation, a part of the electromagnetic energy is transferred back to particles, leading to stochastic-type plasma heating.

At this step several remarks are due :

(i) in the early phase, i.e. for  $t\omega_p \leq 5$ , the growth rate of the  $\epsilon_{e,y}$  component is somewhat higher, close to  $\Gamma/\omega_p \simeq 0.48$ , a signature of the dominant mode *CFI*, in well agreement with the theoretical value of  $\Gamma_{CFI}/\omega_p = 0.49$  (for  $k_y = 0$ ), while  $\epsilon_{e,x}$  is Landau damped.

(ii) For an excitation and growth of the magnetic field with a mode  $k_0$  in the case of a linear polarisation, as it is considered here, we expect that the mode  $2k_0$ , induced by the Lorentz force, plays a key role. In particular, this is the dominant mode observed in the electrostatic component. Such a behaviour can be clearly observed in the  $\epsilon_{e,x}$  contribution, which clearly exhibits a linear growth rate  $\Gamma_{2k_0}/\omega_p = 0.82$ . Note that  $\epsilon_{e,y}$  is strongly modified and does not exhibit a linear growth, indicating here the possibility of a coupling with TSI.

(iii) We have also analytically resolved the dispersion relation in the limit of a cold relativistic plasma. For values of wave vector  $k_x = 10\frac{2\pi}{L_x} \simeq 1.667$  and  $k_y = 3\frac{2\pi}{L_y} \simeq 0.50$ , (the dominant modes observed in phase space, as we will see later) the solution of the dispersion relation

Growth rates $\Gamma/\omega_p$	Numerical values
0.413755	$0.41 \pm 0.02$
0.201339	$0.21 \pm 0.02$

Table 2: Analytical solution of the cold relativistic dispersion relation for physical parameters used in the simulation shown in Figs. 9 , 10 and 11.

leads to twelve non zero complex roots of type  $\omega = \omega_r + i\Gamma$ . The corresponding growth rates, shown in Tab. 2, are estimated from linear theory to be  $\frac{\Gamma}{\omega_p}(k_x = 1.667, k_y = 0.50) \simeq 0.41375$ , in good agreement with the numerical value of  $\Gamma_{num}/\omega_p \simeq 0.41 \pm 0.02$ , while another root is obtained at  $\Gamma/\omega_p \simeq 0.20134$ . This last root corresponds to the TSI branch and is observed in the simulation, the corresponding numerical growth rate being  $\Gamma_{num}/\omega_p \simeq 0.21$ .

A surprising aspect appears in the long time diagnostics of the energy transfer. The latter may trigger secondary electrostatic-type instabilities such as the longitudinal two-stream instability. The self-reorganisation of the plasma, at saturation, begins with a symmetry-breaking instability of coalescence of phase space vortices, leading to a stochastic heating process. The plasma evolution is characterised by an “inverse cascade” process. The saturation regime is associated with a strong heating in both longitudinal and perpendicular directions (to the beam flow). In the initial stage of the instability (see Fig. 9 on bottom panel), the kinetic energy is transferred to the magnetic field. After saturation, part of the energy stored in the electromagnetic field is transferred back to the plasma particles, which leads to strong heating, thus reducing the initial anisotropy in momentum.

Detailed particle dynamics from global SL Vlasov solver is shown in Fig. 10 (the reference solution) for  $y - p_y$  representation. Here the distribution function has been integrated over the other corresponding variables of the phase space. Fig. 11 illustrates the same diagnostics for the numerical simulation running using VLEM, where high resolution details of  $f$  are depicted. First examination of Fig. 11, with respect to the reference solution shown in Fig. 10, reveals very similar behaviour of the distribution function in phase space. The magnetic trapping structures are formed at saturation.

The structures resulting from the growth of TSI consist of strong electrostatic trapping vortices in the distribution function, which can be clearly observed in the  $\tilde{f}(y, p_y)$  averaged distribution. In the Vlasov simulations discussed here, the main process linked to the plasma reorganisation starts with the occurrence of a secondary instability, i.e. the phase space vortex merging observed in Figs. 10 and 11. Such a secondary instability is usually expected to appear in the electrostatic two-stream instability as indicated in Refs. [38, 39] or in laser- plasma interaction giving rise to a similar vortex- merging mechanism of low-frequency electron acoustic waves (EAWs) in [7].

### 5.3 The full relativistic regime

**In contrast to previous subsections, the fully relativistic regime addressed here involves, for both the counterstreaming plasma and beam populations, initial**

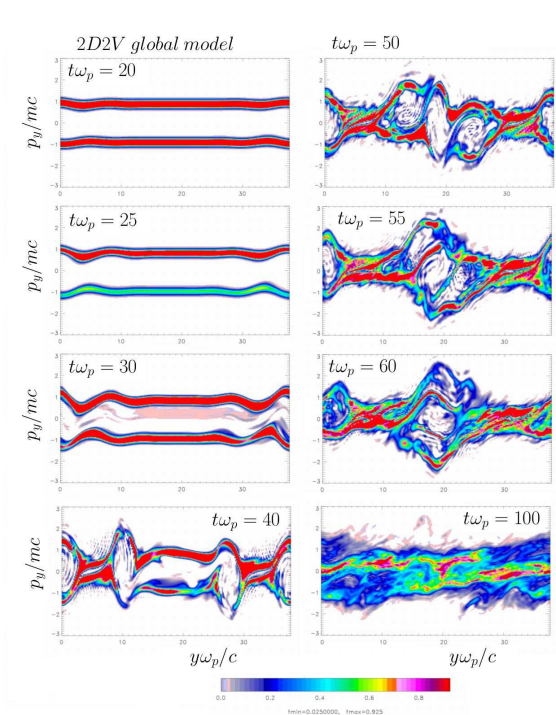


Figure 10:  $y - p_y$  phase space illustration obtained with the reference code showing the vortex-merging scenario induced by TSI .

unperturbed distribution functions in the form of drifting Maxwell- Jüttner distributions of type

$$F_{0j}(\mathbf{p}) = I(\mu_j) e^{-\mu_j(\gamma(\mathbf{p}) - \beta_j p_y)} \quad (60)$$

where the index  $j = (1, 2)$  stands for the plasma and the beam component respectively. Here  $\beta_j = \frac{v_j}{c}$  is the normalized  $y$ - aligned mean drift velocity and  $\mu_j = \frac{mc^2}{k_B T_j}$  the normalized inverse temperature of the electron component  $j$ . The distribution function is normalized to unity:  $\int F_{0j} d^2 p = 1$  and  $I$  is a normalization constant. Charge and current neutralization are again assumed, so as to ensure a field- free equilibrium. This implies the condition  $n_1 + n_2 = 1$  (the electron densities being normalized to the ion density as usually) and  $n_1 \beta_1 + n_2 \beta_2 = 0$ . Ions form a fixed neutralizing background and collisions are neglected. To maintain the connection with the study of the “triple point” mode analysis, previously described in sec. 5.2 in the non relativistic regime, we restrict here the study to a two-dimensional case both in space and in momentum (thus here  $\gamma = \gamma(p_x, p_y)$ ). Having exploring the non relativistic coupling of OI , TSI and CFI in sec. 5.2, we now address how CFI is modified in the relativistic regime. In general the Maxwell- Jüttner distribution has three components in momentum (in that case the normalization constant writes as  $\mu_j / 4\pi \gamma_j^2 K_2(\mu_j / \gamma_j)$  where  $\gamma_j$  is the relativistic

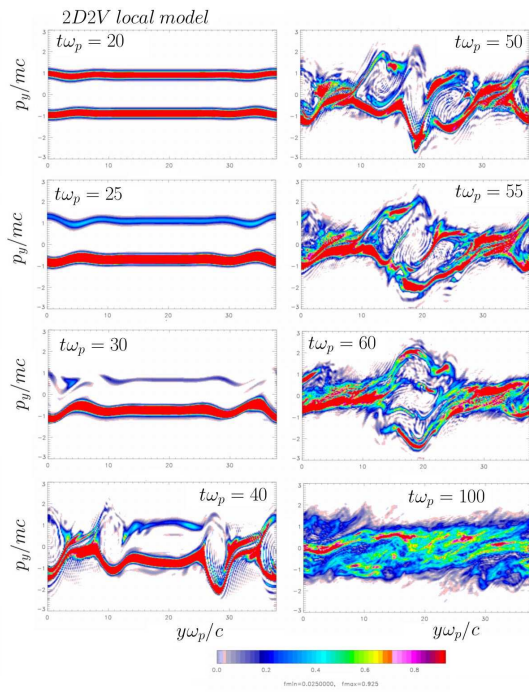


Figure 11:  $y - p_y$  phase space illustration afforded by VLEM to compare with the reference solution of Fig. 10. Here the Vlasov code is parallelized via domain decomposition in the configuration space

factor of the plasma or beam component and  $K_2$  denotes the modified Bessel function of the second kind). Two arguments can be made for using a Maxwell-Jüttner function. The first is that it allows an exact treatment of the 2D axisymmetric case for an affordable numerical cost. The second is the availability of theoretical approaches related to the influence of oblique modes in the linear and relativistic regime (see Ref. [40] for more details). One should note, that this function is not suited to modeling temperature anisotropy effects as Weibel instability [47], since it depends on only one thermal spread parameter  $\mu_j$ .

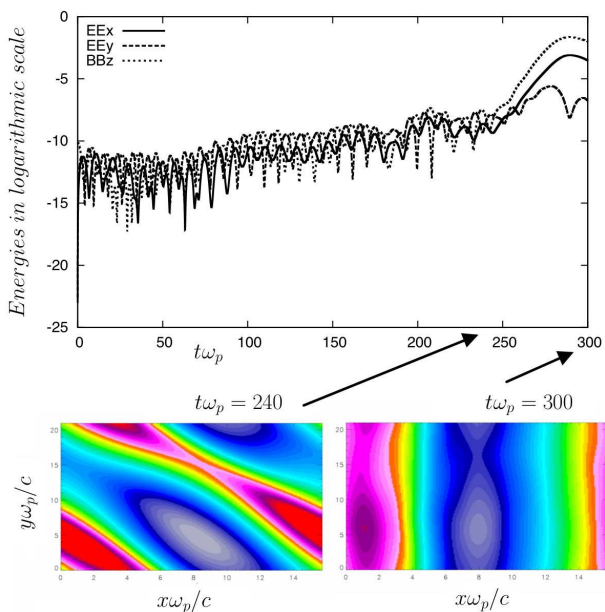


Figure 12: Time evolution of the different energies plotted in a logarithmic scale, on the top panel, exhibiting an oscillatory behavior, in the full relativistic regime. The corresponding magnetic field representation in space taken at two different instants during the time evolution is plotted in the bottom panels. The electron beam temperature is  $T_2 = 2000keV$  while the plasma temperature is  $T_1 = 20keV$ .

The study of the CFI to OI transition has been investigated in Ref. [40]. We choose to initiate our system with a condition described in Fig. 3 of the Ref. [40], a plot of the solution of the dispersion relation showing the excitation of a dominant oblique mode for the following physical parameters:  $\frac{n_1}{n_2} = 1$ ,  $\gamma_2 = 1.5$  (or equivalently  $\beta_2 = -\beta_1 \simeq 0.75$ ) and a beam temperature of  $T_2 = 2000keV$  and a plasma temperature of  $T_1 = 5keV$ . Although an accurate analysis of the OI depends on the shape of the initial distribution function, we think, at least qualitatively, that the prediction will hold for a wide range of initial distribution shapes. However, due to the fact we have chosen an initial distribution with only two momentum components ( $p_x$  and  $p_y$ ), instead of three, a stronger growth rate

may be expected. For this reason we have chosen to increase the plasma temperature to  $20keV$ , allowing to reduce the growth rate of the oblique instability to a comparable level to the 3D case. Several numerical simulations, performed for weaker plasma temperatures, while keeping other physical parameters identical, have shown a similar global behavior. The lengths of the plasma box in space are  $L_x = 15.707c\omega_p^{-1}$  (or  $k_x c/\omega_p = 0.40$ ) and  $L_y = 20.943c\omega_p^{-1}$  (or equivalently  $k_y c/\omega_p = 0.30$ ). A perturbation on the magnetic field of type  $\delta B_0 \sin(n_x k_x x + n_y k_y y)$ , with  $n_x = n_y = 1$ , is applied with an amplitude perturbation of  $\delta B_0 = 0.0125$ . The numerical simulation was carried out with a phase space sampling of  $N_x N_y N_{p_x} N_{p_y}$  of  $128^3 \times 256$ . The time step is here  $\Delta t \omega_p = 0.005$  and the momentum space is defined by  $|\frac{p_x}{mc}| \leq 20$  and  $-20 \leq \frac{p_y}{mc} \leq 60$ .

While filamentation mode has a zero phase velocity (since its frequency is zero), the oblique mode exhibits a finite phase velocity. As an example of the change of CFI (usually expected for  $\frac{n_1}{n_2} = 1$  in the relativistic regime), we have displayed in Fig. 12, on the same plot and on top panel, the time evolution of the different energies,  $\epsilon_{e,x}$ ,  $\epsilon_{e,y}$  and  $\epsilon_{m,z}$ , in a logarithmic scale, which exhibit an oscillatory behavior at the same frequency close to  $\omega \simeq 0.375\omega_p$ . In this equal density case, the observed frequency of the oblique mode (occurring in the first step of the instability till a time of  $t\omega_p \leq 250$ ) is larger than the expected growth rate of the mode observed at the saturation (and close to  $\frac{\Gamma_{CFI-OI}}{\omega_p} \simeq 0.08$ ) from which follows a non zero phase velocity of  $v_\varphi = \frac{\omega}{|k|} = \frac{0.375}{\sqrt{0.4^2+0.3^2}}c \simeq 0.695c$  close to  $v_2$ . This contrasts with the zero wave phase velocity of the non oscillating dominant filamentation mode. Strong modification in the topology of the magnetic field (shown in the bottom panels in Fig. 12) is observed between the linear phase (where OI occurs as expected) and at saturation (where the magnetic field exhibits a quasi-homogeneous structure in the  $y$  direction indicating a CFI-OI coupling).

We shall not explore there further in detail here, except to note that such an oscillatory behavior (linked to a non zero frequency of the excited mode) can also be a source of resonant wave- particle interactions. One example of the distribution function  $\tilde{f}(p_x, p_y)$  (averaged over the other directions) is shown in Fig. 13 on top panel. Fig. 13 shows how the growth of OI (with a non zero phase velocity) results in the disymetrization of the distribution function in momentum, due to the resonance between beam particles and wave. This is inherently a kinetic process, depending on the structure of the particle phase space density at the resonance  $v_\varphi \sim v_2$  (the beam velocity), a physical mechanism not predicted in the linear theory. This secondary instability, which is now associated with the relativistic electron beam streaming, relative to plasma, looks like the Buneman instability and can accelerate the conversion of the beam energy into plasma heating in the nonlinear saturation regime. Such a heating process is observed in the  $\tilde{f}(x, p_x)$  and  $\tilde{f}(x, p_y)$  representations of the electron distribution function, plotted on bottom panels in Fig. 13. We observe clearly the deformation of the electron distribution in the resonance region located in  $2 \leq \frac{x\omega_p}{c} \leq 6$ .

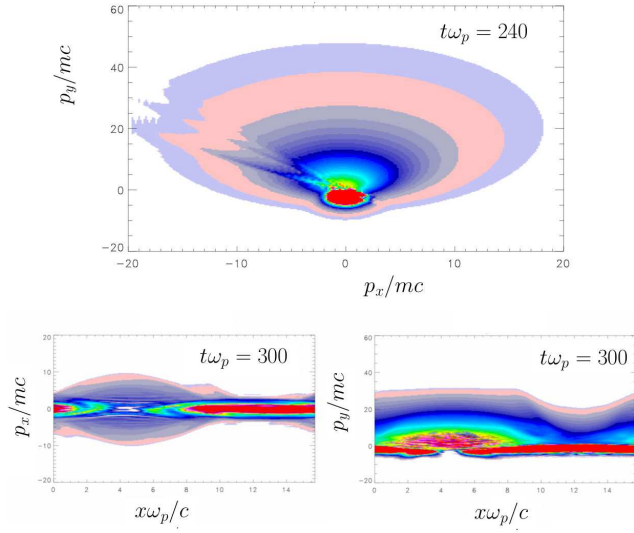


Figure 13: Representation of the electron distribution function in the momentum space at a given time during the oscillating phase of the oblique mode on top panel, showing a disymmetrization de la distribution function induced by particle- wave resonance. On botton panel the  $\tilde{f}(x, p_x)$  and  $\tilde{f}(x, p_y)$  representation of the distribution function at the saturation showing the resulting heating process due to the resonance between the electron beam and the wave in the full relativistic regime. The electron beam temperature is  $T_2 = 2000kev$  while the plasma temperature is  $T_1 = 20keV$ .



## 6 Numerical Performance tests.

To study the performance of the parallel algorithms, we introduce the speed-up  $S(N_c)$  and the efficiency  $E(N_c)$ , defined as follows :

$$S(N_c) = \frac{\text{total elapsed time on one core}}{\text{total elapsed time on } N_c \text{ cores}} \quad \text{and} \quad E(N_c) = \frac{S(N_c)}{N_c} \quad (61)$$

The present algorithm is parallelized via domain decomposition only in the coordinate space. However, the electromagnetic field has been computed in a sequential way, making easier and simpler the implementation of a version conserving the charge. The performance of the different implementations of the Vlasov code are examined in this section, the target platform being an IBM Intel Sandy Bridge E5 having 332 SMP nodes (only 2048 cores being accessible for an user), connected with a high performance infinBand FRD-10 Mellanox (with two links by node) interconnect. Each node is composed of 4 Intel Sandy Bridge E5- 4650 processor of  $2.7GHz$ , each processor integrating eight cores.

### 6.1 Performance, speed-up and MPI

Different optimisation techniques are applied to two different parts of the code, and they concern different aspects : OpenMP multithreading and MPI communications. In this first example, our purpose is to examine the parallelism efficiency using MPI in detail by increasing the number of cores without using OpenMP (only one thread had been used). Our MPI program involves a number of processes running concurrently, exchanging data via explicit message passing. We choose to begin with a 2D2V version of the Vlasov-Maxwell code using a fixed sampling of the phase space,  $N_x N_y N_{p_x} N_{p_y} = 1024 \times 512 \times 16^2$  for 500 temporal iterations. Results are shown in Fig. 14. The theoretical speed-up is plotted in solid line, while the values given by the simulations have been represented by crosses : till 256 cores, a nice parallel behaviour is observed. **However, for a number of cores greater than 256, we observe a loss of efficiency which can be attributed to the Amdahl's law.** The reason of the loss in performance can be linked to the sequential treatment of the electromagnetic field. When subtracting the computational cost of the electromagnetic solver (close to 0.15%) from the total elapsed time, a quasi-linear speed-up has been recovered (dotted line in Fig. 14). We obtain a linear speed-up  $S(N_c) \sim N_c$  and a very good efficiency  $E(N_c) \simeq 1$  in this case. We find an excellent scaling till 2048 cores. This is a clear indication that a parallel implementation of the Maxwell solver becomes necessary and would lead to a substantial performance improvement. We leave this additional parallelization work of the Maxwell solver to future work.

### 6.2 OpenMP Parallel Performance

Now, we execute VLEM2D2V with multiple OpenMP threads for only one processor using a phase space sampling of  $N_x N_y = 256^2$  and  $N_{p_x} N_{p_y} = 16^2$ , again for 500 time iterations.

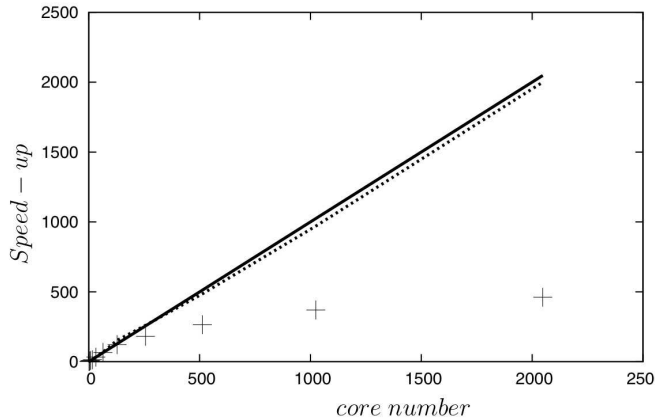


Figure 14: The theoretical speed-up is plotted in full line, while the values given by the simulations have been represented by crosses. When subtracting the computational cost of the electromagnetic solver (close to 0.15%) from the total elapsed time, a quasi-linear speed-up has been recovered (dotted line).

Table 3 gives the measured results. As expected, the speed-up  $S(N_t)$  ( $N_t$  being the number of threads) grows linearly with the number of threads, for small values of  $N_t \leq 8$ . We ran this test program until the maximum of  $N_t = 16$  threads. Up to eight threads, a nice parallel behaviour is observed, with a speedup proportional to the number of threads.

**A few comments are in order:**

(i) The gain in performance till eight threads is encouraging. Because each processor has eight cores and 16 threads, e.g. eight “true” threads (available in multi-threading processes) and eight other threads (which can be used in hyperthreading), the hyperthreading seems to be not well adapted to our numerical Vlasov code.

(ii) On the other hand, modern architectures rely on vectorization to deliver a substantial fraction of the full core performance. Thus additional options, not directly related to multithreading, can also be explored, such as vectorization or optimized memory accesses. In well-adapted data parallel applications - and VLEM might be one of them due to its “eulerian” approach using a fixed grid- the combination of multithreading and vectorization might provide significant performance enhancements. Once the external loop over  $x$  (in the 3D  $p$  advection) has been distributed across threads, the inner loop ( $y$  for instance), executed by each thread, might be an excellent candidate for vectorization.

$N_t$	Process Elapsed time (s)	$S(N_t)$	$E(N_t) = \frac{S(N_t)}{N_t}$
1	4800	1	1
2	2410	1.99	0.995
4	1302	3.680	0.92
8	722	6.64	0.83
16	997	4.81	0.3

Table 3:  $S(p)$  and efficiency  $E(N_t) = \frac{S(N_t)}{N_t}$  as a function of the number of threads  $N_t$  for the code VLEM2D2V running on **one processor** for a phase space sampling  $N_x N_y N_{p_x} N_{p_y}$  of  $256^2 \times 16^2$  grid points and 500 iterations.  $S(N_t)$  is the ratio of the total process **elapsed** time for one thread over the total process elapsed time for  $N_t$  threads.

## 7 Conclusion

In order to investigate the transverse geometric effects of Weibel-type instabilities, a two-dimensional relativistic semi-lagrangian Vlasov code has been implemented and uses a domain decomposition in space. This Vlasov-Maxwell code employs a time splitting scheme, local Hermite interpolation or B-spline interpolation techniques, adapted to optimally use distributed computational environments without using global transposition methods. A new technique preserving the local charge conservation has been investigated. The method is based on the equivalence between the form factor used in PIC codes and the spline or Hermite function used in the Semi-Lagrangian approach. This allows to rely on backward SL techniques.

The typical examples discussed in this paper have been chosen to reflect the specific difficulties associated with the physics but also the parallelization processes of more general electromagnetic Vlasov codes. Performance tests have been performed on the same cluster composed of several hundreds of SMP nodes incorporating two Sandy Bridge sockets, each one with eight cores. The performances obtained in these examples are very satisfying and correspond to realistic situations of high performance computing.

### Acknowledgment

The authors thank the IDRIS computational centre, Orsay, France, for computer time allocation on their computers. This work was granted across to the HPC resources (Grant 2016- 057290) made by GENCI (Grand Equipement National de Calcul Intensif). The authors thank N. Crouseilles, P. Navaro, G. Latu, E. Sonnendrücker for useful discussions and support.

## Appendix A: Kinetic dispersion relation for CFI- TSI and Oblique modes

The non relativistic dispersion relation can be written in the general form

$$D_{xx}D_{yy} - D_{xy}^2 = 0 \quad (\text{A.1})$$

where the matrix  $\mathbf{D}$  is the dispersion matrix which takes the form

$$D = \begin{pmatrix} D_{xx} & D_{xy} \\ D_{xy} & D_{yy} \end{pmatrix} \quad (\text{A.2})$$

In the dispersion relation to CFI- TSI and OI modes, the matrix elements are:

$$D_{xx} = \frac{\omega^2}{c^2} \varepsilon_{xx} - k^2 \sin^2 \theta \quad (\text{A.3})$$

$$D_{yy} = \frac{\omega^2}{c^2} \varepsilon_{yy} - k^2 \cos^2 \theta \quad (\text{A.4})$$

$$D_{xy} = \frac{\omega^2}{c^2} \varepsilon_{xy} + k^2 \sin \theta \cos \theta \quad (\text{A.5})$$

Here  $\theta$  is the angle between the wave vector  $\mathbf{k}$  and the axis ( $Ox$ ), the beams being along  $y$ .  $\varepsilon$  is the dielectric matrix of the plasma whose coefficients are given by the following relations:

$$\varepsilon_{xx} = 1 + \sum_{\alpha=1,2} \frac{\omega_{p\alpha}^2}{\omega^2} \left( \frac{\Omega_\alpha^2}{k^2 v_{th\alpha}^2} (1 + \xi_\alpha Z(\xi_\alpha)) \cos^2 \theta + \xi_\alpha Z(\xi_\alpha) \sin^2 \theta \right) \quad (\text{A.6})$$

$$\begin{aligned} \varepsilon_{yy} = & 1 + \sum_{\alpha=1,2} \frac{\omega_{p\alpha}^2}{\omega} \frac{\Omega_\alpha}{k^2 v_{th\alpha}^2} (1 + \xi_\alpha Z(\xi_\alpha)) \sin^2 \theta + \sum_{\alpha=1,2} \frac{\omega_{p\alpha}^2 v_{0\alpha} \sin \theta}{\omega k v_{th\alpha}^2} (1 + \xi_\alpha Z(\xi_\alpha)) \\ & + \sum_{\alpha=1,2} \frac{\omega_{p\alpha}^2 k v_{0\alpha}}{\omega} \frac{\sin \theta \cos^2 \theta}{\omega} \frac{\Omega_\alpha}{k^2 v_{th\alpha}^2} (1 + \xi_\alpha Z(\xi_\alpha)) - \sum_{\alpha=1,2} \frac{\omega_{p\alpha}^2 k v_{0\alpha}}{\omega} \frac{\sin \theta \cos^2 \theta}{\omega} \frac{Z(\xi_\alpha)}{\sqrt{2} k v_{th\alpha}} \\ & + \sum_{\alpha=1,2} \frac{\omega_{p\alpha}^2}{\omega} \cos^2 \theta \frac{Z(\xi_\alpha)}{\sqrt{2} k v_{th\alpha}} + \sum_{\alpha=1,2} \frac{\omega_{p\alpha}^2 v_{0\alpha}^2 \cos^2 \theta}{\omega^2} \frac{1 + \xi_\alpha Z(\xi_\alpha)}{v_{th\alpha}^2} \end{aligned} \quad (\text{A.7})$$

$$\begin{aligned} \varepsilon_{xy} = & \sum_{\alpha=1,2} \frac{\omega_{p\alpha}^2}{\omega} \frac{\Omega_\alpha}{k^2 v_{th\alpha}^2} (1 + \xi_\alpha Z(\xi_\alpha)) \left( \sin \theta \cos \theta + \frac{k v_{0\alpha}}{\omega} \cos^3 \theta \right) \\ & + \sum_{\alpha=1,2} \frac{\omega_{p\alpha}^2}{\omega} \frac{Z(\xi_\alpha)}{\sqrt{2} k v_{th\alpha}} \left( \frac{k v_{0\alpha} \sin \theta}{\omega} - 1 \right) \sin \theta \cos \theta \end{aligned} \quad (\text{A.8})$$

The velocity of beam  $\alpha$  is  $v_{0\alpha}$ ,  $Z(\xi_\alpha)$  is the plasma dispersion relation of argument  $\xi_\alpha = \frac{\Omega_\alpha}{\sqrt{2} k v_{th\alpha}}$ , with the thermal velocity  $v_{th\alpha} = \sqrt{\frac{k_B T_\alpha}{m}}$  for the beam  $\alpha$ , and  $\Omega_\alpha = \omega - k_y v_{0\alpha} = \omega - k \sin \theta v_{th\alpha}$ .

## Appendix B: Order of the global scheme in time

To show how the numerical scheme presented in section 4 integrates the following Vlasov equation given by Eq. (52) restricted to the 2D2V case (the extension to the 2D3V being straightforward), consider first what happens to the distribution function as the operator  $\frac{X}{2}\frac{Y}{2}P\frac{Y}{2}\frac{X}{2}$  is applied to the initial distribution function  $f^n(x, y, p_x, p_y, t = n\Delta t)$ ,  $\frac{X}{2}$ ,  $\frac{Y}{2}$ , and  $P$  denoting the shift in position  $x$  and  $y$  (over half a time step) and in momentum (over a full time step) respectively. Thus we obtain:

$$f^{n+1}(x, y, p_x, p_y) = f^n(\bar{x}, \bar{y}, \bar{p}_x, \bar{p}_y) \quad (\text{B.1})$$

with

$$\bar{x} = x - \frac{\bar{p}_x}{m\bar{\gamma}}\Delta t + O(\Delta t^3) \quad (\text{B.2})$$

$$\bar{y} = y - \frac{\bar{p}_y}{m\bar{\gamma}}\Delta t + O(\Delta t^3) \quad (\text{B.3})$$

with  $\bar{\gamma} = \bar{\gamma}(\bar{p}_x, \bar{p}_y)$  and

$$\bar{p}_x = p_x - e \left( \bar{E}_x + \frac{v_y + \bar{v}_y}{2} \bar{B}_z \right) \Delta t + O(\Delta t^3) \quad (\text{B.4})$$

$$\bar{p}_y = p_y - e \left( \bar{E}_y - \frac{v_x + \bar{v}_x}{2} \bar{B}_z \right) \Delta t + O(\Delta t^3) \quad (\text{B.5})$$

By noting that, at the end of the sequence  $\frac{X}{2}\frac{Y}{2}$  we have

$$\frac{1}{\bar{\gamma}} \simeq \frac{1}{\gamma} + e (\bar{E}_x + v_y \bar{B}_z) \frac{p_x}{m^2 c^2 \gamma^3} \Delta t + e (\bar{E}_y - v_x \bar{B}_z) \frac{p_y}{m^2 c^2 \gamma^3} \Delta t \quad (\text{B.6})$$

we can write

$$\bar{x} = x - \frac{p_x}{m\gamma} \Delta t + e (\bar{E}_x + v_y \bar{B}_z) \frac{1 + \frac{p_y^2}{m^2 c^2}}{2m\gamma^3} \Delta t^2 - \frac{p_x p_y e}{2m^3 c^2 \gamma^3} (\bar{E}_y - v_x \bar{B}_z) \Delta t^2 + O(\Delta t^3) \quad (\text{B.7})$$

$$\bar{y} = y - \frac{p_y}{m\gamma} \Delta t + e (\overline{E}_y - v_x \overline{B}_z) \frac{1 + \frac{p_x^2}{m^2 c^2}}{2m\gamma^3} \Delta t^2 - \frac{p_x p_y e}{2m^3 c^2 \gamma^3} (\overline{E}_x + v_y \overline{B}_z) \Delta t^2 + O(\Delta t^3) \quad (\text{B.8})$$

where  $\overline{E}_{x,y} = E_{x,y} \left( x - \frac{p_x}{2m\gamma} \Delta t, y - \frac{p_y}{2m\gamma} \Delta t \right)$  and  $\overline{B}_z = B_z \left( x - \frac{p_x}{2m\gamma} \Delta t, y - \frac{p_y}{2m\gamma} \Delta t \right)$ . Taking into account Eq. (B.6), Eqs. (B.4) and (B.5) leads to

$$\overline{v}_x \equiv \frac{\overline{p}_x}{m\overline{\gamma}} = \frac{p_x}{m\gamma} - \frac{e\Delta t}{m\gamma} (\overline{E}_x + v_y \overline{B}_z) + \frac{p_x^2 (\overline{E}_x + v_y \overline{B}_z) \Delta t}{m^3 c^2 \gamma^3} + \frac{e (\overline{E}_y - v_x \overline{B}_z) p_x p_y \Delta t}{m^3 c^2 \gamma^3} \quad (\text{B.9})$$

$$\overline{v}_y \equiv \frac{\overline{p}_y}{m\overline{\gamma}} = \frac{p_y}{m\gamma} - \frac{e\Delta t}{m\gamma} (\overline{E}_y - v_x \overline{B}_z) + \frac{p_y^2 (\overline{E}_y - v_x \overline{B}_z) \Delta t}{m^3 c^2 \gamma^3} + \frac{e (\overline{E}_x + v_y \overline{B}_z) p_x p_y \Delta t}{m^3 c^2 \gamma^3} \quad (\text{B.10})$$

Finally using (B.9) and (B.10), Eqs. (B.4) and (B.5) give now

$$\begin{aligned} \overline{p}_x = p_x - e\overline{E}_x \Delta t - e\Delta t \overline{B}_z \left[ \frac{p_y}{m\gamma} - e (\overline{E}_y - v_x \overline{B}_z) \frac{1 + \frac{p_x^2}{m^2 c^2}}{2m\gamma^3} \Delta t \right. \\ \left. + e (\overline{E}_x + v_y \overline{B}_z) \frac{p_x p_y \Delta t}{2m^3 c^2 \gamma^3} \right] + O(\Delta t^3) \end{aligned} \quad (\text{B.11})$$

$$\begin{aligned} \overline{p}_y = p_y - e\overline{E}_y \Delta t - e\Delta t \overline{B}_z \left[ \frac{p_x}{m\gamma} - e (\overline{E}_x + v_y \overline{B}_z) \frac{1 + \frac{p_y^2}{m^2 c^2}}{2m\gamma^3} \Delta t \right. \\ \left. + e (\overline{E}_y - v_x \overline{B}_z) \frac{p_x p_y \Delta t}{2m^3 c^2 \gamma^3} \right] + O(\Delta t^3) \end{aligned} \quad (\text{B.12})$$

On the other hand, let us now consider the equations describing the particle motion in a two-dimensional system (we keep the same notation to simplify the presentation):

$$\frac{dx}{dt} = v_x(t) = \frac{p_x(t)}{m\gamma(t)} \quad (\text{B.13})$$

$$\frac{dy}{dt} = v_y(t) = \frac{p_y(t)}{m\gamma(t)} \quad (\text{B.14})$$

$$\frac{d}{dt}(mv_x\gamma) = e(E_x + v_y B_z) \quad (\text{B.15})$$

$$\frac{d}{dt}(mv_y\gamma) = e(E_y - v_x B_z) \quad (\text{B.16})$$

and where the Lorentz factor  $\gamma$  is now defined in Eqs (B.15) and (B.16) by

$$\gamma = \frac{1}{\sqrt{1 - \frac{v_x^2}{c^2} - \frac{v_y^2}{c^2}}}. \quad (\text{B.17})$$

From Eqs. (B.13) to (B.16) and since

$$\frac{dv_x}{dt} = \frac{1}{\gamma^3} \left(1 + \frac{v_y^2 \gamma^2}{c^2}\right) \frac{d}{dt}(\gamma v_x) - \frac{v_x v_y}{\gamma c^2} \frac{d}{dt}(\gamma v_y) \quad (\text{B.18})$$

$$\frac{dv_y}{dt} = \frac{1}{\gamma^3} \left(1 + \frac{v_x^2 \gamma^2}{c^2}\right) \frac{d}{dt}(\gamma v_y) - \frac{v_x v_y}{\gamma c^2} \frac{d}{dt}(\gamma v_x) \quad (\text{B.19})$$

we have

$$x^n = x^{n+1} - v_x^{n+\frac{1}{2}} \Delta t = x^{n+1} - \left(\frac{v_x^{n+1} + v_x^n}{2}\right) \Delta t \quad (\text{B.20})$$

$$y^n = y^{n+1} - v_y^{n+\frac{1}{2}} \Delta t = y^{n+1} - \left(\frac{v_y^{n+1} + v_y^n}{2}\right) \Delta t \quad (\text{B.21})$$

and for velocities

$$v_x^n = v_x^{n+1} - \frac{e\Delta t}{m\gamma^3} \left(1 + \frac{(v_y^{n+1})^2 \gamma^2}{c^2}\right) \left(E_x^{n+\frac{1}{2}} + v_y^{n+\frac{1}{2}} B_z^{n+\frac{1}{2}}\right) + \frac{e\Delta t v_x^{n+1} v_y^{n+1}}{mc^2 \gamma} \left(E_y^{n+\frac{1}{2}} - v_x^{n+\frac{1}{2}} B_z^{n+\frac{1}{2}}\right) \quad (\text{B.22})$$

$$v_y^n = v_y^{n+1} - \frac{e\Delta t}{m\gamma^3} \left( 1 + \frac{(v_x^{n+1})^2 \gamma^2}{c^2} \right) \left( E_y^{n+\frac{1}{2}} - v_x^{n+\frac{1}{2}} B_z^{n+\frac{1}{2}} \right) + \frac{e\Delta t v_x^{n+1} v_y^{n+1}}{mc^2 \gamma} \left( E_x^{n+\frac{1}{2}} + v_y^{n+\frac{1}{2}} B_z^{n+\frac{1}{2}} \right) \quad (\text{B.23})$$

Using relations (B.22) and (B.23), we have then

$$x^n = x^{n+1} - v_x^{n+1} \Delta t + \frac{e}{2m\gamma^3} \left( 1 + \frac{(v_y^{n+1})^2 \gamma^2}{c^2} \right) \left( E_x^{n+\frac{1}{2}} + v_y^{n+1} B_z^{n+\frac{1}{2}} \right) \Delta t^2 - \frac{e v_x^{n+1} v_y^{n+1}}{2mc^2 \gamma} \left( E_y^{n+\frac{1}{2}} - v_x^{n+1} B_z^{n+\frac{1}{2}} \right) \Delta t^2 + O(\Delta t^3) \quad (\text{B.24})$$

$$y^n = y^{n+1} - v_y^{n+1} \Delta t + \frac{e}{2m\gamma^3} \left( 1 + \frac{(v_x^{n+1})^2 \gamma^2}{c^2} \right) \left( E_y^{n+\frac{1}{2}} - v_x^{n+1} B_z^{n+\frac{1}{2}} \right) \Delta t^2 - \frac{e v_x^{n+1} v_y^{n+1}}{2mc^2 \gamma} \left( E_x^{n+\frac{1}{2}} + v_y^{n+1} B_z^{n+\frac{1}{2}} \right) \Delta t^2 + O(\Delta t^3) \quad (\text{B.25})$$

where  $\gamma = \gamma(v_x^{n+1}, v_y^{n+1})$ . A similar analysis for the momentum variables  $p_x$  and  $p_y$  leads to

$$p_x^n = p_x^{n+1} - eE_x^{n+\frac{1}{2}} \Delta t - eB_z^{n+\frac{1}{2}} \Delta t \left[ v_y^{n+1} - \frac{e\Delta t}{2m\gamma^3} \left( 1 + \frac{(v_x^{n+1})^2 \gamma^2}{c^2} \right) \left( E_y^{n+\frac{1}{2}} - v_x^{n+1} B_z^{n+\frac{1}{2}} \right) + \frac{e v_x^{n+1} v_y^{n+1}}{2mc^2 \gamma} \left( E_x^{n+\frac{1}{2}} + v_y^{n+1} B_z^{n+\frac{1}{2}} \right) \Delta t \right] + O(\Delta t^3) \quad (\text{B.26})$$

$$p_y^n = p_y^{n+1} - eE_y^{n+\frac{1}{2}} \Delta t + eB_z^{n+\frac{1}{2}} \Delta t \left[ v_x^{n+1} - \frac{e\Delta t}{2m\gamma^3} \left( 1 + \frac{(v_y^{n+1})^2 \gamma^2}{c^2} \right) \left( E_x^{n+\frac{1}{2}} + v_y^{n+1} B_z^{n+\frac{1}{2}} \right) + \frac{e v_x^{n+1} v_y^{n+1}}{2mc^2 \gamma} \left( E_y^{n+\frac{1}{2}} - v_x^{n+1} B_z^{n+\frac{1}{2}} \right) \Delta t \right] + O(\Delta t^3) \quad (\text{B.27})$$



By comparing the equations (B.7), (B.8), (B.11), (B.12) and (B.24), (B.25), (B.26) and (B.27), we see without difficulty that the splitting scheme integrates the distribution function along the characteristics correctly to the second order in  $\Delta t$ .

## References

- [1] J.M. Dawson, Rev. Mod. Phys. **55**, 403 (1983)).
- [2] C.A. Birdsall, A.B. Langdon, Plasma Physics via computer simulations, Mc Graw-Hill, New-York 1985.
- [3] G. Lapenta, J. Comput. Physics **231**, 795 (2012).
- [4] P. Bertrand, A. Ghizzo, T.W. Johnston, M. Shoucri, E. Fijalkow, M. R. Feix, Phys. Fluids B **2** (5) 1028 (1990).
- [5] T.W. Johnston, P. Bertrand, A. Ghizzo, M. Shoucri, E. Fijalkow, M.R. Feix, Phys. Fluids B **4**, 2523 (1992).
- [6] A. Ghizzo, P. Bertrand, J. Lebas, T.W. Johnston, M. Shoucri, Phys. Plasmas **3**, 650 (1996).
- [7] M. Albrecht-Marc, A. Ghizzo, T.W. Johnston, T. Reveille, D. Del Sarto, P. Bertrand, Phys. Plasmas **14**, 072704 (2007).
- [8] E. Sonnendrücker, J. Roche, P. Bertrand, A. Ghizzo, J. Comput. Physics **149**, 201 (1999).
- [9] M.L. Begue, A. Ghizzo, P. Bertrand, E. Sonnendrücker, O. Coulaud, J. Plasma Physics **62**, 367 (1999).
- [10] M.L. Begue, A. Ghizzo, P. Bertrand, J. Comput. Physics **151**, 458 (1999).
- [11] A. Ghizzo, F. Huot, P. Bertrand, J. Comput. Physics **186**, 47 (2003).
- [12] T. Nakamura, T. Yabe, Comput. Phys. Comm. **120**, 122 (1999).
- [13] E. Fijalkow, Comput. Phys. Comm. **116**, 319 (1999).
- [14] F. Filbet, E. Sonnendrücker, P. Bertrand, J. Comput. Phys. **172**, 166 (2001).
- [15] F. Filbet, E. Sonnendrücker, Comput. Phys. Comm. **150**, 247 (2003).
- [16] X. Garbet, Y. Idomura, L. Villard, T.H. Watanabe, Nuclear Fusion **50**, 043002 (2010).

- [17] V. Grandgirard, M. Brunetti, P. Bertrand, N. Besse, X. Garbet, P. Ghendrih, E. Sonnendrücker, J. Vaclavik, L. Villard, *J. Comput. Phys.* **217**, 395 (2006).
- [18] N. Crouseilles, G. Latu, E. Sonnendrücker, *Int. J. Appl. math. Comput. Sci.* **17**, 335 (2007).
- [19] A.J. Klimas, W.M. Farrell, *J. Comput. Phys.* **110**, 150 (1994).
- [20] M.C. Pinto, E. Sonnendrücker, A. Friedman, D.P. Grote, S. M. Lund, *J. Comput. Physics* **275**, 236 (2014).
- [21] N. Crouseilles, M. Mehrenberger, E. Sonnendrücker, *J. Comput. Phys.* **229**, 1927 (2010).
- [22] F. Huot, A. Ghizzo, P. Bertrand, E. Sonnendrücker, O. Coulaud, *J. Comput. Physics* **185**, 512 (2003).
- [23] L.K.S. Daldorff, B. Eliasson, *Parallel computing* **35**, 109 (2009).
- [24] T. Umeda, K. Togano, T. Ogino, *Comp. Phys. Comm.* **180**, 365 (2009).
- [25] T. Minoshima, Y. Matsumoto, T. Amano, *J. Comput. Physics* **230**, 6800 (2011).
- [26] Y. Cheng, A.J. Christlieb, X. Zhang, *J. Comput. Phys.* **279**, 145 (2014).
- [27] A. Ghizzo, T.W. Johnston, T. Reveille, P. Bertrand, M. Albrecht-Marc, *Phys. Rev. E* **74**, 046407 (2006).
- [28] A. Ghizzo, D. Del Sarto, R. Reveille, N. Besse, R. Klein, *Phys. Plasmas* **14**, 062702 (2007).
- [29] O. Coulaud, E. Sonnendrücker, E. Dillon, P. Bertrand, A. Ghizzo, *J. Plasma Physics* **3**, 435 (1999).
- [30] F. Valentini, P. Travnicek, F. Califano, P. Hellinger, A. Mangeney, *J. Comput. Phys.* **225**, 753 (2007).
- [31] A. Mangeney, F. Califano, C. Cavazzoni, P. Travnicek, *J. Comput. Phys.* **179**, 495 (2002).
- [32] C.G. Cheng, G. Knorr, *J. Comput. Phys.* **22**, 330 (1976).
- [33] K. Yee, *IEEE Trans. On Ant. and Prop.* **14**, 302 (1966).
- [34] T. Zh Esirkepov, *Comput. Phys. Comm* **135**, 144 (2001).
- [35] N. Crouseilles, P. Navaro, E. Sonnendrücker, *CR Mécanique* **342**, 636 (2014).
- [36] A. Bret, L. Gremillet, D. Bénisti, E. Lefebvre, *Phys. Rev. Letters* **100**, 205008, (2008).
- [37] B.D. Fried, *Physics Fluids* **2**, 337 (1959).

- [38] A. Ghizzo, B. Izrar, P. Bertrand, E. Fijalkow, M.R. Feix, M.M. Shoucri, *Phys. Fluids* **31**, 72 (1988).
- [39] A. Ghizzo, B. Izrar, P. Bertrand, E. Fijalkow, M.R. Feix, M.M. Shoucri, *Phys. Lett. A* **120**, 191 (1987).
- [40] A. Bret, L. Gremillet, D. Benisti, *Phys. Rev. E* **81**, 036402 (2010).
- [41] G. Kalman, C. Montes and D. Guemada, *Phys. Fluids* **11**, 1797 (1968).
- [42] R. Bermejo, *Numer. Math* **60**, 163 (1991).
- [43] M. Falcone and R. Ferreti, *SIAM J. Numer. Anal.* **35**, 909 (1998).
- [44] P. Bartello and S.J. Thomas, *Mon. Weather Rev.* **124**, 2883 (1996).
- [45] N. Besse, *SIAM J. Numer. Anal.* **42**, 350 (2004).
- [46] A. Staniforth, J. Coté, *Mon. Weather Reveiw* **119**, 206 (1991).
- [47] E.W. Weibel, *Phys. Rev. Letters* **2**, 83 (1959).
- [48] G.A. Askar'Yan, S.V. Bulanov, F. Pegoraro and A.M. Pukhov, *JETP Lett.* **60**, 251 (1994).
- [49] R. Sudan, *Phys. Rev. Lett.* **70**, 3075 (1993).
- [50] F. Pegoraro, S.V. Bulanov, F. Califano and M. Lontano, *Phys. Scripta* **T63**, 262 (1996).
- [51] Y. Kaziura, F. Califano, J.-I. Sakai, T. Neubert, F. Pegoraro, S. Bulanov, *J. Phys. Soc. Jpn.*, **67**, 1079 (1998).
- [52] R. Schlickeiser and P.K. Shukla, *Astophys. J.* **599**, L60 (2003).
- [53] F. Califano, R. Prandi, F. Pegoraro and S. Bulanov, *Phys. Rev. E.* **58**, 7837 (1998).