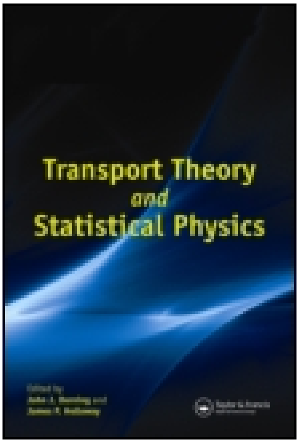


Transport Theory and Statistical Physics

Publication details, including instructions for authors and subscription information:
http://www.tandfonline.com/loi/ltty20

Semi-Lagrangian Schemes for the Two-Dimensional Vlasov-Poisson System on Unstructured Meshes

Nicolas Besse [a], Jacques Segré [b] & Eric Sonnendrücker [a]

[a] Institut de Recherche Mathematique Avancée, Université Louis Pasteur—CNRS, Strasbourg, France

[b] Commissariat à l'Energie Atomique de Saclay, France

Available online: 18 Aug 2006

Transport Theory and Statistical Physics, 34: 311–332, 2005
Copyright © Taylor & Francis, Inc.
ISSN 0041-1450 print/1532-2424 online
DOI: 10.1080/00411450500274592

Semi-Lagrangian Schemes for the Two-Dimensional Vlasov-Poisson System on Unstructured Meshes

Nicolas Besse
Institut de Recherche Mathematique Avancée, Université Louis
Pasteur—CNRS, Strasbourg, France

Jacques Segré
Commissariat à l'Energie Atomique de Saclay, France

Eric Sonnendrücker
Institut de Recherche Mathematique Avancée, Université Louis
Pasteur—CNRS, Strasbourg, France

Abstract: In this article, we present new high-order, semi-Lagrangian schemes for solving the Vlasov-Poisson system on an unstructured four-dimensional phase-space mesh. The method is based on the propagation of the distribution function and its jacobian by following the characteristic curves backward. Then the distribution function is reconstructed using high-order and few diffusive interpolation operators coming from the finite element and the computer aided geometric design (CAGD) literature. Numerical tests in plasma physics and charged-particle beam transport are investigated.

Keywords: Semi-Lagrangian schemes, Vlasov-Poisson, charge particle beams, plasma physics

1. INTRODUCTION

Many problems in plasma physics as well as in beam physics are modeled by the Vlasov equation, nonlinearly coupled to Poisson's equation or the Maxwell

Received 16 February 2004, Accepted 15 July 2004

Address correspondence to Nicolas Besse, 7 rue Rene Descartes, 67084, Strasbourg Cedex, France. E-mail: besse@math.u-strasbg.fr

system for the self-consistent fields. Such a model can be solved numerically using the particle-in-cell (PIC) method. This method solves the Vlasov equation by directly integrating the equations of motion of the particles in time, the self-consistent fields being computed on a grid of configuration space. This method has been very successful for a vast variety of physical problems. However, in some cases it is useful to get a more precise description of the distribution function, especially in the less-populated parts of phase space. In this case, it is useful to use a more precise numerical method to directly solve the Vlasov equation on a grid of phase space. Many such methods have been investigated (Fijalkow 1999; Filbet et al. 2001; Feix et al. 1994; Filbet and Sonnendrücker 2003; Ghizzo et al. 1990; Nakamura and Yabe 1999). One method that has been particularly successful is the semi-Lagrangian method which has been introduced in a simplified form, using a time-splitting procedure and cubic spline interpolation, by Cheng and Knorr (1976). This method has been extended to a more general context in (Sonnendrücker et al. 1999) still using a uniform grid of phase space. The semi-Lagrangian method consists in following the characteristic curves backward at each time-step and interpolating the values at the feet of characteristics by an interpolation scheme. Therefore, the semi-Lagrangian method rests on solving the ordinary differential equations (ODEs) ($\dot{X}(t) = f(t, X(t))$) associated to the transport operator and building the less diffusive and the more accurate stable interpolation operator. If we choose to solve the full transport operator we have to solve nonlinear ODEs which can be solved by appropriate ODE solver like Runge-Kutta schemes. Nevertheless, as the second member of the ODEs ($f(t, X(t))$) is known at time t^n and the unknowns of the ODEs ($X(t)$) are known at time t^{n+1} (the mesh nodes), then solving the ODE leads to solving a fixed-point problem where convergence numerical difficulties can arise. Another strategy is to split the complete transport operator in several simple transport operators whose associated ODEs are more simple to solve. In the case of the Vlasov-Poisson system, if we split the transport in physical space and the transport in velocity space, we can integrate the characteristic curves backward analytically. Then we will choose the latter strategy. Therefore the method presented in this article still belongs to the class of semi-Lagrangian methods but has the distinctive feature of using a splitting of the transport operator.

In some cases, for geometrical reasons in particular, it is very convenient to use an unstructured mesh of configuration space. In addition, unstructured meshes are well suited for adaptive mesh refinement which is very useful to increase the ratio of the accuracy on the computational cost as Vlasov equation describes multi-scale phenomena. A method using an unstructured mesh of phase space in 1D and 1D1/2 has been introduced in Besse and Sonnendrücker 2003. This method is based on finite element-like interpolation techniques on triangular grids and on time-splitting schemes which are also applied to inhomogeneous equations of the gradients of the distribution function. A convergence analysis of our method has been carried out in Besse (submitted). This article presents a generalization of this technique to 2D, i.e., 4D in phase space which is not trivial, as it is not easy and perhaps not efficient to generate a fully unstructured mesh of 4D space. Moreover, in the 1D and 1D1/2 case treated in Besse and Sonnendrücker (2003) we do not need to directly compute the gradient of the electric field or the gradient of the force applied to the particles because we can deduce it from other physical quantities (in 1D $\partial_x \; E(t, \; x) = \rho(t, \; x)$ and in 1D1/2 $\partial_r \; F(t, \; r) = \rho(t, \; r) - E_r(t, \; r)/r - 2(I/r^2)^2 - \omega_0^2$ where $I = rv_\theta$). In the two-dimensional case we have to compute the gradient of the electric field by finding an equation on the gradient. Besides, the treatment of boundary conditions is more complex on curve domains when we solve the Poisson equation with high-order finite element methods. This article is devoted to introducing a new method based on a tensor product like mesh of phase space which is fully unstructured in configuration space and uniform in velocity space.

The model we consider throughout the article is the two-dimensional nonrelativistic Vlasov-Poisson system, which is four-dimensional in phase space. The 2D nonrelativistic Vlasov equation reads

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla_{\mathbf{x}} f + \frac{q}{m}(\mathbf{E}_s + \mathbf{E}_a) \cdot \nabla_{\mathbf{v}} f = 0, \tag{1}$$

where $f = f(t, \mathbf{x}, \mathbf{v}) = f(t, x, y, v_x, v_y)$, is the particle distribution function, $\mathbf{E}_a(\mathbf{x})$ is a given applied field which we assume for the sake of simplicity not to depend on time, and $\mathbf{E}_s(t, \; \mathbf{x}) = \mathbf{E}_s(t, \; x, \; y)$ is the self-consistent electric field which is obtained by solving the Poisson equation

$$\begin{cases} \mathbf{E}_s(t, x, v) = -\nabla_{\mathbf{x}} \phi(t, x, y) \\ -\Delta_{\mathbf{x}} \phi(t, x, y) = \dfrac{\rho}{\epsilon_0}(t, x, y) \\ \rho(t, x, y) = q \int_{\mathbb{R}^2} f(t, \mathbf{x}, \mathbf{v}) d\mathbf{v} \end{cases} \tag{2}$$

In the sequel, we shall denote by

$$\mathbf{F}(t, x, y) = \frac{q}{m}(\mathbf{E}_s + \mathbf{E}_a),$$

the total force field (divided by the particle mass m), and $(F_x, \; F_y)$ its components.

The article is organized as follows: first we present the numerical algorithm for the Vlasov equation. It makes use of a time-splitting procedure which enables the decoupling of the transport in configuration space from the transport in velocity space, and thus the associated meshes. The interpolation part uses cubic Hermite-type finite elements where the degree of freedom are the values of the function and its derivatives. Because of this, the gradients of the force field are needed which calls for a specific, careful procedure to solve the Poisson equation. This will be described in details. Finally, the code is validated on two classical problems of plasma physics and beam physics.

2. NUMERICAL RESOLUTION OF THE VLASOV EQUATION

We numerically solve the Vlasov equation using a semi-Lagrangian algorithm. This algorithm is based on the property of the distribution function, that it is conserved along the characteristics of the Vlasov equation (which are the particle trajectories). Using this feature the distribution function at all the grid points can be updated with the following two steps:

1. Follow the characteristics of the Vlasov equation backward in time for one time step.
2. Interpolate the old value of the distribution function at the origin of the characteristics.

The first step can be rendered very simple thanks to a time-splitting procedure between position and velocity so that the origin of the characteritics at each split step can be computed explicitly. The interpolation step is based on a Hermite-type finite element interpolation procedure. This interpolation makes use of the values of f as well as of those of the derivatives.

2.1. Time Advance

Let us define the transport operators in position and velocity space associated to the Vlasov equation. The transport equation with velocity $\mathbf{v} = (v_x, v_y)$ reads

$$\frac{\partial f}{\partial t} + v_x \frac{\partial f}{\partial x} + v_y \frac{\partial f}{\partial y} = 0.$$

Its solution on one time step is given explicitly by

$$f(t + \Delta t, x, y, v_x, v_y) = f(t, x - v_x \Delta t, y - v_y \Delta t, v_x, v_y).$$

We denote by $\mathcal{S}_{x,y}$ (Δt) the operator $f(t) \mapsto f(t + \Delta t)$ associated to this transport equation. In the same way, we define the transport operator in velocity space at a given position by

$$\frac{\partial f}{\partial t} + F_x \frac{\partial f}{\partial v_x} + F_y \frac{\partial f}{\partial v_y} = 0. \tag{3}$$

Here again, the solution on one time step is given explicitly by

$$f(t + \Delta t, x, y, v_x, v_y) = f(t, x, y, v_x - F_x \Delta t, v_y - F_y \Delta t).$$

Indeed, integrating the transport Equation (3) with respect to velocity, it follows that $\rho = q \int f dv_x dv_y$ is conserved by this equation and thus also $\mathbf{E}_s$. For this reason, F_x and F_y are constants in this setting. We then denote by $\mathcal{S}_{v_x},v_y(\Delta t)$ the operator $f(t) \mapsto f(t+\Delta t)$ associated to this transport equation.

Note that the explicit values of the solution of both these transport operators can be derived with respect to all the position and velocity variables and thus yield their espressions with respect to the old ones.

The solution operator $\mathcal{S}(\Delta t)$ associated to the full Vlasov equation can then be approached with third-order accuracy in time by

$$\mathcal{S}(\Delta t) \approx \mathcal{S}_{x,y}(\Delta t/2) \circ \mathcal{S}_{v_x,v_y}(\Delta t) \circ \mathcal{S}_{x,y}(\Delta t/2).$$

This procedure is called the Strang splitting. It is third-order accurate in time on one time step, and thus second-order accurate globally in time.

Assume the distribution function $f(t^n, x, y, v_x, v_y)$ is known at time t^n; then the following steps are necessary to compute the distribution function at time t^{n+1} with our splitting scheme (note that the terms on the right-hand-side need to be evaluated thanks to an interpolation procedure that will be detailed later):

1. First half advection in position space and computation of the derivatives:

$$\begin{aligned}
f^*(x, y, v_x, v_y) &= f(t^n, x - v_x\Delta t/2, y - v_y\Delta t/2, v_x, v_y),\\
\partial_x f^*(x, y, v_x, v_y) &= \partial_x f(t^n, x - v_x\Delta t/2, y - v_y\Delta t/2, v_x, v_y),\\
\partial_y f^*(x, y, v_x, v_y) &= \partial_y f(t^n, x - v_x\Delta t/2, y - v_y\Delta t/2, v_x, v_y),\\
\partial_{v_x} f^*(x, y, v_x, v_y) &= \partial_{v_x} f(t^n, x - v_x\Delta t/2, y - v_y\Delta t/2, v_x, v_y)\\
&\quad - \frac{\Delta t}{2}\partial_x f(t^n, x - v_x\Delta t/2, y - v_y\Delta t/2, v_x, v_y)\\
\partial_{v_y} f^*(x, y, v_x, v_y) &= \partial_{v_y} f(t^n, x - v_x\Delta t/2, y - v_y\Delta t/2, v_x, v_y)\\
&\quad - \frac{\Delta t}{2}\partial_y f(t^n, x - v_x\Delta t/2, y - v_y\Delta t/2, v_x, v_y).
\end{aligned}$$

2. Computation of the electric field $\mathbf{E}_s{}^*(x, y)$ by using f^* as a source term in the Poisson equation:

$$\begin{cases}
\mathbf{E}_s^*(x, y) = -\nabla_\mathbf{x}\phi^*(x, y)\\
-\Delta_\mathbf{x}\phi^*(x, y) = \dfrac{\rho^*}{\epsilon_0}(x, y)\\
\rho^*(x, y) = q\int_{\mathbb{R}^2} f^*(x, y, v_x, v_y)dv_x dv_y.
\end{cases}$$

3. Computation of the gradient of the electric field $\nabla_\mathbf{x}\, \mathbf{E}_s{}^*(x, y)$ by solving the following derived Poisson equation:

$$\begin{cases}
\nabla_\mathbf{x}\mathbf{E}_\mathbf{s}^*(x, y) = -\nabla_\mathbf{x}(\nabla_\mathbf{x}\phi^*(x, y))\\
-\Delta_\mathbf{x}(\nabla_\mathbf{x}\phi^*(x, y)) = \nabla_\mathbf{x}\rho^*(x, y)\\
\nabla_\mathbf{x}\rho^*(x, y) = q\int_{\mathbb{R}^2} \nabla_\mathbf{x} f^*(x, y, v_x, v_y)dv_x dv_y.
\end{cases}$$

4. Advection in velocity space and computation of the derivatives:

$$\begin{aligned}
f^{**}(x, y, v_x, v_y) =& f^*(x, y, v_x - F_x^*(x, y)\Delta t, v_y - F_y^*(x, y)\Delta t),\\
\partial_x f^{**}(x, y, v_x, v_y) =& \partial_x f^*(x, y, v_x - F_x^*(x, y)\Delta t, v_y - F_y^*(x, y)\Delta t)\\
&- \Delta t \partial_x F_x^*(x, y)\partial_{v_x} f^*(x, y, v_x - F_x^*(x, y)\Delta t, v_y\\
&- F_y^*(x, y)\Delta t),\\
&- \Delta t \partial_x F_y^*(x, y)\partial_{v_y} f^*(x, y, v_x - F_x^*(x, y)\Delta t, v_y\\
&- F_y^*(x, y)\Delta t),\\
\partial_y f^{**}(x, y, v_x, v_y) =& \partial_y f^*(x, y, v_x - F_x^*(x, y)\Delta t, v_y - F_y^*(x, y)\Delta t)\\
&- \Delta t \partial_y F_x^*(x, y)\partial_{v_x} f^*(x, y, v_x - F_x^*(x, y)\Delta t, v_y\\
&- F_y^*(x, y)\Delta t),\\
&- \Delta t \partial_y F_y^*(x, y)\partial_{v_y} f^*(x, y, v_x - F_x^*(x, y)\Delta t, v_y\\
&- F_y^*(x, y)\Delta t),\\
\partial_{v_x} f^{**}(x, y, v_x, v_y) =& \partial_{v_x} f^*(x, y, v_x - F_x^*(x, y)\Delta t, v_y\\
&- F_y^*(x, y)\Delta t)\\
\partial_{v_y} f^{**}(x, y, v_x, v_y) =& \partial_{v_y} f^*(x, y, v_x - F_x^*(x, y)\Delta t, v_y\\
&- F_y^*(x, y)\Delta t)
\end{aligned}$$

where

$$F_x^*(x, y) = E_x^*(x, y) + E_{a_x}(x, y), F_y^*(x, y) = E_y^*(x, y) + E_{a_y}(x, y).$$

5. Second half advection in position space and computation of the derivatives:

$$\begin{aligned}
f(t^{n+1}, x, y, v_x, v_y) =& f^{**}(x - v_x\Delta t/2, y - v_y\Delta t/2, v_x, v_y),\\
\partial_x f(t^{n+1}, x, y, v_x, v_y) =& \partial_x f^{**}(x - v_x\Delta t/2, y - v_y\Delta t/2, v_x, v_y),\\
\partial_y f(t^{n+1}, x, y, v_x, v_y) =& \partial_y f^{**}(x - v_x\Delta t/2, y - v_y\Delta t/2, v_x, v_y),\\
\partial_{v_x} f(t^{n+1}, x, y, v_x, v_y) =& \partial_{v_x} f^{**}(x - v_x\Delta t/2, y - v_y\Delta t/2, v_x, v_y)\\
&- \frac{\Delta t}{2}\partial_x f^{**}(x - v_x\Delta t/2, y - v_y\Delta t/2, v_x, v_y)\\
\partial_{v_y} f(t^{n+1}, x, y, v_x, v_y) =& \partial_{v_y} f^{**}(x - v_x\Delta t/2, y - v_y\Delta t/2, v_x, v_y)\\
&- \frac{\Delta t}{2}\partial_y f^{**}(x - v_x\Delta t/2, y - v_y\Delta t/2, v_x, v_y)
\end{aligned}$$

2.2. Interpolation Procedure

In order to perform the necessary interpolations, the phase space is gridded in a tensor product manner, in order to be convenient for the different split steps. We use a triangular unstructured mesh in position space and a structured rectangular mesh in velocity space overlying each mesh point of position space. Each rectangle of the velocity space is then subdivided into two triangles for the interpolation scheme.

Interpolation schemes on unstructured meshes have been thoroughly studied in the frame of the finite-element method; see, for example, Ciarlet (1978). Lagrange-type interpolation methods have been proven to be unstable on triangles (Besse 2003). On the other hand, Hermite-type methods, in the spirit of the CIP method introduced by Nakamura and Yabe (1999), work well. They have been studied for the Vlasov equation in one dimension in Besse and Sonnendrücker (2003). Before describing the interpolation procedure we choose for each advection step, we briefly recall the description of the finite element we use. In order to use local interpolation operators, we have to define the triple (T, P_T, Σ_T) (see Ciarlet 1978). In fact we just have to define Σ_T, the set of degrees of freedom on the finite element T and Π_T the local interpolation operator on T. Next we denote, respectively, by P_2, and P_3 the sets of polynomials of degree two and three. Finally, we introduce the notation mod where $i \bmod j = i - \mathrm{int}(i/j) \times j$.

The $\mathscr{C}^1$ cubic Nielson Rational Singular Element (NC1)

The set of the degrees of freedom is given by

$$\Sigma_T = \{f(a_i):\ 1 \leq i \leq 3;\ \partial_x f(a_i),\ \partial_y f(a_i):\ 1 \leq i \leq 3\},$$

and the local interpolation operator Π_T is determined for all f in $\mathscr{C}^1\bar{(}\ \bar{T})$ by

$$\begin{aligned}\Pi_T f = \sum_{i=1}^{3} & f(a_i)(\lambda_i^2(3-2\lambda_i) + 6w\lambda_i(\lambda_k\alpha_{ij} + \lambda_j\alpha_{ik})) \\ & + \nabla_{\mathbf{x}} f(a_i)\cdot(a_j - a_i)[\lambda_i^2\lambda_j + w\lambda_i(3\lambda_j\alpha_{ik} + \lambda_k - \lambda_j)] \\ & + \nabla_{\mathbf{x}} f(a_i)\cdot(a_k - a_i)[\lambda_i^2\lambda_k + w\lambda_i(3\lambda_k\alpha_{ij} + \lambda_j - \lambda_k)],\end{aligned}$$

where

$$w = \frac{\lambda_1\lambda_2\lambda_3}{\lambda_1\lambda_2 + \lambda_2\lambda_3 + \lambda_1\lambda_3}, \quad \alpha_{ij} = \frac{||e_i||^2 + ||e_j||^2 - ||e_k||^2}{2||e_j||^2},$$

and $||e_i||$ denotes the length of edge e_i opposite to the vertex a_i. We have the relationship $P_2 \subset P_T \subset P_3$. In fact, we recover the $\mathscr{C}^1$ continuity through the edge by adding rational polynomials (see Ciarlet 1978; Nielson 1980). Moreover, NC1 operator has a good stability property since it is a discretization of the operator $\mathscr{M}[f]$ characterized as the unique interpolant which minimizes the pseudo-norm

$$\left(\int_T \left|\frac{\partial^2 f}{\partial x \partial y}(x,y)\right|^2 dxdy\right)^{1/2}$$

among all functions in $\mathscr{C}^4(T)$ which interpolate any $f \in \mathscr{C}^4(T)$ and its first derivative on the boundary of T (see Nielson 1980).

The $\mathscr{C}^1$ Cubic Hsieh-Clough-Tocher Element (HCTC)

If a_i is a vertex of a triangle T, then we denote, respectively, by l_i, and m_i the length and the middle of the edge of T opposite to the vertex a_i. We denote by h_i the intersection point of the edge opposite to the vertex a_i and the perpendicular to this edge which goes through a_i. Then we introduce $n_i = |h_i - a_i|$ and ν_i the unit exterior normal of the edge opposite to a_i. Let a be the barycentre of T, then K_i denotes the subtriangle of T built with the vertex a, a_j, and a_k where $1 \le i \le 3$, $j = i\text{mod}3 + 1$ and $k = j\text{mod}3 + 1$. The set of the degrees of freedom is given by

$$\Sigma_T = \{f(a_i):\ 1 \le i \le 3;\ \partial_x f(a_i),\ \partial_y f(a_i), \partial_{\nu_i} f(m_i):\ 1 \le i \le 3\},$$

where ∂_{ν_i} denotes the normal derivative. We can replace Σ_T by $\Sigma_T{}'$ where

$$\Sigma'_T = \{f(a_i):\ 1 \le i \le 3;\ \partial_x f(a_i),\ \partial_y f(a_i), \partial_x f(m_i), \partial_y f(m_i):\ 1 \le i \le 3\}.$$

The local interpolation operator Π_T is determined for all f in $\mathscr{C}^{1}\bar{(}\ T)$ by

$$\begin{aligned}\Pi_T f_{|K_l} = & \sum_{i=l}^{(l+1)\text{mod}3+1} f(a_i)\Psi^0_{l,i} + (\nabla_{\mathbf{x}} f(a_i).\overrightarrow{a_i a_k})\Psi^1_{l,i,k} \\ & + (\nabla_{\mathbf{x}} f(a_i).\overrightarrow{a_i a_j})\Psi^1_{l,i,j} + (\nabla_{\mathbf{x}} f(m_i).\overrightarrow{h_i a_i})\Psi^1_{\perp,l,i}\end{aligned}$$

or by

$$\begin{aligned}\Pi_T f_{|K_l} = & \sum_{i=l}^{(l+1)\text{mod}3+1} f(a_i)\Psi^0_{l,i} + (\nabla_{\mathbf{x}} f(a_i).\overrightarrow{a_i a_k})\Psi^1_{l,i,k} \\ & + (\nabla_{\mathbf{x}} f(a_i).\overrightarrow{a_i a_j})\Psi^1_{l,i,j} - \left(n_i \frac{\partial f}{\partial \nu_i}(m_i)\right)\Psi^1_{\perp,l,i};\end{aligned}$$

where the basis functions $\{\Psi_l\}$ are detailed in Bernadou and Boisserie (1982) and Besse (2003). We have the equality $P_T = P_3$. For the proof of $\mathscr{C}^1$ continuity and the interpolation error estimates we refer to Ciarlet (1978).

We can also use other interpolation operators coming from finite element literature like the reduced Hermite element (HR), the reduced Hsieh-Clough-Tocher element (HCTR), the $\mathscr{C}^0$ Nielson element, and the Ganev-Dimitrov element (GD). Their description can be found in Bernadou and Boisserie (1982), Bernadou (1994), Besse (2003), and Ciarlet 1978 and a comparison of some of them are investigated in Besse (2003) and Besse and Sonnendrücker (2003).

Now we present the interpolation procedure we choose for each advection step.

2.2.1. Advection in Position Space

1. The reconstruction of f_h in position space is done using one of the Hermite interpolation schemes HCTC, HCTR, NC1, or GD.
2. The gradient $\nabla_{x,y} f_h$ which is not one of the degrees of freedom is computed by taking the derivatives of the reconstructed f_h.
3. The gradient $\nabla_{v_x, v_y} f_h$ is computed using a first-order Lagrange interpolation on the triangles or a modified second-order interpolation which preserves maxima and minima (see Berzins (2000)), as second-order Lagrange interpolation does not ensure the numerical stability of the gradients.

2.2.2. Advection in Velocity Space

1. The reconstruction of f_h can be performed using the NC1 or HCTR schemes or using symmetric Lagrange interpolation of any order on the regular grid. This is stable provided the reconstruction is used only in the middle cell which is the middle of the support of the local interpolation function (see Besse 2003).
2. The gradient $\nabla_{v_x, v_y} f_h$ which is not one of the degrees of freedom is computed by taking the derivatives of the reconstructed f_h or using a symmetric Lagrange interpolation (of any order) of the derivatives.
3. The gradient $\nabla_{x,y} f_h$ is computed using a Lagrange interpolation of any order.

Remark 1

Positivity of the distribution function and mass conservation property can be enforced a posteriori by optimizing between a low-order and a high-order interpolation. In fact, as described in Besse and Sonnendrücker (2003), for each advection step, we recover at the discrete level the following maximum principle:

$$f^n_{\min} \le f^{n+1}_{h,l} \le f^n_{\max}, \quad \forall l, \ \forall n \tag{4}$$

where

$$f^n_{\min} = \min_k\{f^n_{h,k}\}, \quad f^n_{\max} = \max_k\{f^n_{h,k}\},$$

and the conservation of mass, i.e.,

$$\sum_k f^n_{h,k}\mathcal{A}_k = \sum_k f^0_{h,k}\mathcal{A}_k = M_0, \tag{5}$$

where $\mathcal{A}_k$ is the area associated to the node N_k such that $\cup_k\mathcal{A}_k = \overline{\Omega}$. The final values $f^n_{h,k}$ of the distribution function at the node $N_k = (x_k, v_k)$ and at the time t^n are given by

$$f^n_{h,k} = \gamma^n_k f^n_{H_h,k} + (1 - \gamma^n_k)f^n_{L_h,k}$$

where $f^n_{H_h,k}$ and $f^n_{L_h,k}$ are, respectively, the linear approximation and a high-order approximation of f at the node N_k and at the time t^n. The coefficients γ^n_k are computed such that the properties (4) and (5) are satisfied (for more details see Besse and Sonnendrücker, 2003 and references therein).

Remark 2

In order to have good efficiency, two different data structures have been constructed for the advections in position and velocity space. Moreover, the code is parallelized by distributing the velocity variables on the processors for the position advection and by distributing the position variables for the velocity advection. This procedure allows no communication during each split step. However, a global redistribution of the data is necessary when going from one advection step to the next.

3. RESOLUTION OF THE POISSON EQUATION

Due to our Hermite interpolation method, we need to compute not only the electric field **E** but also its gradient ∇**E** at each time step. Because of this, special care needs to be taken for the resolution of the Poisson equation.

3.1. Poisson Equation with Periodic Boundary Conditions

We denote the computational domain by $\Omega = [x_0, x_0 + L_x[\times]y_0, y_0 + L_y]$, where L_x and L_y are, respectively, the periods in the directions x and y.

The Poisson problem consists in computing the electric field **E** such that

$$\begin{cases} \text{curl } \mathbf{E} = 0 \text{ in } \Omega \\ \text{div}\mathbf{E} = \dfrac{\rho}{\varepsilon_0} \text{ in } \Omega \\ \rho(x, y) = q \int_{\mathbb{R}^2} f(x, y, v_x, v_y) dv_x dv_y, \\ \mathbf{E}(x + L_x, y) = \mathbf{E}(x, y) \\ \mathbf{E}(x, y + L_y) = \mathbf{E}(x, y) \end{cases}$$

or, equivalently, introducing the scalar potential ϕ, in computing ϕ and **E** such that

$$\begin{cases} \mathbf{E} = -\nabla\phi \\ -\Delta\phi(x, y) = \dfrac{\rho}{\epsilon_0}(x, y) \\ \rho(x, y) = q \int_{\mathbb{R}^2} f\,(x, y, v_x, v_y) dv_x dv_y, \\ \phi(x + L_x, y) = \phi(x, y) \\ \phi(x, y + L_y) = \phi(x, y). \end{cases} \tag{6}$$

This last problem (6) can be solved numerically using a finite element technique based on P1, HR, NC1, HCTR, or HCTC elements on the periodic torus. To this aim, we introduce a variational formulation of the system (6) which reads:

Find $\phi \in H^1(\Omega)$ such that

$$\int_\Omega \nabla\phi.\nabla\psi d\Omega = \int_\Omega \rho\psi, \quad \forall\psi \in H^1(\Omega).$$

Remark 3

If Hermite-type finite elements like HR, NC1, HCTC, or HCTR are used, periodic boundary condition on $-\nabla\phi = E$ are also needed. Then ϕ belongs to $H^2(\Omega)$, where $H^m(\Omega)$ denotes the space of square integrable functions whose derivatives up to order m are square integrable.

As the Poisson problem is linear, the gradient of the electric field can be computed by taking the derivative of the Poisson equation. Denoting by $\mathbf{\Phi} = \nabla\phi$ the unknown vector, we get

$$\begin{cases} \nabla\mathbf{E} = -\nabla\mathbf{\Phi} \\ -\mathbf{\Delta\Phi} = \dfrac{\nabla\rho}{\varepsilon_0} \\ \nabla\rho(x, y) = q \int_{\mathbb{R}^2} \nabla_{x,y} f(x, y, v_x, v_y) dv_x dv_y, \\ \mathbf{\Phi}(x + L_x, y) = \mathbf{\Phi}(x, y) \\ \mathbf{\Phi}(x, y + L_y) = \mathbf{\Phi}(x, y). \end{cases} \tag{7}$$

The variational formulation associated to (6) reads:

Find $\mathbf{\Phi} \in (H^1(\Omega))^2$ such that

$$\int_\Omega \nabla\mathbf{\Phi} : \nabla\mathbf{\Psi} d\Omega = \int_\Omega \nabla\rho \cdot \mathbf{\Psi}, \quad \forall\mathbf{\Psi} \in (H^1(\Omega))^2$$

that can be solved by a traditional $P1$ (using linear basis functions) finite element method on a torus.

3.2. Poisson Equation with Dirichlet Boundary Conditions

3.2.1. Variational Formulation

The Poisson problem with Dirichlet boundary conditions consists in finding the electric field $\mathbf{E}$ such that

$$\begin{cases} \text{curl } \mathbf{E} = 0 \text{ in } \Omega \\ \text{div } \mathbf{E} = \dfrac{\rho}{\varepsilon_0} \text{in } \Omega \\ \rho(x, y) = q \int_{\mathbb{R}^2} f(x, y, v_x, v_y) dv_x dv_y, \\ \mathbf{E} \cdot \boldsymbol{\tau} = 0 \text{ on } \partial\Omega \end{cases} \tag{8}$$

where $\boldsymbol{\tau}$ is the tangent at the boundary $\partial\Omega$ of Ω. Problem (8) is equivalent to problem

$$\begin{cases} \mathbf{\Delta E} = \nabla \dfrac{\rho}{\epsilon_0} \text{ in } \Omega \\ \text{div } \mathbf{E} = \dfrac{\rho}{\epsilon_0} \text{ in } \partial\Omega \\ \text{curl } \mathbf{E} = 0 \text{ on } \partial\Omega \\ \mathbf{E} \cdot \boldsymbol{\tau} = 0 \text{ on } \partial\Omega \\ \rho(x, y) = q \int_{\mathbb{R}^2} f(x, y, v_x, v_y) dv_x dv_y. \end{cases} \tag{9}$$

Proof. Let us show first that (8) ⇒ (9). As

$$\mathbf{curl} \text{ curl } \mathbf{E} = \nabla(\text{div } \mathbf{E}) - \mathbf{\Delta E} \tag{10}$$

using the first two equations of (8) we get

$$\begin{aligned} 0 &= \mathbf{curl} \text{ curl } \mathbf{E} \\ &= \frac{\nabla\rho}{\varepsilon_0} - \mathbf{\Delta E} \end{aligned}$$

whence the first equation of (9). If div $\mathbf{E} = \rho/\varepsilon_0$ in $\mathscr{C}(\Omega)$, then div $\mathbf{E} = \rho/\varepsilon_0$ on $\partial\Omega$. In the same way curl $\mathbf{E} = 0$ in $\mathscr{C}(\Omega)$ then curl $\mathbf{E} = 0$ on $\partial\Omega$. Let us prove now that (9) ⇒ (8). Using relation (10), the first equation of (9) rewrites

$$\mathbf{curl} \text{ curl } \mathbf{E} = \nabla\left(\text{div } \mathbf{E} - \frac{\rho}{\varepsilon_0}\right). \tag{11}$$

Let $\mathbf{F}$ be a test function belonging to $\mathbf{H}(\mathrm{curl},\ \mathrm{div};\ \Omega) = \mathbf{H}(\mathrm{curl};\ \Omega) \cap \mathbf{H}(\mathrm{div};\ \Omega)$, where

$$\mathbf{H}(\mathrm{curl};\Omega) = \left\{\mathbf{u} \in \mathbf{L}^2(\Omega),\ \mathrm{curl}\ \mathbf{u} \in L^2(\Omega)\right\},\ \mathbf{H}(\mathrm{div};\Omega)$$
$$= \left\{\mathbf{u} \in \mathbf{L}^2(\Omega),\ \mathrm{div}\ \mathbf{u} \in L^2(\Omega)\right\}$$

with $\mathbf{L}^2(\Omega) = L^2(\Omega) \otimes\ L^2(\Omega)$.

Multiplying (11) by a test function $\mathbf{F}$ and using the Green formulae

$$\int_\Omega \mathbf{u} \cdot \mathbf{curl}\,\varphi\ d\Omega - \int_\Omega \mathrm{curl}\ \mathbf{u}\,\varphi\ d\Omega = \int_{\partial\Omega} \mathbf{u} \cdot \boldsymbol{\tau}\varphi\, d\Gamma, \qquad (12)$$
$$\forall \mathbf{u} \in \mathbf{H}(\mathrm{curl};\ \Omega),\ \ \varphi \in H^1(\Omega),$$

and

$$\int_\Omega \mathbf{u} \cdot \nabla\varphi\ d\Omega + \int_\Omega \mathrm{div}\,\mathbf{u}\,\varphi\ d\Omega = \int_{\partial\Omega} \mathbf{u} \cdot \boldsymbol{\nu}\ \varphi\, d\Gamma, \qquad (13)$$
$$\forall \mathbf{u} \in \mathbf{H}(\mathrm{div};\ \Omega),\ \ \varphi \in H^1(\Omega),$$

with $\boldsymbol{\nu}$ the normal vector at the boundary $\partial\Omega$, we obtain

$$\int_\Omega \mathrm{curl}\,\mathbf{F}\,\mathrm{curl}\ \mathbf{E}\ d\Omega\ + \int_\Omega \mathrm{div}\,\mathbf{F}\,\mathrm{div}\,\mathbf{E}\,d\Omega\ + \int_{\partial\Omega} \mathbf{F} \cdot \boldsymbol{\tau}\ \mathrm{curl}\ \mathbf{E}\ d\Gamma$$
$$= \int_{\partial\Omega} \mathbf{F} \cdot \boldsymbol{\nu}\ \mathrm{div}\,\mathbf{E}\ d\Gamma - \int_\Omega \nabla\rho \cdot \mathbf{F}\,d\Omega. \qquad (14)$$

Using again the Green formula (13), (14) becomes

$$\int_\Omega \mathrm{curl}\,\mathbf{F}\,\mathrm{curl}\,\mathbf{E}\ d\Omega + \int_\Omega \mathrm{div}\,\mathbf{F}\,\mathrm{div}\,\mathbf{E}\,d\Omega + \int_{\partial\Omega} \mathbf{F} \cdot \boldsymbol{\tau}\ \mathrm{curl}\ \mathbf{E}\ d\Gamma$$
$$= \int_\Omega \mathrm{div}\,\mathbf{F}\,\rho\,d\Omega + \int_{\partial\Omega} \mathbf{F} \cdot (\mathrm{div}\mathbf{E} - \rho)d\Gamma \qquad (15)$$

Using relations $\mathrm{div}\,\mathbf{E} = \rho/\epsilon_0$ on $\partial\Omega$ and curl $\mathbf{E} = 0$ on $\partial\Omega$ we finally obtain

$$\int_\Omega \mathrm{curl}\,\mathbf{F}\,\mathrm{curl}\,\mathbf{E}\,d\Omega + \int_\Omega \mathrm{div}\ \mathbf{F}\,\mathrm{div}\,\mathbf{E}\,d\Omega = \int_\Omega \mathrm{div}\,\mathbf{F}\ \rho\,d\Omega. \qquad (16)$$

If $\mathbf{F} = \nabla\psi$, with $\psi\ \in\ H^2(\Omega)$ then (16) becomes

$$\int_\Omega \Delta\psi \left(\mathrm{div}\,\mathbf{E} - \frac{\rho}{\epsilon_0}\right) d\Omega = 0 \Longrightarrow \mathrm{div}\,\mathbf{E} = \frac{\rho}{\epsilon_0}\ \text{ in } L^2(\Omega)$$

If $\mathbf{F} = \mathbf{curl}\varphi$, with $\varphi\ \in\ H^2(\Omega)$ then (15) becomes

$$\int_\Omega \Delta\varphi\ \mathrm{curl}\,\mathbf{E}\,d\Omega = 0 \Longrightarrow \mathrm{curl}\,\mathbf{E} = 0 \text{ in } L^2(\Omega).$$

Problem (9) can be solved numerically using a finite element method using HR, NC1, HCTR, or HCTC elements. To this aim the following variational formulation is used:
Find $\mathbf{E} \in \mathbf{X}$ such that

$$\int_\Omega \operatorname{curl} \mathbf{F} \operatorname{curl} \mathbf{E} \, d\Omega + \int_\Omega \operatorname{div} \mathbf{F} \operatorname{div} \mathbf{E} \, d\Omega = \int_\Omega \operatorname{div} \mathbf{F} \rho d\Omega, \forall \mathbf{F} \in \mathbf{H}(\text{curl, div}; \Omega) \tag{17}$$

with

$$\mathbf{X} = \{\mathbf{u} \in H^1(\Omega) \otimes H^1(\Omega), \ \mathbf{E} \cdot \boldsymbol{\tau} = 0 \text{ on } \partial\Omega, \partial_s(\mathbf{E} \cdot \boldsymbol{\tau}) = 0 \text{ on } \partial\Omega\}$$

where s is the curvilinear abscissa.

Remark 4

Note that when using Hermite-type finite elements, where both function values and values of the derivatives belong to the degrees of freedom, then the values of both $\mathbf{E}$ *and* $\nabla\mathbf{E}$ *are directly obtained from the finite element solution.*

3.2.2. Boundary Conditions

Using Hermite-type finite elements requires the additional use of the boundary conditions

$$\mathbf{E} \cdot \boldsymbol{\tau} = 0 \text{ on } \partial\Omega, \partial_s(\mathbf{E} \cdot \boldsymbol{\tau}) = 0 \text{ on } \partial\Omega.$$

The resolution of the linear system stemming from the variational formulation (17) is performed in the Cartesian frame with basis vectors $(e_1(x^1, x^2), e_2(x^1, x^2))$ where (x^1, x^2) are the Cartesian coordinates. Only the boundary conditions for the electric field cannot be expressed in a simple way in the basis $(e_1(x^1, x^2), e_2(x^1, x^2))$. However, they have a simple expression in the moving Frenet frame (τ, ν) associated to the domain boundary illustrated by Figure 1. Let us introduce the local change of basis $(a_1(u^1, u^2), a_2(u^1, u^2))$ with $\tau = a_1(u^1, 0)$ and $\nu = a_2(u^1, 0)$ (see Figure 1). In the Frenet frame, for a point belonging to the boundary, the boundary conditions can be expressed by

$$E^\tau = 0, \quad \partial_{u^1} E^\tau = 0.$$

Denote by N_{fr} the set of all boundary points.

Then for all $i \in N_{fr}$ the unknown vector Λ_i whose expression in the basis $(e_1(x^1, x^2), e_2(x^1, x^2))$ is

$$\Lambda_i = (E_i^{x^1}, E_i^{x^2}, \partial_{x^1} E_i^{x^1}, \partial_{x^2} E_i^{x^1}, \partial_{x^1} E_i^{x^2}, \partial_{x^2} E_i^{x^2})^T$$

becomes in the basis $(\tau, \nu) = (a_1(u^1, 0), a_2(u^1, 0))$ the unknown vector Ξ_i

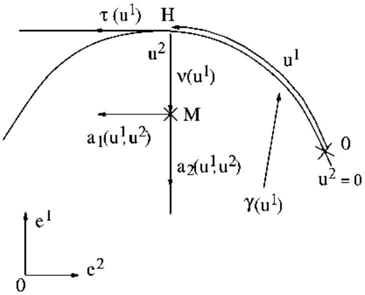


Figure 1. Change of variables: the (x^1, x^2)-Cartesian coordinate system versus the (u^1, u^2)-curvilign coordinate system.

whose expression is

$$\Xi_i = (E_i^\tau, E_i^\nu, \partial_{u^1} E_i^\tau, \partial_{u^2} E_i^\tau, \partial_{u^1} E_i^\nu, \partial_{u^2} E_i^\nu)^T.$$

Hence, we need the transformation matrix R_i enabling to pass from Ξ_i to Λ_i, that is such that we have

$$\Lambda_i = R_i \Xi_i, \;\; i \in N_{fr}.$$

The global change of variables then reads

$$\Lambda = R\Xi,$$

where

$$R = \begin{pmatrix} \ddots & 0 & 0 & 0 & 0 \\ 0 & R_i & 0 & 0 & 0 \\ 0 & 0 & \ddots & 0 & 0 \\ 0 & 0 & 0 & Id_6 & 0 \\ 0 & 0 & 0 & 0 & \ddots \end{pmatrix}$$

The matrix Id_6 is the 6×6 identity matrix and represents the identity transform for the points that are inside the domain. The initial linear system which reads

$$(K_{<\text{curl},\text{curl}>} + M_{<\text{div},\text{div}>})\Lambda = L_{\text{div}} F_\rho$$

where $K_{<\text{curl},\text{curl}>}$, $M_{<\text{div},\text{div}>}$, and L_{div}, represent, respectively, an approximation of the bilinear forms

$$\langle \cdot, \cdot \rangle_{\text{curl},\text{curl}} = \int_\Omega \text{curl}(\cdot)\,\text{curl}(\cdot)\, d\Omega, \;\; \langle \cdot, \cdot \rangle_{\text{div},\text{div}} = \int_\Omega \text{div}(\cdot)\,\text{div}(\cdot)\; d\Omega,$$

and of the linear form

$$\mathcal{L}_{\text{div}}(\cdot) = \int_{\Omega} \rho \text{div}(\cdot)\, d\Omega$$

becomes

$$R^T(K_{<\text{curl},\text{curl}>} + M_{<\text{div},\text{div}>})R\Xi = R^T L_{\text{div}} F_{\rho}.$$

We now need to determine R_i. To this aim, we use tensor calculus and the fact that the absolute differential dE does not depend on any bases. A point x can be expressed in the following way $x = M(u^1, u^2) = \gamma\,(u^1) + u^2\nu\,(u^1)$. The parameterization of the point x in the curvilign coordinate system (u^1, u^2) is illustrated by the Figure 1. Then, thanks to the following classical definition

$$a_i = \frac{\partial x}{\partial u^i} = \frac{\partial M}{\partial u^i} \text{ and } a^i a_j = \delta^i_j = \begin{cases} 1 \text{ if } i = j \\ 0 \text{ if } i \neq j \end{cases} \tag{18}$$

We can compute the covariant components a_i and the contravariant components a^i of the basis vectors. On the one hand, we have

$$E = E^{\alpha} a_{\alpha}$$

whence the expression of the absolute differential dE

$$dE = D_u E^{\alpha} a_{\alpha} = D_{u^{\beta}} E^{u^{\alpha}} du^{\beta} a_{\alpha} \tag{19}$$

where $D_{u^{\beta}} E^{u^{\alpha}}$ is the covariant derivative. On the other hand, we have

$$E = E^{x_i} e_i,$$

where the expression of the absolute derivative dE is

$$dE = D_x E^{x_j} e_j = D_{x^i} E^{x_j} dx^i e_j = \partial_{x^i} E^{x_j} dx^i e_j. \tag{20}$$

Identifying (19) and (20) we obtain

$$\partial_{x^i} E^{x^j} = D_{u^{\beta}} E^{u^{\alpha}} \frac{\partial u^{\beta}}{\partial x^i} \frac{\partial x^j}{\partial u^{\alpha}}. \tag{21}$$

With the notations

$$E^1(u^1, 0) = E^{\tau},\ \ E^2(u^1, 0) = E^{\nu}, \partial_{u^1} E^1(u^1, 0) = \partial_{u^1} E^{\tau},$$

$$\partial_{u^2} E^1(u^1, 0) = \partial_{u^2} E^{\tau},\ \ \partial_{u^1} E^2(u^1, 0) = \partial_{u^1} E^{\nu}, \partial_{u^2} E^2(u^1, 0) = \partial_{u^2} E^{\nu},$$

and relation (21) we find for R_i

$$R_i(\nu_i, \mathfrak{R}_{c_i}) = \begin{pmatrix} \nu_{y,i} & \nu_{x,i} & 0 & 0 & 0 & 0 \\ -\nu_{x,i} & \nu_{y,i} & 0 & 0 & 0 & 0 \\ 0 & -\frac{\nu_{y,i}^2}{\mathfrak{R}_{c_i}} & \nu_{y,i}^2 & \nu_{x,i}\nu_{y,i} & \nu_{x,i}\nu_{y,i} & \nu_{x,i}^2 \\ -\frac{1}{\mathfrak{R}_{c_i}} & \frac{\nu_{x,i}\nu_{y,i}}{\mathfrak{R}_{c_i}} & -\nu_{x,i}\nu_{y,i} & \nu_{y,i}^2 & -\nu_{x,i}^2 & \nu_{x,i}\nu_{y,i} \\ \frac{1}{\mathfrak{R}_{c_i}} & \frac{\nu_{x,i}\nu_{y,i}}{\mathfrak{R}_{c_i}} & -\nu_{x,i}\nu_{y,i} & -\nu_{x,i}^2 & \nu_{y,i}^2 & \nu_{x,i}\nu_{y,i} \\ 0 & -\frac{\nu_{x,i}^2}{\mathfrak{R}_{c_i}} & \nu_{x,i}^2 & -\nu_{x,i}\nu_{y,i} & -\nu_{x,i}\nu_{y,i} & \nu_{y,i}^2 \end{pmatrix}$$

where $\mathfrak{R}_{c_i}$ is curvature radius at the point P_i, $i \in N_{fr}$.

4. NUMERICAL VALIDATION

We validate our code using two classical problems of plasma physics and beam physics. The first test case is the linear Landau damping, the second is the evolution of a gaussian beam in a uniform focusing channel.

4.1. Linear Landau Damping

We here solve the system

$$\frac{\partial f}{\partial t} + \mathbf{v} \cdot \nabla_{\mathbf{x}} f + \mathbf{E}_s \cdot \nabla_{\mathbf{v}} f = 0,$$

$$\mathbf{E}_s(t, x, y) = -\nabla \phi(t, x, y), \ -\Delta\phi(t, x, y) = \int_{\mathbb{R}^2} f(t, x, y, v_x, v_y) dv_x dv_y - 1.$$

where $\mathbf{x} = (x, y)$, $\mathbf{v} = (v_x, v_y)$ and $\mathbf{E}_s = (E_{x_s}, E_{y_s})$.

The initial data is

$$f(0, x, y, v_x, v_y) = \frac{1}{2\pi} e^{-(v_x^2+v_y^2)/2}(1 + \alpha \cos(k_x x)\cos(k_y y)),$$
$$\forall (x, y) \in [0, L_x] \times [0, L_y], \ (v_x, v_y) \in [-v_{\max}, v_{\max}]^2$$

where $\alpha = 0.05$, $v_{\max} = 7.5$, and $T_{final} = 22\omega_p^{-1}$. The wave numbers k_x and k_y are equal to 0.5. The size of the periodic computation domain is chosen according to the wave numbers such that $L_x = L_y = 2\pi/k_x = 2\pi/k_y = 4\pi$. The position space discretization consists in an unstructured mesh of 7938 triangles (~64 points in x and 64 points in y) and of 32258 right angle triangles in velocity space (~128 points in vx by 128 points in v_y). A HCTC finite-element interpolation is used in (x, y) and a NC1 scheme in (v_x, v_y). We choose $\Delta t = 1/8$ for the time discretization step. The result is displayed on Figure 2. The evolution of the two components of the electric field are

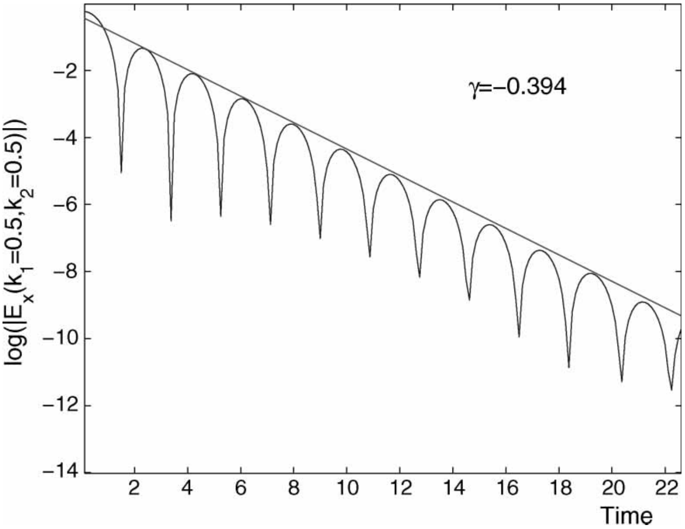


Figure 2. Evolution of the mode $\widehat{E}_x(k_x = 0.5, k_y = 0.5)$.

identical as the initial data is symmetric with respect to the x and y axes. The $\widehat{E}_x(k_x = 0.5, k_y = 0.5)$ mode that carries the potential energy during the linear phase decreases exponentially with a damping rate $\gamma = -0.394$ and a frequency $\omega = 1.69$, exactly as predicted by the linear Landau theory.

4.2. Gaussian Beam

The initial data is

$$f_0(x, y, v_x, v_y) = \frac{n_0}{2\pi v_{th}^2 \pi a^2} e^{-(v_x^2+v_y^2)/2v_{th}^2} e^{-(x^2+y^2)/2R^2}$$

with $a = 1$. In order to have an adapted beam, we compute v_{th}, ω_0, n_0 and R using the concept of equivalent beams, which consists in matching the second-order moments of the beam to those of a Kapchinsky-Vladimirsky (KV) beam which can be matched analytically (Kapchinsky and Vladimirsky 1959; Davidson and Qin 2001). The K-V distribution is of the form

$$f_0(x, y, x', y') = \delta\left(\frac{x^2}{a^2} + \frac{y^2}{a^2} + \frac{x'^2}{a'^2} + \frac{y'^2}{a'^2} - 1\right).$$

This distribution is a steady state solution of the Vlasov equation for which the self-consistent electric field is linear. Hence, the total electric field reads

$$\mathbf{E}_s(t, \mathbf{x}) + \mathbf{E}_a(t, \mathbf{x}) = -\omega^2 \mathbf{x},$$

where $\mathbf{E}_s$ is the solution of the Poisson Equation (2) and $\mathbf{E}_a = -\omega_0^2\mathbf{x}$. Defining the tune depression $\eta = \omega/\omega_0$, we match the beam by setting

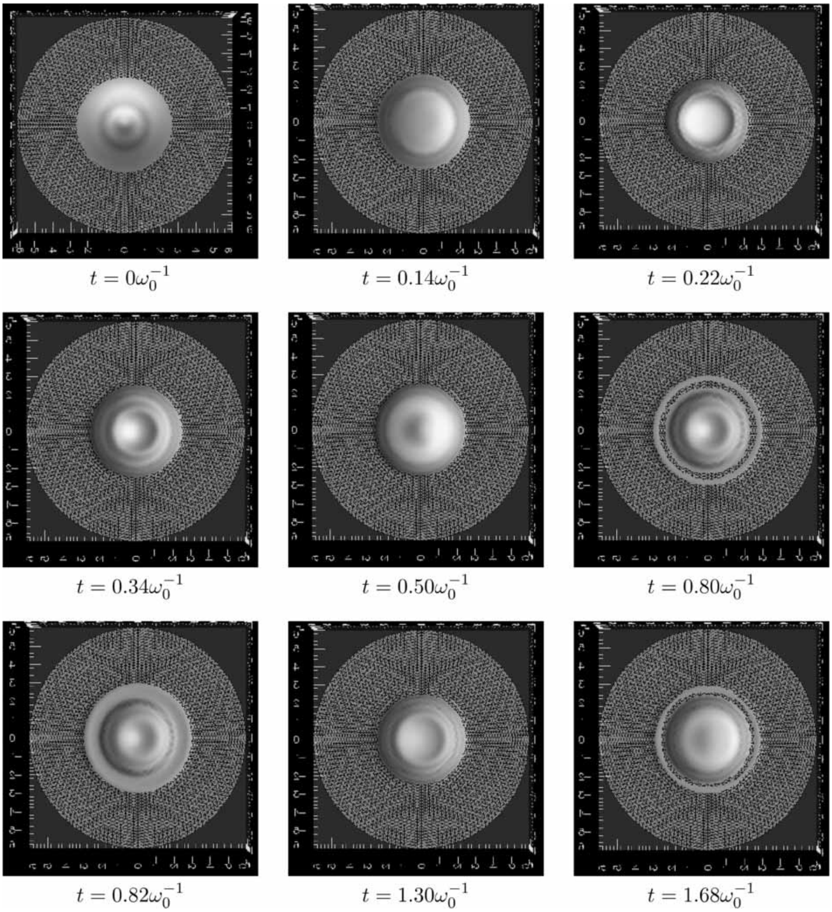
$$R^2 = (x^f_{rms})^2 = \frac{\int_{\mathbb{R}^4} x^2 f' d\mathbf{x}' d\mathbf{v}'}{\int_{\mathbb{R}^4} f' d\mathbf{x}' d\mathbf{v}'} = (x^{\text{K-V}}_{rms})^2 = \frac{a^2}{4},$$

$$R^2 = (y^f_{rms})^2 = \frac{\int_{\mathbb{R}^4} y^2 f' d\mathbf{x}' d\mathbf{v}'}{\int_{\mathbb{R}^4} f' d\mathbf{x}' d\mathbf{v}'} = (y^{\text{K-V}}_{rms})^2 = \frac{a^2}{4},$$

$$\frac{v_{th}^2}{u_z^2} = (v^f_{x_{rms}})^2 = \frac{\int_{\mathbb{R}^4} x'^2 f' d\mathbf{x}' d\mathbf{v}'}{\int_{\mathbb{R}^4} f' d\mathbf{x}' d\mathbf{v}'} = (v^{\text{K-V}}_{x_{rms}})^2 = \frac{a'^2}{4} = \frac{(ka)^2}{4},$$

$$\frac{v_{th}^2}{u_z^2} = (v^f_{y_{rms}})^2 = \frac{\int_{\mathbb{R}^4} y'^2 f' d\mathbf{x}' d\mathbf{v}'}{\int_{\mathbb{R}^4} f' d\mathbf{x}' d\mathbf{v}'} = (v^{\text{K-V}}_{y_{rms}})^2 = \frac{a'^2}{4} = \frac{(ka)^2}{4},$$

where $\omega = ku_z$, and u_z being the longitudinal velocity of the beam that we have supposed constant. Setting $\eta = 1/4$ and $v_{th} = 1$ and computing the matching condition for the KV beam which yields $\omega_0 = \sqrt{\frac{n_0}{2(1-\eta^2)}}$, the model is perfectly determined. The physical space which is a disk of radius 6 is discretized by 6084 triangles and the velocity space is a $[-v_{x_{\min}}, -v_{x_{\max}}] \times [-v_{y_{\min}}, -v_{y_{\max}}]$ square with $v_{x_{\min}} = v_{x_{\max}} = v_{y_{\min}} = v_{y_{\max}} = 9$ which is discretized by 32258 triangles. We set Dirichlet boundary conditions for the Poisson problem as it is described in section 3.2. The triangulation must be regular in the sense of finite element, i.e., there exists a constant σ such that $h_T/\rho_T \leq \sigma$, $\forall T \in \mathscr{T}_h$, where h_T and ρ_T, respectively, stand for the diameter of the smallest ball which contains T and the diameter of the biggest ball contained in T. We set the time step $\Delta t = T/250$ where $T = 2\pi\omega_0^{-1}$. Even if the time step is not restricted by a stability condition (like CFL condition), here we choose a small time step in order to keep good accuracy. The computation runs on 128 processors of a Compaq Alphaserver ES45 1GHz and takes around 5 hours when the final time step $T_{final} = 2T$. The algorithm is inevitably slower on unstructured meshes than on structured grids because characteristic tracking is more complex and expensive on unstructured meshes than on structured grids. In the case of structured grids, the characteristic tracking cost is reduced to N divisions if N is the dimension of the phase space, whereas in the case of unstructured meshes the cost is equivalent to $3dn_T$ where d is the cost of a N-dimensional determinant (whose sign indicates if the characteristic origin is contained in the element or in which direction you have to search the element) and n_T is the number of element T you have to cross in order to reach the origin of the characteristic. Like for the Landau damping case test, a HCTC finite-element interpolation operator is used in (x, y) and a NC1 reconstruction in (v_x, v_y). Looking at Figure 3, we notice that a hole starts to form at the center of the beam, so that the density becomes higher at the edge of the beam and the particles start moving inwards. Then a density wave is



$t = 0\omega_0^{-1}$ $t = 0.14\omega_0^{-1}$ $t = 0.22\omega_0^{-1}$

$t = 0.34\omega_0^{-1}$ $t = 0.50\omega_0^{-1}$ $t = 0.80\omega_0^{-1}$

$t = 0.82\omega_0^{-1}$ $t = 1.30\omega_0^{-1}$ $t = 1.68\omega_0^{-1}$

Figure 3. Evolution of the density ρ (t, x, y).

created with particles moving from the inner part to the outer part of the beam and in again.

Starting around time $t = 0.80\omega_0^{-1}$ and again at $t = 1.68\omega_0^{-1}$, a halo surrounding the core of the beam appears. The halo appears at radius $r = 2.6$ and its density represents 3 to 5% of the core density. The same result has been already obtained in Besse and Sonnendrücker (2003) with a three-dimensional code in the variable (r, v_r, v_θ) in the case of the axisymetric geometry. Other simulations related to the halo formation can be found in Sonnendrücker et al.

5. CONCLUSION

A new method for solving the Vlasov equation on a mesh of phase space, which is completely unstructured in configuration space, has been introduced and validated on some problems of plasma physics and transport of charged particle beams. This method relies on semi-Lagrangian principle (the characteristic curves associated to the transport operator are integrating backward), time splitting techniques (the full dimensional complex transport operator is decomposed in several transport operators of lower dimension that are more simple to solve) and uses finite element (HCT-C) and/or CAGD (computer-aided geometric design) (NC1) reconstruction schemes. The method involves the propagation of the jacobian of the distribution function and leads to the computation of the gradient of the force field (here the electric field) for which the boundary conditions have been treated carefully. The method works well and should be useful for some problems with specific geometries. This method can also be extended in an adaptive mesh refinement version which will improve result accuracy and computational effectiveness since Vlasov equation develops multi-scale phenomena. The cost of this method is probably more expensive on unstructured meshes than on structured grids because of the characteristic tracking problem. Nevertheless, unlike classical Eulerian methods such as, for example, finite volume, finite element, finite differences, and discontinuous-Galerkin methods, semi-Lagrangian methods are not restricted by CFL condition which can be very restrictive and leads to a high computational cost when we solve the Vlasov equation in high velocity regions. A good compromise is to use our method (semi-Lagrangian scheme with propagation of the partial derivatives of *f*) on unstructured meshes near the curved boundaries and on structured or adaptive grids (adaptive dyadic grids generated, for example, by wavelet multiresolution analysis, see Besse et al. (2001) elsewhere in a self-consistent code. Besides, our method can easily be used to solve the Vlasov-Maxwell system provided that we have a Maxwell solver which supplies the gradient of the electromagnetic field. This numerical scheme needs important computing resources; that is why the code is parallelized.

REFERENCES

Bernadou, M. (1994). *Méthodes d'éléments finis pour les problèmes de coques minces. Recherche en Mathématiques Appliquées*. Ciarlet, P. G., Lions, J-.L., Eds., Paris: Masson.

Bernadou, M., Boisserie, J. M. (1982). *The finite Method in Thin Shell Theory: Application to Arch Dam Simulations*. Boston: Birkhäuser.

Berzins, M. (2000). A data-bounded quadratic interpolant on triangles and tetrahedra. *SIAM, J. Comput.* 22:177–197.

Besse, N. (2003). *Etude mathématique et numérique de l'équation de Vlasov non linéaire sur des maillages non structurés de l'espace des phases*. PhD thesis. Institut de Recherche Mathématique Avancée, Université Louis Pasteur: Strasbourg.

Besse, N., Convergence of a high-order semi-lagrangian scheme with propagation of the gradients for the Vlasov-Poisson system, submitted.

Besse, N., Sonnendrücker, E. (2003). Semi-Lagrangian schemes for the Vlasov equation on an unstructured mesh of phase space. *J. Comput. Phys.* 191:341–376.

Besse, N., Filbet, F., Gutnic, M., Paun, I., Sonnendrücker, E. (2001). An adaptive numerical method for the Vlasov equation based on a multiresolution analysis. In *Numerical Mathematics and Advanced Applications ENUMATH 2001*. Brezzi, F., Buffa, A., Escorsaro, S., Murli, A., Eds.; Springer, pp. 437–446.

Cheng, C. Z., Knorr, G. (1976). The integration of the Vlasov equation in configuration space. *J. Comput. Phys.* 22:330–348.

Ciarlet, P. G. (1978). *The Finite Element Method for Elliptic Problems*. Amsterdam: North-Holland.

Davidson, R. C., Qin, H. (2001). *Physics of Intense Charged Particle Beams in High Energy Accelerators*. Singapore: Imperial College Press, World Scientific.

Feix, M. R., Bertrand, P., Ghizzo, A. (1994). Eulerian codes for the Vlasov equation. *Series on Advances in Mathematics for Applied Sciences* 22, *Kinetic Theory and Computing*.

Fijalkow, E. (1999). A numerical solution to the Vlasov equation. *Comput. Phys. Communications* 116:319–328.

Filbet, F., Sonnendrücker, E. (2003). Comparison of Eulerian Vlasov solvers. *Comput. Phys. Comm.* 150:247–266.

Filbet, F., Sonnendrücker, E., Bertrand, P. (2001). A positive and flux conservative scheme for the numerical resolution of the Vlasov equation. *J. Comput. Phys.* 172:166–187.

Ghizzo, A., Bertrand, P., Shoucri, M., Johnston, T. W., Filjakow, E., Feix, M. R. (1990). A Vlasov code for the numerical simulation of stimulated Raman scattering. *J. Comput. Phys.* 90:431.

Kapchinsky, I. M., Vladimirsky, V. V. (1959). Limitations of proton beam current in a strong focusing linear accelerator associated with the beam space charge. In: Proceedings of the Conference on High Energy Accelerators and Instrumentation, Geneva: CERN, p. 274.

Nakamura, T., Yabe, T. (1999). Cubic interpolated propagation scheme for solving the hyper-dimensional Vlasov-Poisson equation in phase space. *Comput. Phys. Communications* 120:122–154.

Nielson, G. (1980). Minimum norm interpolation in triangles. *SIAM J. Numer. Anal.* 17:44–62.

Sonnendrücker, E., Barnard, J., Friedman, A., Grote, D., Lund, S. (2001). Simulation of heavy ion beams with a semi-Lagrangian Vlasov solver. *Nuclear Inst. and Methods in Physics Research*, Section A, 464:470–476.

Sonnendrücker, E., Roche, J., Bertrand, P., Ghizzo, A. (1999). The semi-Lagrangian method for the numerical resolution of Vlasov equations. *J. Comput. Phys.* 149: 201–220.