

A DIRECT AND ACCURATE ADAPTIVE SEMI-LAGRANGIAN SCHEME FOR THE VLASOV-POISSON EQUATION

MARTIN CAMPOS PINTO*,

*IRMA (CNRS and Université Louis Pasteur)

7, rue René Descartes
67084 Strasbourg Cedex
France

e-mail: campos@math.u-strasbg.fr

This article aims at giving a simplified presentation of a new adaptive semi-Lagrangian scheme for solving the 1+1-dimensional Vlasov-Poisson system, that has been developed in 2005 with Michel Mehrenberger and first described in (Campos Pinto and Mehrenberger, submitted). The main steps of the analysis are also given, which yield the first error estimate for an adaptive scheme in the context of the Vlasov equation. This article focuses on a key feature of our method, which is a new algorithm to transport multiscale meshes along a smooth flow, in a way that can be said *optimal* in the sense that it satisfies both accuracy and complexity estimates which are likely to lead to optimal convergence rates for the whole numerical scheme. From the regularity analysis of the numerical solution and how it gets transported by the numerical flow, it is shown that the accuracy of our scheme is monitored by a prescribed tolerance parameter ε which represents the local interpolation error at each time step. As a consequence, the numerical solutions are proved to converge in L^∞ towards the exact ones as ε and Δt tend to zero, and in addition to the numerical tests presented in (Campos Pinto and Mehrenberger, submitted), some complexity bounds are established which are likely to prove the optimality of the meshes.

Keywords: fully adaptive scheme, semi-Lagrangian method, Vlasov-Poisson system, error estimates, convergence rates, optimal transport of adaptive multiscale meshes.

1 Introduction

In one space dimension, the Vlasov equation

$$\begin{aligned} \partial_t f(t, x, v) + v \partial_x f(t, x, v) + E(t, x) \partial_v f(t, x, v) &= 0, \\ f(0, \cdot, \cdot) &= f_0 \in C_c^0(\mathbb{R}^2), \quad t \geq 0, \quad x, v \in \mathbb{R} \end{aligned} \quad (1)$$

coupled with the Poisson equation

$$\partial_x E(t, x) = \int_{\mathbb{R}} f(t, x, v) dv \quad (2)$$

describes the evolution of a collisionless electron gas essentially subject to its own electric field, and can be seen as a reduced model for the tri-dimensional Vlasov-Maxell system that describes the dynamic behavior of hot plasmas and particle beams. Here a background ion distribution is assumed to be uniform and constant, and the magnetic effects are neglected in the Lorentz forces. The variables x and v respectively standing for the space position and the velocity, $f(t, \cdot, \cdot)$ and $E(t, \cdot)$ denote the electron phase space density and the electric field at time t .

In order to save computational resources while approximating the complex and thin structures that may

appear in the solutions of the Vlasov equation, several adaptive schemes have been proposed in the past few years, see in particular (Besse *et al.*, 2001 ; Gutnic *et al.*, 2004 and Sonnendrücker *et al.*, 2004), where the authors use moving phase-space grids or interpolatory wavelets. Inspired from the semi-Lagrangian method of (Cheng and Knorr, 1976), later revisited by (Sonnendrücker *et al.*, 1999), these schemes gave encouraging results in practice, but no one was *proven* to be more efficient than the uniform ones.

Our main objective was therefore to derive from these adaptive strategies a new error estimate that would prove their superiority over any uniform method. By introducing a new adaptive scheme based on multiscale (hierarchical) finite element bases, this objective has been achieved, and we have proven a rigorous error estimate, together with a partial complexity bound, for the adaptive approximation of problem (1)-(2).

The outline is as follows: in section 2, the use of adaptive strategies for approximating solutions of non uniform smoothness - already described by many authors in the context of conservation laws, see *e.g.* (Cohen *et*

al., 2003 or Roussel *et al.*, 2003) - is first justified with basic arguments, and a practical algorithm for designing adaptive P_1 finite element approximations is given. In section 3, the principle of the semi-Lagrangian method for solving the Vlasov-Poisson equation is recalled, as well as the error analysis given by (Besse, 2004) in the uniform setting. Section 4 is then devoted to our adaptive scheme. We first address one of the major issues in designing adaptive methods for evolution problems, namely their ability to predict meshes fine enough to achieve a given accuracy without being much more refined than what is necessary. We then describe the key feature of our numerical scheme, that is a new algorithm that transport multiscale meshes along smooth flows in a way that can be said *optimal* in the sense of remark 5. A global error estimate is eventually stated, together with a partial result on the cardinality of the adaptive meshes. For numerical results confirming these theoretical results and exhibiting optimal adaptive meshes obtained in practice, the reader is referred to (Campos Pinto and Mehrenberger, submitted).

2 Adaptive finite elements

In this section, we recall how adaptive strategies can be proven to be more efficient than uniform ones in terms of convergence rates, and we propose a simple algorithmic setting to perform P_1 adaptive approximations (for a comprehensive introduction to adaptive approximation, we refer to the excellent review article (DeVore, 1998)).

2.1 Uniform vs. adaptive meshes

We consider here the problem of interpolating some continuous function g known on the unit square $\Omega = [0, 1]^2$. Since we are to use P_1 finite elements, that is piecewise affine interpolations on conforming triangulations of Ω , we shall think of two different approaches.

The first one consists in using a uniform mesh \mathcal{K}_h made of shape regular triangles of diameter $\mathcal{O}(h)$, *i.e.* triangles K that contain and that are contained in balls of respective diameter d_K and d'_K , with

$$Ch \leq d_K \leq d'_K \leq C'h.$$

with absolute constants C and C' . If g belongs to the space $W^{2,\infty}(\Omega)$, *i.e.* if it is bounded on Ω and if its second order derivatives are also bounded on Ω , the classical estimate

$$\|(I - P_K)g\|_{L^\infty(K)} \leq Ch^2|g|_{W^{2,\infty}(K)} \quad (3)$$

(where $|g|_{W^{2,\infty}(K)} := \max\{\|D^\alpha g\|_{L^\infty(K)} : |\alpha| = 2\}$ denotes the usual Sobolev semi-norm), valid on any shape

regular triangle K of diameter $\mathcal{O}(h)$, allows to bound the global interpolation error on \mathcal{K}_h by

$$\|(I - P_h)g\|_{L^\infty(\Omega)} \leq Ch^2|g|_{W^{2,\infty}(\Omega)}. \quad (4)$$

Since the cardinality $\#(\mathcal{K}_h)$ of this triangulation is on the order of h^{-2} , the above estimate yields

$$\|(I - P_h)g\|_{L^\infty(\Omega)} \leq C\#(\mathcal{K}_h)^{-1}|g|_{W^{2,\infty}(\Omega)}. \quad (5)$$

In the perspective of estimating the (a priori) efficiency of a given approximation method, the above inequality, often referred to as a convergence rate, is a key information which somehow expresses the trade-off between the *accuracy* of the numerical approximations and their *complexity*, closely related to their computational cost. In particular, relation (5) allows

- to impose a maximal cardinality on the meshes, while guaranteeing the accuracy of the interpolations;
- or to prescribe a given accuracy on the interpolations, while giving a complexity bound for the associated meshes.

To improve the above trade-off between accuracy and complexity, a different approach consists in designing a mesh which is locally adapted to the function g . A way of doing this could be, according to (3), to use bigger triangles (hence larger values of h) where g has a small $W^{2,\infty}$ semi-norm, and smaller ones elsewhere. Intuitively, this should reduce the cardinality of the triangulation, while not increasing much the global interpolation error. A more convenient setting, however, is given by the following local estimate - substantially stronger than (3):

$$\|(I - P_K)g\|_{L^\infty(K)} \leq C|g|_{W^{2,1}(K)} \quad (6)$$

also valid on any shape regular triangle K , and where the scale invariance (*i.e.* the fact that C does not depend on the diameter h of K) corresponds to the fact that the Sobolev embedding of $W^{2,1}(K)$ into $L^\infty(K)$ is critical (since we have $\frac{1}{\infty} = \frac{1}{1} - \frac{2}{d}$ in dimension $d = 2$). According to this estimate, a natural desire is to find a triangulation \mathcal{K}_ε that equilibrates the local semi-norms $|g|_{W^{2,1}(K)}$, in the sense that it satisfies

$$C\varepsilon \leq |g|_{W^{2,1}(K)} \leq C'\varepsilon \quad (7)$$

for any $K \in \mathcal{K}_\varepsilon$. Clearly, the associated interpolation P_ε would satisfy

$$\|(I - P_\varepsilon)g\|_{L^\infty(\Omega)} \leq C\varepsilon,$$

and because summing the left inequalities in (7) yields

$$\#(\mathcal{K}_\varepsilon) \leq C\varepsilon^{-1}|g|_{W^{2,1}(\Omega)},$$

the resulting adaptive approximation $(\mathcal{K}_\varepsilon, P_\varepsilon g)$ would satisfy

$$\|(I - P_\varepsilon)g\|_{L^\infty(\Omega)} \leq C\#\mathcal{K}_\varepsilon^{-1}|g|_{W^{2,1}(\Omega)}. \quad (8)$$

Note that this rate is not identical to (5), since it is achieved for functions which are only in $W^{2,1}(\Omega)$. And in the case where g is in $W^{2,\infty}$ but has highly nonuniform smoothness, the adaptive approach is also justified by the fact that the quantity $|g|_{W^{2,1}(\Omega)}$ is likely to be much smaller than $|g|_{W^{2,\infty}(\Omega)}$.

2.2 Multiscale algorithms

From the above arguments, it now appears that an adaptive strategy is likely to yield better results when interpolating some function of nonuniform smoothness. What we did not mention is an algorithm to design a triangulation \mathcal{K}_ε that fulfills the relations (7), and this might in practice be quite a difficult task. For sake of simplicity, we shall therefore restrict ourselves to a particular class of triangulations that are obtained by recursive splittings of dyadic square cells. The resulting multiscale finite elements should then be seen as a *compromise* between the uniform and the pure adaptive above-described triangulations. As it is usual in compromises, we will have to choose between the two inequalities in (7), and because what we are first interested in is the accuracy of the approximations, we will choose the upper one. Nevertheless, we shall see later on that making such a choice still allows to write some complexity estimates.

We then begin by introducing multiscale quad meshes, from which we shall later on derive conforming triangulations. To this end, consider at any level $\ell \in \mathbb{N}$ the sets

$$\mathcal{Q}_\ell := \{\alpha_{\ell,i,k} \subset \Omega : 0 \leq i, k \leq 2^\ell - 1\}$$

made of the dyadic square cells $\alpha_{\ell,i,k} := (i2^{-\ell}, (i+1)2^{-\ell}) \times (k2^{-\ell}, (k+1)2^{-\ell})$ included in Ω , and denote by $\ell(\alpha)$ the level of a given cell α . Since the partitions \mathcal{Q}_ℓ are nested, we may equip them with a natural quadtree structure by defining the *children* of a given cell α as

$$\mathcal{C}(\alpha) := \{\beta \in \mathcal{Q}_{\ell(\alpha)+1} : \beta \subset \alpha\}$$

and its *parent* as

$$\mathcal{P}(\alpha) := \{\beta \in \mathcal{Q}_{\ell(\alpha)-1} : \beta \supset \alpha\}.$$

A set $\Lambda \subset \cup_{\ell \geq 0} \mathcal{Q}_\ell$ will then be called a *tree* if it satisfies

$$\mathcal{Q}_0 = \{\Omega\} \subset \Lambda \quad \text{and} \quad \alpha \in \Lambda \implies \mathcal{P}(\alpha) \in \Lambda,$$

and if it also satisfies

$$\alpha \in \Lambda \implies \mathcal{C}(\alpha) \subset \Lambda \quad \text{or} \quad \mathcal{C}(\alpha) \cap \Lambda = \emptyset, \quad (9)$$

it will be said *consistent* with respect to a partitioning principle. In fact, the tree structure should only be seen as a convenient setting for performing algorithmic refinements: just as refining a cell in a mesh consists in replacing it by its children, refining it in a tree Λ consists in adding its children to Λ . The corresponding mesh is then expressed as the *leaves* of Λ , that is the set

$$M(\Lambda) := \{\alpha \in \Lambda : \mathcal{C}(\alpha) \cap \Lambda = \emptyset\}, \quad (10)$$

which according to (9), is a partition of Ω .

Being interested in P_1 interpolations, a natural criterion of refinement is, considering (6), that the local $W^{2,1}$ semi-norm of the function g be larger than some prescribed tolerance ε .

Definition 1 (ε -adapted meshes). *A quad mesh M will be said to be ε -adapted to a function g (with respect to the $W^{2,1}$ semi-norm) if it satisfies*

$$\sup_{\alpha \in M} |g|_{W^{2,1}(\alpha)} \leq \varepsilon.$$

This definition can be extended to triangulations.

Algorithm 1 (quad mesh ε -adapted to g).

- Starting from the root cell $\Lambda_0 := \mathcal{Q}_0 = \{\Omega\}$,
- build for $\ell \geq 0$

$$\Lambda_{\ell+1} := \Lambda_\ell \cup \{\beta \in \mathcal{C}(\alpha) : \alpha \in M(\Lambda_\ell), |g|_{W^{2,1}(\alpha)} > \varepsilon\}$$

until $\Lambda_{L+1} = \Lambda_L$,

- and set $\tilde{M}_\varepsilon(g) := M(\Lambda_L)$.

Now in order to establish some stability properties for the transport scheme, we will need the quad meshes to be *graded*, in the sense that any couple (α, β) of their cells satisfy

$$\bar{\alpha} \cap \bar{\beta} \neq \emptyset \implies |\ell(\alpha) - \ell(\beta)| \leq 1. \quad (11)$$

The smallest graded mesh adapted to g can be seen as the smallest graded refinement of $\tilde{M}_\varepsilon(g)$, which is expressed by the following algorithm.

Algorithm 2. (graded mesh ε -adapted to g)

- Starting from $\Lambda_0^G = \Lambda_L$, given by algorithm 1,
- build for $\ell = 0, \dots, L-2$

$$\Lambda_{\ell+1}^G := \Lambda_\ell^G \cup \{\mathcal{C}(\alpha) : \alpha \in \Lambda_\ell^G, \ell(\alpha) = \ell \text{ and}$$

$$\exists \beta \in \Lambda_\ell^G, \text{ s.t. } \ell(\beta) = \ell(\alpha) + 2 \text{ and } \bar{\alpha} \cap \bar{\beta} \neq \emptyset\},$$

- and set $M_\varepsilon(g) := M(\Lambda_{L-1}^G)$.

Note that we would have $\Lambda_L^G = \Lambda_{L-1}^G$ if the former was to be computed. Because $M_\varepsilon(g)$ is a refinement of $\tilde{M}_\varepsilon(g)$, it is obviously ε -adapted to g . But we also have

$$\#(M_\varepsilon(g)) \leq C\#(\tilde{M}_\varepsilon(g))$$

with an absolute constant, see for instance lemma 2.4 in (Dahmen, 1982). Concerning the first algorithm, it is easily seen that it converges (*i.e.* that the level L is finite) as soon as g is in $W^{2,1}(\Omega)$. However, due to the tree structure, it is not possible to write an upper bound for L (hence for $\#(\tilde{M}_\varepsilon(g))$) from the only $W^{2,1}$ semi-norm of g . This can be seen by thinking of an arbitrary ϕ in $W^{2,1}(\Omega)$ and considering the functions $\phi_j(x, v) := \phi(2^j x, 2^j v)$ which somehow concentrate the $W^{2,1}$ semi-norm of ϕ in the small domains $(0, 2^{-j})^2$: as long as $|\phi|_{W^{2,1}(\Omega)} > \varepsilon$, j levels of refinements are needed in $\tilde{M}_\varepsilon(\phi_j)$, while all the semi-norms $|\phi_j|_{W^{2,1}(\Omega)}$, $j \geq 0$, are equal to $|\phi|_{W^{2,1}(\Omega)}$.

Writing down an estimate for the complexity of the ε -adapted meshes is therefore only possible if this hard kind of concentration is prevented. One way of doing so is to ask a bit more smoothness for g : if for instance g is in $W^{2,p}(\Omega)$ with $p > 1$ (but arbitrarily close to 1), then $\#(\tilde{M}_\varepsilon(g))$ is bounded by $|g|_{W^{2,p}} \varepsilon^{-1}$, up to some multiplicative constant. For a proof, and for another way to prevent this concentration phenomenon, see chapters 2 and 3 in (Campos Pinto, 2005).

2.3 Hierarchical P_1 finite elements based on dyadic meshes

To perform P_1 interpolation, we shall now associate to any graded quad mesh M a conforming triangulation $\mathcal{K}(M)$ that is equivalent to M , in a sense that will be soon made precise. First, a non-conforming triangulation $\tilde{\mathcal{K}}(M)$ is obtained by simply splitting each cell α in M in two triangles, with the following rule: if α is an upper left or a lower right child (of its parent cell), it is splitted into its lower left and upper right halves, and the splitting is symmetric in the other two cases. We can observe on figure 1 that unless M is uniform, the resulting triangulation $\tilde{\mathcal{K}}(M)$ is nonconforming. Indeed, when a quad cell α shares an edge with two finer cells β and λ , this gives rise to a big triangle (say α^-) that shares an edge with two smaller triangles (say β^- and λ^+). Now because M is graded, it can be seen that this is the only possible configuration where the triangles are nonconforming, and that a conforming triangulation $\mathcal{K}(M)$ can be obtained by simply merging any such pair (β^-, λ^+) of small triangles. Because every cell in M (resp. every triangle in $\mathcal{K}(M)$) intersects at most two triangles in $\mathcal{K}(M)$ (resp. two cells in M), we simultaneously have

$$\#(\mathcal{K}(M)) \sim \#(M)$$

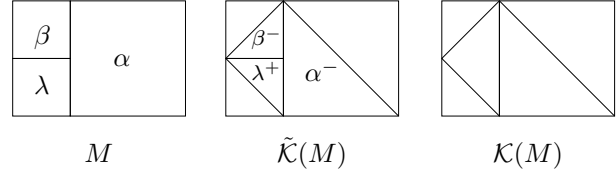


Figure 1. deriving a conforming triangulation from a graded quad mesh.

($A \sim B$ meaning that $CA \leq B \leq C'A$ holds with absolute constants) and

$$\sup_{K \in \mathcal{K}(M)} |g|_{W^{2,1}(K)} \sim \sup_{\alpha \in M} |g|_{W^{2,1}(\alpha)}$$

for any g in $W^{2,1}$, and this is why we say that M and $\mathcal{K}(M)$ are equivalent. In particular, the piecewise affine interpolation operator P_M associated to M via the conforming triangulation $\mathcal{K}(M)$ satisfies

$$\|(I - P_M)g\|_{L^\infty} \lesssim \sup_{\alpha \in M} |g|_{W^{2,1}(\alpha)}$$

so that we have

$$\|(I - P_M)g\|_{L^\infty} \lesssim \varepsilon$$

as long as M is a graded quad mesh ε -adapted to g , which for instance is the case for $M_\varepsilon(g)$.

3 The semi-Lagrangian scheme for the Vlasov equation

The principle of the semi-Lagrangian method is to combine Lagrangian transport and interpolation on a given mesh.

3.1 Non-linear transport along the characteristic curves

Regarding the Vlasov equation (1), the property on which the method is based is the fact that the exact solution f is constant on some particular trajectories, referred to as the *characteristic curves* of the system. Denoted $(X, V)(t) = (X, V)(t; s, x, v)$, these are defined as the unique solution of

$$(X, V)(s) = (x, v) \quad \text{and} \quad \begin{cases} X'(t) = V(t) \\ V'(t) = E(t, X(t)) \end{cases},$$

so that $(X, V)(t; s, x, v)$ should be seen as the position at time t of some phase space “particle” that was standing in position (x, v) at time s , and which speed is given by the non-linear, time depending field $(V, E(t, X))$.

Of course, in order to define these trajectories, we need to know that the solution f does exist and that the electric field E has some smoothness. For what regards the existence of solutions in our case of interest, that is when the initial data f_0 is continuous and compactly supported, we refer to a theorem of (Cooper and Klimas, 1980) (based on a previous result of (Iordanskii, 1964)), according to which there exists a continuous solution f that is also compactly supported.

Remark 1. *In order that the solution remains supported in Ω during the time interval $[0, T = N\Delta t]$, we can assume in the sequel that f_0 is supported in a small ball centered in Ω , and which radius only depends on T (see lemma 2.3 in (Campos Pinto and Mehrenberger, submitted) for details).*

Regarding the smoothness of the solutions, estimates for large (but finite) times are also well known, see for instance (Raviart, 1985) or (Glasse, 1996). Basically, what we need to know is that the Sobolev spaces $W^{m,p}$ are preserved by the equation, but precise smoothness estimates are stated and established in (Campos Pinto and Mehrenberger, submitted). Now from the smoothing effect of the Poisson coupling, as long as the initial condition f_0 is continuous, the electric field E is a Lipschitz function in space and the characteristic curves are well defined. The Vlasov equation can therefore be written in Lagrangian form:

$$\frac{d}{dt} f(t, X(t; 0, x, v), V(t; 0, x, v)) = 0, \quad \forall (x, v) \in \Omega.$$

In particular, a natural way of looking at the solution f in the time interval $[n\Delta t, (n+1)\Delta t]$ is to introduce the backward flow

$$\mathcal{B}(n): (x, v) \rightarrow (X, V)(n\Delta t; (n+1)\Delta t, x, v), \quad (12)$$

so that

$$f((n+1)\Delta t, \cdot, \cdot) = f(n\Delta t, \cdot, \cdot) \circ \mathcal{B}(n).$$

In fact, it follows from the Cauchy-Lipschitz theorem that the flow $\mathcal{B}(n)$ is a C^1 -diffeomorphism from \mathbb{R}^2 to itself, which will be a key ingredient of our adaptive scheme, as it is in the smoothness analysis of the solutions.

3.2 Principle of the semi-Lagrangian scheme

Based on this pointwise transport property, the semi-Lagrangian scheme computes the numerical solution f_{n+1} from f_n in three steps (see figure 2 below for an illustration with underlying graded quad meshes):

1. approach the exact backward flow $\mathcal{B}(n)$ (12) by some computable C^1 -diffeomorphism $\mathcal{B}[f_n]$,

2. define an intermediate solution by transporting f_n along this flow:

$$\mathcal{T}f_n := f_n \circ \mathcal{B}[f_n], \quad (13)$$

3. and construct f_{n+1} by interpolating (by piecewise affine elements) this transported solution on some triangulation \mathcal{K} .

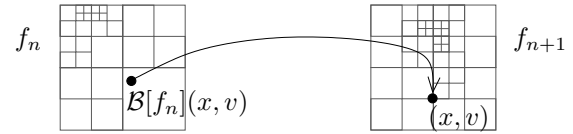


Figure 2. in the semi-Lagrangian method, the numerical density is transported by (i) following approximated characteristic curves and (ii) interpolating on a given mesh.

Remark 2 (computational cost). *Like the numerical flow, the intermediate solution $\mathcal{T}f_n$ is computable everywhere, but only computed on the nodes of \mathcal{K} . The cost of one iteration is then on the order of $C_{\mathcal{B}}\#\mathcal{K}$, where $C_{\mathcal{B}}$ is the cost of computing one position $\mathcal{B}[f_n](x, v)$.*

Remark 3 (high order interpolations). *Because first order interpolations are known to be very diffusive, one may wish to perform higher order interpolations, which is easily doable as soon as we have a conforming triangulation at hand. Nevertheless, due to the fact that they do not diminish the L^∞ norm (as the piecewise affine interpolation do), they lead to a more difficult analysis.*

In our particular case, we should mention that the exact flow is approached by a splitting scheme introduced by (Cheng and Knorr, 1976). Let us describe it for sake of completeness: denoting $E[g]$ the electric field associated to some arbitrary phase space distribution g , i.e. satisfying

$$\partial_x E[g](x) = \int g(x, v) dv, \quad (14)$$

the splitting consists in defining the one-directional flows

$$\mathcal{B}_x^{1/2}(x, v) = (x - v\Delta t / 2, v) \quad (15a)$$

and

$$\mathcal{B}_v[g](x, v) = (x, v - E[g]\Delta t). \quad (15b)$$

The corresponding (partial) transport operators are then

$$\mathcal{T}_x^{1/2}: g \rightarrow g \circ \mathcal{B}_x^{1/2} \quad \text{and} \quad \mathcal{T}_v: g \rightarrow g \circ \mathcal{B}_v[g],$$

and the full transport operator is $\mathcal{T} = \mathcal{T}_x^{1/2}\mathcal{T}_v\mathcal{T}_x^{1/2}$. Following this principle, we obtain a computable approximation (13) where the backward flow $\mathcal{B}[f_n]: (x, v) \rightarrow (\tilde{x}, \tilde{v})$ is given by

$$\begin{cases} \tilde{x} = x - v\Delta t + (\Delta t^2/2)E[\mathcal{T}_x^{1/2}f_n](x - v\Delta t/2) \\ \tilde{v} = v - \Delta t E[\mathcal{T}_x^{1/2}f_n](x - v\Delta t/2). \end{cases}$$

In (Campos Pinto and Mehrenberger, submitted), it is proved that this splitting scheme is (locally) third order accurate in time, as the error resulting from the approximation of the exact characteristics - if ever performed - would be bounded by

$$\|\mathcal{B}(n)(x, v) - \mathcal{B}[f(n\Delta t)](x, v)\|_{\ell^2(\mathbb{R}^2)} \leq C\Delta t^3 \quad (16)$$

uniformly in (x, y) , as long as the initial density f_0 (and thus $f(n\Delta t)$) is in $W^{1,\infty}(\mathbb{R}^2)$.

Remark 4. *In practice, the numerical electric field is thus given by the one-dimensional integration (14) of a numerical approximation to $\mathcal{T}_x^{1/2} f_n$ (see also remark 6).*

3.3 Error analysis in the uniform setting

In (Besse, 2004), Nicolas Besse gave an a priori error estimate for the semi-Lagrangian method in the case where \mathcal{K} is a fixed conforming, shape regular triangulation \mathcal{K}_h , and where the initial data f_0 is in $W^{2,\infty}$. His numerical scheme reads then

$$\mathbb{S}_{h,\Delta t} : f_n \rightarrow f_{n+1} := P_h \mathcal{T} f_n, \quad n = 0, \dots, N-1$$

where h denotes the maximal diameter of the elements in \mathcal{K}_h and P_h the associated P_1 interpolation. Besse's analysis consists in decomposing the error

$$e_{n+1} := \|f((n+1)\Delta t) - f_{n+1}\|_{L^\infty}$$

in three parts, namely $e_{n+1,1}$, $e_{n+1,2}$ and $e_{n+1,3}$, as follows.

The first part reads

$$e_{n+1,1} := \|f((n+1)\Delta t) - \mathcal{T}f(n\Delta t)\|_{L^\infty}$$

and is only due to the approximation of the characteristics, by the numerical transport operator \mathcal{T} . According to the definition of the approximated (13) and exact (12) backward flows, this first error term is bounded by

$$\|f(n\Delta t)\|_{W^{1,\infty}} \|\mathcal{B}(n) - \mathcal{B}[f(n\Delta t)]\|_{L^\infty} \leq C\Delta t^3$$

as long as the initial data f_0 is in $W^{1,\infty}(\mathbb{R}^2)$, according to section 3.1 and the accuracy (16) of the numerical flow. Note that the constant C now depends on the initial data, and on the maximal time $T = N\Delta t$.

The second part reads

$$e_{n+1,2} := \|(I - P_h)\mathcal{T}f(n\Delta t)\|_{L^\infty}$$

and corresponds to the interpolation error on K_h . Using (4), it is estimated by

$$e_{n+1,2} \leq Ch^2 |\mathcal{T}f(n\Delta t)|_{W^{2,\infty}}.$$

Since f_0 is assumed to be in $W^{2,\infty}$, so is $f(n\Delta t)$ according to section 3.1. Now the question is to know whether the transport operator is stable in this semi-norm, which is shown in (Besse, 2004). We therefore have

$$e_{n+1,2} \leq Ch^2 |f(n\Delta t)|_{W^{2,\infty}} \leq Ch^2.$$

Finally, the remainder part reads

$$e_{n+1,3} := \|P_h(\mathcal{T}f(n\Delta t) - \mathcal{T}f_n)\|_{L^\infty}$$

and is easily bounded by $\|\mathcal{T}f(n\Delta t) - \mathcal{T}f_n\|_{L^\infty}$, which can be seen as the propagation of the numerical error at the previous time step. Note that if the operator \mathcal{T} was linear, $\mathcal{T}f(n\Delta t) - \mathcal{T}f_n$ would simply read $\mathcal{T}(f(n\Delta t) - f_n)$ and because a pointwise transport operator can only diminish the amplitude of a function, we would clearly have $\|\mathcal{T}f(n\Delta t) - \mathcal{T}f_n\|_{L^\infty} \leq e_n$. In fact, we know that the operator \mathcal{T} is non linear, but it is stable enough to satisfy

$$\|\mathcal{T}f(n\Delta t) - \mathcal{T}f_n\|_{L^\infty} \leq (1 + C\Delta t)e_n.$$

Gathering the above estimates, Besse obtains that

$$e_{n+1} \leq (1 + C\Delta t)e_n + C(\Delta t^3 + h^2),$$

which yields, according to the Gronwall lemma,

$$e_n \leq C(\Delta t^2 + h^2/\Delta t)$$

where C only depends on the initial data f_0 and on the final time T . Balancing $\Delta t^3 \sim h^2$ in this estimate, we eventually find the following convergence rate:

$$\|f(n\Delta t) - f_n\|_{L^\infty} \leq Ch^{4/3} \leq C\#(\mathcal{K}_h)^{-2/3}. \quad (17)$$

4 The adaptive semi-Lagrangian scheme

Following the three steps described in section 3.2, the principle of the adaptive semi-Lagrangian (AdSL) scheme simply consists in interpolating the transported $\mathcal{T}f_n$ on a mesh which is adapted to it rather than fixed.

4.1 Predicting the adaptive meshes

In order to simplify the ‘‘mesh design’’ as much as possible, we now restrict ourselves to triangulations that can be derived from quad meshes as described in section 2.3. We shall then associate to any numerical solution f_n a quad mesh M_n . Our problem now consists in building at each time step n , and given the couple (M_n, f_n) , a *new mesh* M_{n+1} that is well adapted to $\mathcal{T}f_n$. In the light of section 2, one may want to use the ε -adapted mesh $M_\varepsilon(\mathcal{T}f_n)$ given by algorithms 1 and 2. Unfortunately,

and as remark 2 stressed out, $\mathcal{T}f_n$ is not to be computed everywhere, so that algorithm 1 cannot be applied on it. One has then to find another way to guess what the adaptive M_{n+1} should be.

As a starter, here is the prediction strategy that was proposed to us by Eric Sonnendrücker during the CEM-RACS 2003 summer school, and implemented in the YODA code described in (Campos Pinto and Mehrenberger, 2005). For an accurate prediction strategy in the context of scalar conservation laws and finite volume schemes, see also (Cohen *et al.*, 2003). Remember that we have at hand a graded quad mesh M_n and assume that we know how to compute from the numerical solution f_n not only a backward flow $(x, v) \rightarrow \mathcal{B}[f_n](x, v)$ but also *forward* one $\mathcal{F}[f_n] := \mathcal{B}[f_n]^{-1}$. For each cell α in M_n , we then advect its center $c(\alpha)$ along $\mathcal{F}[f_n]$, and denote by $\mathbb{F}[f_n](\alpha)$ the cell of level $\ell(\alpha)$ that contains $\mathcal{F}[f_n](c(\alpha))$ (up to choosing a cell if this point falls on an edge). A reasonable mesh for interpolating $\mathcal{T}f_n$ is given by the smallest one containing the set $\{\mathbb{F}[f_n](\alpha) : \alpha \in M_n\}$. Clearly, its building cost is about $C_{\mathcal{F}}\#(M_n)$ (where $C_{\mathcal{F}}$ is an upper bound for the computational cost of one position $\mathcal{F}[f_n](x, v)$, see remark 2) which is on the same order than the computational cost of a numerical solution, provided $C_{\mathcal{F}}$ and $C_{\mathcal{B}}$ are on the same order. The major drawback is that there is no guarantee that the corresponding interpolation achieves a given accuracy.

4.2 Optimal transport of multiscale meshes

Based on both the desire of constructing ε -adapted meshes and the will to remain within a building cost on the order of $C_{\mathcal{B}}\#(M_n)$, the algorithm we now describe, and that was first introduced in (Campos Pinto and Mehrenberger, submitted), maps the couple (M_n, f_n) to a new graded mesh, denoted $\mathbb{T}[f_n](M_n)$, that has the following properties:

- its building cost is on the order of $C_{\mathcal{B}}\#(M_n)$, where $C_{\mathcal{B}}$ is an upper bound for the computational cost of one position $\mathcal{B}[f_n](x, v) \in \Omega$;
- its cardinality is on the same order than M_n , *i.e.* $\#(\mathbb{T}[f_n](M_n)) \leq C\#(M_n)$, and its maximal level is not larger than the one of M_n ;
- provided that M_n is ε -adapted to f_n , $\mathbb{T}[f_n](M)$ is $C\varepsilon$ -adapted to $\mathcal{T}f_n := f_n \circ \mathcal{B}[f_n]$ with an absolute constant C .

Remark 5 (optimality of $\mathbb{T}[f_n]$). *The operator $\mathbb{T}[f_n]$ is said to be optimal because it satisfies*

$$\|(I - P_{\mathbb{T}[f_n](M_n)})\mathcal{T}f_n\|_{L^\infty} \leq C\varepsilon \leq C'\#(\mathbb{T}[f_n](M_n))^{-1}$$

provided that M_n is ε -adapted to f_n and that it satisfies $\varepsilon \leq C\#(M_n)^{-1}$.

Before writing down the algorithm, let us describe it in a few words. Basically, the main idea is to perform recursive splitting like in algorithm 1, but with a different refinement criterion. As mentioned above, the cells that we wish to subdivide are those on which the $W^{2,1}$ semi-norm of $\mathcal{T}f_n$ is larger than ε , but we do not have this function at hand. Nevertheless, since $\mathcal{T}f_n$ reads $f_n \circ \mathcal{B}[f_n]$, we may ask whether

$$|\mathcal{T}f_n|_{W^{2,1}(\alpha)} \leq C|f_n|_{W^{2,1}(\mathcal{B}[f_n](\alpha))}, \quad (18)$$

or in other words, if the numerical transport operator \mathcal{T} is *stable* with respect to the $W^{2,1}$ semi-norm. For the moment, let us assume that the answer to that question is yes (and as we shall see in section 4.5, the actual answer is: strictly speaking, no, but essentially, yes). If (18) holds, we can decompose the right hand side over the cells of M_n that intersect $\mathcal{B}[f_n](\alpha)$, *i.e.*

$$|\mathcal{T}f_n|_{W^{2,1}(\alpha)} \leq C \sum_{\beta \in \mathcal{I}(M_n, f_n, \alpha)} |f_n|_{W^{2,1}(\beta)}$$

where the set $\mathcal{I}(M_n, f_n, \alpha)$ is precisely defined as

$$\mathcal{I}(M_n, f_n, \alpha) := \{\beta \in M_n, \beta \cap \mathcal{B}[f_n](\alpha) \neq \emptyset\}.$$

Now if M_n is ε -adapted to f_n , as it ought to be, every quantity $|f_n|_{W^{2,1}(\beta)}$ appearing in the above sum is precisely bounded by ε . One question then remains, namely: is it possible to bound the cardinality of the set $\mathcal{I}(M_n, f_n, \alpha)$ by a constant? The good news is that this is possible. More precisely, if the center of α is backward advected into a cell of M_n which level is not larger than that one of α , that is to say, if $\mathcal{B}[f_n](c(\alpha))$ belongs to some $\beta \in M_n$ such that $\ell(\beta) \leq \ell(\alpha)$, then we have

$$\mathcal{I}(M_n, f_n, \alpha) \leq C \quad (19)$$

with an absolute constant, provided Δt is not larger than a constant that depends on f_0 and T . This inequality is deeply connected to the graded property (11) of the mesh M_n , and to the fact that the numerical flow $\mathcal{B}[f_n]$ is a C^1 -diffeomorphism. And again, we refer to (Campos Pinto and Mehrenberger, submitted) or, for french readers, to chapters 5 and 6 in (Campos Pinto, 2005) for a detailed proof. Denoting

$$\ell^*(M_n, f_n, \alpha) := \max\{\ell(\beta) : \beta \in M_n, \mathcal{B}[f_n](c_\alpha) \in \bar{\beta}\}$$

the level of this cell β (or the maximal level over such cells in the case where $\mathcal{B}[f_n](c_\alpha)$ lies on an edge), what we are able to show is essentially that

$$|\mathcal{T}f_n|_{W^{2,1}(\alpha)} \leq C\varepsilon \quad \text{as long as} \quad \ell^*(M_n, f_n, \alpha) \leq \ell(\alpha),$$

up to a few modifications mentioned in section 4.5 below. In other words, we have just found a *computable refinement criterion* for designing a mesh that is well adapted to $\mathcal{T}f_n$. Our algorithm reads then as follows.

Algorithm 3 (optimal mesh transport).

- Starting from $\Lambda_0 = \mathcal{Q}_0 = \{\Omega\}$,
- build for $\ell \geq 0$

$$\Lambda_{\ell+1} := \Lambda_\ell \cup \{\beta \in \mathcal{C}(\alpha) : \\ \alpha \in M(\Lambda_\ell), \ell^*(M_n, f_n, \alpha) > \ell(\alpha)\}$$

until $\Lambda_{L+1} = \Lambda_L$,

- and let $\mathbb{T}[f_n]M_n$ be the smallest graded refinement of Λ_L (see algorithm 2).

4.3 Essential form of the AdSL scheme

We now have at hand a practical algorithm to build at a reasonable cost a graded mesh $\mathbb{T}[f_n]M_n$ that is $C\varepsilon$ -adapted to $\mathcal{T}f_n$ as long as M_n is ε -adapted to f_n , so that the interpolation error $\|(I - P_{\mathbb{T}[f_n]M_n})\mathcal{T}f_n\|_{L^\infty}$ is on the order of ε . Note that having a multiplicative constant in front of ε is not a major problem for a few iterations, but that using the only algorithm 3 in the scheme would construct meshes $\mathbb{T}[f_n]M_n$ that are only $C^n\varepsilon$ -adapted to the transported $\mathcal{T}f_n$, which is very poor as Δt goes to zero. To prevent this accumulation, we therefore need to apply algorithm 2 as a *correction step* to guarantee that M_n is not only $C\varepsilon$ but actually ε -adapted to f_n .

Essentially, the AdSL scheme reads then as follows:

$$\mathbb{S}_{\varepsilon, \Delta t} : f_n \rightarrow f_{n+1} := P_{M_{n+1}} P_{\tilde{M}_{n+1}} \mathcal{T}f_n \quad (20)$$

for $n = 0, \dots, N-1$. Here \tilde{M}_{n+1} and M_{n+1} are respectively given by algorithms 3 and 2 (up to the modifications described in section 4.5 below), following

$$\tilde{M}_{n+1} := \mathbb{T}[f_n]M_n \quad (21a)$$

$$M_{n+1} := M_\varepsilon(P_{\tilde{M}_{n+1}} \mathcal{T}f_n). \quad (21b)$$

Remark 6. In (Campos Pinto and Mehrenberger, submitted), the form of the scheme is a bit more complicated, which is mainly due to the splitting technique (15) used to approach the characteristic curves. Another complication comes from the use of a truncation operator to guarantee that the numerical solutions - like the exact ones - remain compactly supported in spite of the diffusion induced by the interpolations.

4.4 Error and complexity estimates

Here is our main result: up to the modifications mentioned in remark 6 above and in section 4.5 below, the adaptive solutions f_n computed by the AdSL scheme (20)-(21) satisfy

$$\|f(n\Delta t) - f_n\|_{L^\infty} \leq C(\Delta t^2 + \varepsilon/\Delta t) \quad (22)$$

which a constant C depending only on the initial solution f_0 and on the final time $T = N\Delta t$ of the simulation, provided:

- the initial solution is in $W^{1,\infty}(\Omega) \cap W^{2,1}(\Omega)$ and has a compact support (see remark 1);
- the parameter Δt is not larger than a constant which only depends on the L^∞ norm of f_0 and on the final time T (this requirement is essential for establishing the bound (19));
- the parameter ε is not larger than Δt^2 (which might only be a technical requirement, but not very restrictive, since balancing the terms in the above estimate leads to choosing $\varepsilon \sim \Delta t^3$ as both parameters tend to 0).

We refer to (Campos Pinto and Mehrenberger, submitted) or to chapter 6 in (Campos Pinto, 2005) for a complete proof of this result.

In addition to this error estimate, we know that the cardinality of the predicted meshes is on the same order than that one of the corrected ones, *i.e.*

$$\#(\tilde{M}_{n+1}) \leq C\#(M_n).$$

Now according to section 2.2, we know that up to preventing the concentration-of-curvature phenomenon, the complexity of these corrected meshes M_n should be monitored by ε and by the $W^{2,1}$ semi-norm (or better, by the total curvature introduced in section 4.5 below) of the numerical solutions, but at the present stage we do not know how to rigorously estimate the growth of this quantity as time evolves. Our conjecture is that the cardinality of the corrected meshes M_n is bounded by $C\varepsilon^{-1}$. Balancing $\varepsilon \sim \Delta t^3$ in estimate (22) would then yield

$$\|f(n\Delta t) - f_n\|_{L^\infty} \leq C\varepsilon^{2/3} \\ \leq C[\max(\#(M_n), \#(\tilde{M}_n))]^{-2/3},$$

provided the initial data is in $W^{1,\infty} \cap W^{2,1}$. Note that our “adaptive” estimates require less regularity than the “uniform” one (17) (that only holds provided f_0 is in $W^{2,\infty}$), which is a classical way to express the superiority of the adaptive methods over the uniform ones.

4.5 A few precisions

Now let us mention why the stability (18) does not exactly hold in this way. The first reason is that since f_n is piecewise affine, its second derivatives are Dirac measures supported on the edges of a given triangulation, and the $W^{2,1}$ semi-norm of f_n (as well as that of $\mathcal{T}f_n$) is not even defined. In a way, this difficulty comes from the fact that, unlike Besse, we do not consider the “dynamic interpolation error” $(I - P)\mathcal{T}$ on the exact $f(n\Delta t)$ but on the numerical f_n as it actually happens in the scheme. In other words, instead of bounding the error $e_{n+1} = \|f((n+1)\Delta t) - f_{n+1}\|_{L^\infty}$ by $e_{n+1,1} + A$ with

$$A = \|(I - P)\mathcal{T}f(n\Delta t)\|_{L^\infty} + \|P(\mathcal{T}f(n\Delta t) - \mathcal{T}f_n)\|_{L^\infty}$$

as in section 3.3, we bound it using

$$A = \|(I - P)\mathcal{T}f_n\|_{L^\infty} + \|\mathcal{T}f(n\Delta t) - \mathcal{T}f_n\|_{L^\infty}.$$

Had we chosen the first approach, there would be no problem about the smoothness of $\mathcal{T}f(n\Delta t)$, but on the other side, because algorithm 1 cannot be applied on the unknown $P_{\tilde{M}_{n+1}}\mathcal{T}f(n\Delta t)$, we would not have been able to perform the correction step (21b) and guarantee that the adaptive mesh M_n is always ε -adapted to the solution.

Now this difficulty can be overcome by introducing the *total curvature* of a function, defined as the total mass of its second derivatives seen as Radon measures. By analogy with $BV(\Omega)$, we denote by $BC(\Omega)$ the space consisting of the functions g in $L^1(\Omega)$ which second derivatives are Radon measures of finite total mass, and write

$$|g|_{BC(\Omega)} := |\partial_{xx}^2 g|(\Omega) + |\partial_{xy}^2 g|(\Omega) + |\partial_{yy}^2 g|(\Omega)$$

the corresponding semi-norm. It is then possible to show that the functions in $BC(\Omega)$ are continuous, and that the interpolation error estimate (6) still holds when using the BC semi-norm instead of the $W^{2,1}$ one, see chapter 3 in (Campos Pinto, 2005) or (Campos Pinto, in preparation).

The second reason is that when analyzing the semi-norm $|\mathcal{T}f_n|_{BC(\alpha)}$, we find out that a second quantity comes into play, namely the $W^{1,\infty}$ semi-norm of f_n . Now up to introducing the functional

$$\nu(f_n, \alpha) := |f_n|_{BC(\bar{\alpha})} + \Delta t \text{vol}(\alpha) |f_n|_{W^{1,\infty}(\alpha)}$$

and establishing $W^{2,\infty}$ estimates on the electric field involved in the numerical flow $\mathcal{B}[f_n]$, it is possible to prove the following stability (again, see (Campos Pinto and Mehrenberger, submitted) or (Campos Pinto, 2005) for the proofs)

$$\nu(\mathcal{T}f_n, \alpha) \leq C \sum_{\beta \in \mathcal{I}(M_n, f_n, \alpha)} \nu(f_n, \beta),$$

so that all the estimates stated in the previous sections actually hold, at least when the error indicator $\nu(\cdot, \alpha)$ is used in place of the semi-norm $|\cdot|_{W^{2,1}(\alpha)}$.

Acknowledgments

Once again, I would like to thank Professors Albert Cohen and Eric Sonnendrücker for the competent advise that headed Michel and myself to the present results.

References

- Besse N. (2004): *Convergence of a semi-Lagrangian scheme for the one-dimensional Vlasov-Poisson system*. — SIAM J. Numer. Anal., Vol. 42, No. 1, pp. 350–382.
- Besse N., Filbet F., Gutnic M., Paun I. and Sonnendrücker E. (2001): *An adaptive numerical method for the Vlasov equation based on a multiresolution analysis*. — in Numerical Mathematics and Advanced Applications ENUMATH 2001 (F. Brezzi, A. Buffa, S. Escorsaro and A. Murli Ed.), Springer, pp. 437–446.
- Campos Pinto M. (2005): *Développement et analyse de schémas adaptatifs pour les équations de transport*. — PhD Thesis (in french), Université Pierre et Marie Curie, Paris.
- Campos Pinto M. (in preparation): *P_1 interpolation in the plane and functions of bounded total curvature*.
- Campos Pinto M. and Mehrenberger M. (2005): *Adaptive numerical resolution of the Vlasov equation*. — Numerical methods for hyperbolic and kinetic problems, CEMRACS 2003/IRMA Lectures in Mathematics and Theoretical Physics.
- Campos Pinto M. and Mehrenberger M. (submitted): *Convergence of an adaptive scheme for the one-dimensional Vlasov-Poisson system*. — Submitted to SIAM Journal on Numerical Analysis.
- Cheng C.Z. and Knorr G. (1976): *The integration of the Vlasov equation in configuration space*. — J. Comput. Phys., Vol. 22, pp. 330–351.
- Cohen A., Kaber S.M., Müller S. and Postel M. (2003): *Fully adaptive multiresolution finite volume schemes for conservation laws*. — Math. Comp., Vol. 72, No. 241, pp. 183–225.
- Cooper J. and Klimas A. (1980): *Boundary value problems for the Vlasov-Maxwell equation in one dimension*. — J. Math. Anal. Appl., Vol. 75, No. 2, pp. 306–329.

Dahmen, W. (1982): *Adaptive approximation by multivariate smooth splines*. — J. Approx. Theory, Vol. 36, No. 2, pp. 119–140.

Dahmen W., Gottschlich-Müller B. and Müller S. (2001): *Multiresolution schemes for conservation laws*. — Numer. Math., Vol. 88, No. 3, pp. 399–443.

DeVore, R. (1998): *Nonlinear approximation*. — in Acta Numerica, Vol. 7, Cambridge Univ. Press, pp. 51–150.

Glassey, R.T.(1996): *The Cauchy problem in kinetic theory*. — Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA.

Gutnic M., Haefele M., Paun I. and Sonnendrücker E. (2004): *Vlasov simulations on an adaptive phase-space grid*. — Comput. Phys. Comm., Vol. 164, pp. 214–219.

Iordanskii S. V. (1964): *The Cauchy problem for the Kinetic Equation of Plasma*. — Amer. Math. Soc. Transl. Ser. 2, Vol. 35, pp. 351–363.

Raviart P.-A. (1985): *An analysis of particle methods*. — in Numerical methods in fluid dynamics (Como, 1983), Lecture Notes in Math., Vol. 1127, Springer, Berlin, pp. 243–324.

Roussel O., Schneider K., Tsigulin, A. and Bockhorn H. (2003): *A conservative fully adaptive multiresolution algorithm for parabolic PDEs*. — J. Comput. Phys., Vol. 188, No. 2, pp 493–523.

Sonnendrücker E., Filbet F., Friedman A., Oudet E. and Vay J.L. (2004): *Vlasov simulation of beams with a moving grid*. — Comput. Phys. Comm., Vol. 164, pp. 390–395.

Sonnendrücker E., Roche J., Bertrand P. and Ghizzo A. (1999): *The semi-Lagrangian method for the numerical resolution of the Vlasov equation*. — J. Comput. Phys., Vol. 149, No. 2, pp 201–220.

Yserentant H. (1992): *Hierarchical bases*. — in ICIAM 91 (Washington, DC, 1991), SIAM, Philadelphia, PA, pp. 256–276.