Bulletin of the Institute of Mathematics
Academia Sinica (New Series)
Vol. **2** (2007), No. 2, pp. 533-567

TIME IMPLICIT SCHEMES AND FAST APPROXIMATIONS OF THE FOKKER-PLANCK-LANDAU EQUATION

BY

MOHAMMED LEMOU AND LUC MIEUSSENS

Abstract

In this paper, we are concerned with numerical approximations of the Fokker-Planck-Landau equation which is a kinetic model used to describe the evolution of charged particles in a plasma. In this model, the particle interactions (or collisions) are taken into account by a nonlocal and nonlinear diffusion operator acting on the velocity dependence of the particle distribution function. In a first part of this work, we investigate different strategies to perform efficient time implicit discretisations, while, in the second part, we review various numerical approximations of the collision operator. Both the time discretisation and the approximations of the collision operator are shown to satisfy some important physical properties of conservation and entropy, and to reach the right steady states. Furthermore, various accelerations techniques are used to construct such approximations which would make possible their use in a more realistic setting (inhomogeneous cases). In particular, we combine two new strategies to rapidly and efficiently solve the FPL equation: the first one concerns the time discretisation using time implicit schemes with Krylov solvers, and the second one uses the approximation of the collision operator using the wavelet approximation theory.

Received Received December 10, 2004 and in revised form May 19, 2005.

AMS Subject Classification: 82C40, 82D10, 82C80, 65M06, 65Y20, 65F10.

Key words and phrases: Kinetic equations, Fokker-Planck-Landau equation, implicit schemes, conservative schemes, Krylov methods, wavelets.

1. Introduction

The Fokker-Planck-Landau (FPL) is a kinetic collisional model used to describe a system of particles in plasma physics (see [20] for instance). The particles are described through a distribution function $f(t,x,v)$ depending on time t, particle position $x \in \mathbb{R}^d$, and their velocity $v \in \mathbb{R}^d$ ($d = 2, 3$). In this paper we are concerned with the homogeneous case where $f(t,x,v)$ does not depend on x. The model writes in the so-called Landau form

$$\partial_t f(t,v) = Q(f)(v) = \nabla \cdot \int_{\mathbb{R}^d} \Phi(v-v_*) \left(f(v_*) \nabla f(v) - f(v) \nabla f(v_*) \right) dv_*, \quad (1)$$

where $\Phi(w)$ is the following $d \times d$ matrix

$$\Phi(w) = C|w|^{\gamma+2} S(w) = C|w|^{\gamma+2} \left(I_d - \frac{w \otimes w}{|w|^2} \right).$$

In this expression, C is a positive constant and γ is a parameter leading to the standard classification in hard potentials ($\gamma > 0$), Maxwellian potential ($\gamma = 0$) and soft potentials ($\gamma < 0$). This last case includes the most physically interesting case, the Coulombian case ($\gamma = -3$). The $d \times d$ matrix $S(w)$ is simply the orthogonal projection onto the plane orthogonal to w. For all $w \neq 0$, $\Phi(w)$ is a positive matrix whose null-space is

$$\operatorname{Ker} \Phi(w) = \mathbb{R} w.$$

Throughout this paper, when no confusion is possible, the values of any function f under the integral signs will be denoted by f for $f(v)$ and by f_* for $f(v_*)$. Beside, it is also useful to write the FPL collision operator in the so called "Log form":

$$Q(f)(v) = \nabla \cdot \int_{\mathbb{R}^d} \Phi(v - v_*) f f_* \left(\nabla \log f - \nabla \log f_* \right) dv_*. \quad (2)$$

This collision operator $Q(f)$ satisfies the following weak formulation

$$\int_{\mathbb{R}^d} Q(f) \phi dv = -\frac{1}{2} \int_{\mathbb{R}^d} \int_{\mathbb{R}^d} \Phi(v-v_*) f f_* \left(\nabla \log f - \nabla \log f_* \right) \cdot \left(\nabla \phi - \nabla \phi_* \right) dv dv_*, \quad (3)$$

for all distribution function f and all test function ϕ. From this formulation, one can immediately derive the following conservation and entropy properties

(i) the only functions $\phi(v)$ such that

$$\int_{\mathbb{R}^d} Q(f)\phi \, dv = 0, \qquad \text{for all } f,$$

are linear combinations of $1, v, |v|^2$. In particular, total mass, momentum, and energy are conserved.

(ii) the entropy dissipation inequality

$$\int_{\mathbb{R}^d} Q(f)\log(f) \, dv \leq 0, \qquad \text{for all } f > 0.$$

This also gives the well known H-theorem, saying that the functional $H(f) = \int_{\mathbb{R}^d} f\log(f)\, dv$ is a time non-increasing function. Furthermore, this inequality becomes an equality if and only if f is a Maxwellian

$$f_{eq}(v) = \frac{\rho}{(2\pi T)^{\frac{d}{2}}} \exp(-\frac{|v-u|^2}{2T}), \tag{4}$$

where ρ, u and T are velocity independent parameters. This is formally equivalent to say that f is an equilibrium function, that is $Q(f) = 0$.

Numerical experiments usually show that the exact preservation of properties (i) and (ii) provides efficient and stable numerical schemes. In a recent past, numerous works have been concerned with constructing discretizations of the collision operator that obey the above physical properties of conservation and entropy. The first scheme in this direction was established in [9] and is based on finite difference schemes on a regular velocity grid. Unfortunately, it turned out to be very expensive in terms of CPU time. Indeed, the cost of one evaluation of the collision operator is of the order of $O(N^2)$, where N is the total number of the velocity grid points. Later, various fast algorithms have been constructed to reduce this cost (see [5, 7, 17]) without affecting the conservation and entropy properties. On the other hand, non conservative schemes have to be sufficiently accurate to correctly describe these physical laws. High accuracy could of course be achieved by refining the velocity grid but this becomes very computationally demanding. Note that this is not only due to the complexity of the collision operator but also to its diffusive character which constrains the time step to be excessively small in general. Then, an alternative method using spectral schemes as in [21, 22] has been proposed. This method is well adapted to the convolution structure of the collision operator and yields a $O(N \log N)$ algorithm. However, these methods are not exactly conservative nor entropic.

A part of this paper is then devoted to a review of a method proposed in [1, 2] which combines the advantages of the finite difference schemes (conservation and entropy) and of the spectral method (accuracy) in the isotropic case. The proposed velocity discretization of the FPL operator uses the wavelet approximation theory and is based on the so called multiwavelet method [3]. This leads to conservative, entropic and accurate schemes. Furthermore, the evaluation of the collision operator has a cost of the order of $\mathcal{O}(N)$ only, N being the dimension of the approximation space.

In all these works, the time discretization problem has not been completely solved. Indeed the used schemes are explicit in time, as for instance the usual Euler explicit scheme

$$f^{n+1} = f^n + \Delta t Q(f^n) \tag{5}$$

or higher order versions (explicit Runge-Kutta methods). It is known that such schemes induce a strong parabolic CFL condition of the form $\Delta t \leq C\Delta v^2$, Δt and Δv being the time and velocity steps. This condition is due to the diffusive nature of the FPL operator and has been rigorously established in [5] for the isotropic case (that is where the distribution function only depends on the modulus of the velocity). Therefore, to reach a given simulation time, a large number of iterations n_{it} is required. For instance, in the isotropic case, we have $n_{it} = N^2$, N being the dimension of the approximation space. In that case, even with fast evaluations of the collision operator in $O(N)$ or $O(N \log N)$ (as proposed in the previous works), these explicit schemes require a total simulation cost of the order of $n_{it}O(N) = O(N^3)$ or $O(N^3 \log N)$. However, according to some recent works, it is possible to use explicit schemes with slightly larger time steps. For instance in [12] a high order explicit scheme with a large stability interval has been used. But the gain obtained with this method remains relatively small: the time step can be taken 5 at 10 times as large as that of the Euler method (5), and the total CPU time is divided by a factor 4 only.

Consequently, it is attractive to use time implicit schemes, since it is known that they can be free of restrictive time step conditions. This consists in replacing $Q(f^n)$ in (5) by an approximation that also depends on f^{n+1}. Then the problem is how to construct such implicit schemes in order to satisfy the following two requirements

- the properties (i) and (ii) must be preserved;

- the total computational cost must be smaller than that of the explicit scheme with almost the same accuracy.

Many works in plasma physics area use implicit schemes to solve the FPL equation (see [14, 15] for instance and the references therein). However, the problem of exact preservation of (i) and (ii) is generally not addressed. Moreover, the total complexity of these algorithms is not optimal. We note that, beyond the fact that implicit schemes usually induce an additional computational cost (non-sparse matrix inversion), they may affect also the properties of conservation and entropy. In [10] for instance, Epperlein has proposed an implicit scheme which is conservative, whereas its total numerical complexity is comparable to the usual explicit resolution. This has been clearly shown in [6]. A more recent work by Chacón, Barnes, Knoll and Miley [8] uses a fast linear solver to reduce the cost of their implicit scheme. However, their approach does not exactly respect the conservation and entropy properties. In fact, whereas they claim that their scheme preserves the energy, they also point out that the solvers and the velocity boundary conditions that they used in practice affect the conservation properties.

In a first part of this paper, we develop a strategy leading to exactly conservative implicit schemes with a reduced computational cost, reviewing the results of [18, 19]. One of these schemes also satisfies the entropy property. Iterative Krylov solvers are used to efficiently solve the linear systems generated by the implicit schemes. This strategy is also proved to be conservative, even if the linear systems are only solved approximately, and an important gain in terms of computational cost is obtained. All the schemes developed in this work concern both the 2 and 3-dimensional cases as well as the isotropic FPL equation.

We point out that our method can apply for both the Landau form (1) and the "Log" form (2) of the FPL equation. Hence it can be used with various potentials, and easily be extended to quantum and relatvistic cases. This does not seem to be possible if one uses the so-called Rosenbluth form of the classical FPL equation, as done in [8]. Note also that our schemes can easily be used in the resolution of inhomogeneous problems via standard splitting techniques.

The outline of the paper is as follows. In the next section we focus on the time discretization only, and present the different implicit schemes and summarize the results in [18, 19]. In Section 3, the velocity variable is also discretized using two different strategies: the first one is based on usual

finite difference schemes, and the second uses the wavelet approximation theory. This leads to completely discretized implicit schemes. All these schemes require the resolution of large and non-sparse linear systems. This is addressed in Section 4 where fast linear solvers are proposed. In Section 5, we discuss the numerical complexity of our algorithms, while in Section 6 we present various numerical tests in the isotropic case. The validation of the present strategy for the 3-dimensional case needs further investigations to solve the linear systems. Therefore the numerical tests for the 3-dimensional geometries are defered to a forthcoming paper.

2. Time Implicit Schemes

In this section, we focus on the time discretization only. Suitable discretizations in the velocity variable will be developed in Section 3. Below, we present different strategies to construct linear time implicit schemes that have properties of conservation and entropy.

2.1. Contracted implicit scheme

We first note that the FPL operator (1) can be rewritten in the following diffusive form

$$Q(f) = \nabla \cdot (D(f)\nabla f + F(f)f)\,, \tag{6}$$

where $D(f) = \int_{\mathbb{R}^d} \Phi(v - v_*) f(v_*)\, dv_*$ and $F(f) = \int_{\mathbb{R}^d} \Phi(v - v_*)\nabla f(v_*)\, dv_*$. This shows in particular that the CFL condition resulting from the use of time explicit schemes is due to the diffusive term $\nabla\cdot(D(f)\nabla f)$ in (6). Therefore the first idea is to make ∇f implicit in this last expression. On the other hand, the conservation and entropy properties are a consequence of the symmetry property (between v and v_*) of the collision operator. To preserve this symmetry, the friction coefficient $F(f)$ has to be implicit too. This leads to the following contracted implicit scheme

$$\frac{f^{n+1} - f^n}{\Delta t} \;=\; q^c(f^n, f^{n+1}), \qquad \text{with} \tag{7}$$

$$q^c(f, g) \;=\; \nabla \cdot \int_{\mathbb{R}^d} \Phi(v - v_*)\,(f_*\nabla g - f\nabla g_*)\; dv_*. \tag{8}$$

Note that this operator q^c is not the linearization of Q around f. The linearized operator around f is in fact

$$\begin{aligned} q(f,g) &= \nabla \cdot \int_{\mathbb{R}^d} \Phi(v-v_*)\,(f_*\nabla g - f\nabla g_* + g_*\nabla f - g\nabla f_*)\,dv_* \\ &= q^c(f,g) + q^c(g,f). \end{aligned} \tag{9}$$

Therefore, the contracted scheme (7) is not the usual linearized implicit scheme as used by Epperlein in [10]. The operator q^c is contracted in the sense that we only keep the terms that are necessary to ensure the symmetry between v and v_* in the linearized operator q.

Proposition 2.1.

(i) *The operator q^c satisfies the following weak formulation:*

$$\int_{\mathbb{R}^d} q^c(f,g)\phi dv = -\frac{1}{2}\int_{\mathbb{R}^d}\int_{\mathbb{R}^d} \Phi(v-v_*)\,(f_*\nabla g - f\nabla g_*)\cdot(\nabla\phi - \nabla\phi_*)\,dvdv_*, \tag{10}$$

for any test function ϕ.

(ii) *The contracted scheme given by* (7-8) is conservative:

$$\forall n, \quad \int_{\mathbb{R}^d} (1, v, \tfrac{1}{2}|v|^2)^T f^n\,dv = \int_{\mathbb{R}^d} (1, v, \tfrac{1}{2}|v|^2)^T f^0\,dv.$$

(iii) *The contracted scheme is first order in time.*

For the proof, we refer to [19].

2.2. A θ-scheme

The present approach first consists in a time integration of equation (1) using the standard θ-scheme:

$$\frac{f^{n+1} - f^n}{\Delta t} = (1-\theta)Q(f^n) + \theta Q(f^{n+1}). \tag{11}$$

Then we linearize $Q(f^{n+1})$ around f^n:

$$Q(f^{n+1}) = Q(f^n) + DQ(f^n)(f^{n+1} - f^n), \tag{12}$$

where $DQ(f)(g) = q(f,g)$ is given by formula (9). With this linearization,

the θ-scheme (11) turns to

$$\frac{f^{n+1}-f^n}{\Delta t} = \theta q(f^n, f^{n+1}) + (\frac{1}{2} - \theta) q(f^n, f^n), \tag{13}$$

for any $\theta \in \mathbb{R}$. Note that the contracted scheme (7) is not a θ-scheme and that the linearized implicit scheme used by Epperlein in [10] is obtained for $\theta = 1$. Now we give the main properties of this scheme.

Proposition 2.2.

(i) *The θ-scheme given by (13) is conservative:*

$$\forall n, \quad \int_{\mathbb{R}^d} (1, v, \tfrac{1}{2}|v|^2)^T f^n \, dv = \int_{\mathbb{R}^d} (1, v, \tfrac{1}{2}|v|^2)^T f^0 \, dv.$$

(ii) *The θ-scheme is second order in time if $\theta = \frac{1}{2}$, else it is first order.*

This can be shown using the same arguments as for Proposition 2.1.

Unfortunately, we are not able to prove any entropy property for neither the contracted nor θ-scheme, except in the isotropic case for the contracted scheme (see [18]). Therefore, we propose another strategy that uses the "Log form" (2) of the FPL collision operator. This leads to conservative and entropic schemes that are detailed in the next section.

2.3. "Log" implicit schemes

The first step is to make implicit only the log terms in the "Log form" (2) of the collision operator. This gives the following non-linear implicit scheme:

$$\begin{aligned} \frac{f^{n+1}-f^n}{\Delta t} &= q^{log}(f^n, f^{n+1}) \\ &= \nabla \cdot \int_{\mathbb{R}^d} \Phi(v - v_*) f^n f_*^n \left(\nabla \log f^{n+1} - \nabla \log f_*^{n+1} \right) dv_*. \end{aligned} \tag{14}$$

Proposition 2.3.

(i) *The operator q^{log} satisfies the following weak formulation:*

$$\begin{aligned} &\int_{\mathbb{R}^d} q^{log}(f, g) \phi dv \\ &= -\frac{1}{2} \int_{\mathbb{R}^d} \int_{\mathbb{R}^d} \Phi(v - v_*) f f_* \left(\nabla \log g - \nabla \log g_* \right) \cdot \left(\nabla \phi - \nabla \phi_* \right) dv dv_*, \end{aligned} \tag{15}$$

for any test function ϕ.

(ii) *Scheme* (14) *is conservative:*

$$\forall n, \quad \int_{\mathbb{R}^d} (1, v, \tfrac{1}{2}|v|^2)^T f^n \, dv = \int_{\mathbb{R}^d} (1, v, \tfrac{1}{2}|v|^2)^T f^0 \, dv.$$

(iii) *The collision operator q^{log} dissipates the entropy in the following sense*

$$\int_{\mathbb{R}^d} q^{log}(f, g) \log g \, dv \leq 0.$$

(iv) *Discrete H-theorem: the entropy sequence $H_n = \int_{\mathbb{R}^d} f^n \log f^n \, dv$ is non increasing.*

Again, we refer to [19] for the proof.

Note that $q^{log}(f^n, f^{n+1})$ is non-linear with respect to f^{n+1} which makes it difficult to use in practice. Thus, we propose the following linearization around f^n. We write

$$\log f^{n+1} \approx \log f^n + \frac{f^{n+1} - f^n}{f^n},$$

which is inserted in (14) to obtain the following "log"-linear implicit scheme

$$\frac{f^{n+1} - f^n}{\Delta t} = Q(f^n) + q^l(f^n, f^{n+1}), \tag{16}$$

with

$$q^l(f, g) = \nabla \cdot \int_{\mathbb{R}^d} \Phi(v - v_*) f f_* \left(\nabla \left(\frac{g}{f} \right) - \nabla \left(\frac{g}{f} \right)_* \right) dv_*. \tag{17}$$

Note that $q^l(f, g) = q^l(f, g - f)$ which clearly shows that the scheme is consistent.

Proposition 2.4.

(i) *Weak formulation for q^l:*

$$\int_{\mathbb{R}^d} q^l(f, g) \phi dv = -\frac{1}{2} \int_{\mathbb{R}^d} \int_{\mathbb{R}^d} \Phi(v - v_*) f f_* \left(\nabla \left(\frac{g}{f} \right) - \nabla \left(\frac{g}{f} \right)_* \right) \cdot (\nabla \phi - \nabla \phi_*) \, dv dv_*. \tag{18}$$

(ii) *Scheme* (16-17) *is conservative:*

$$\forall n, \quad \int_{\mathbb{R}^d} (1, v, \tfrac{1}{2}|v|^2)^T f^n \, dv = \int_{\mathbb{R}^d} (1, v, \tfrac{1}{2}|v|^2)^T f^0 \, dv.$$

(iii) *Collisional part of scheme* (16-17) *dissipates the entropy in the following sense:*

$$\int_{\mathbb{R}^d} (Q(f) + q^l(f, g))(\log f + \frac{g}{f}) \, dv \leq 0.$$

(iv) *Discrete H-theorem: the entropy sequence* $H_n = \int_{\mathbb{R}^d} f^n \log f^n \, dv$ *is non increasing if*

$$\inf_{n \in \mathbb{N}, v \in \mathbb{R}^d} \left(\frac{f^{n+1}}{f^n} \right) \geq \frac{1}{2}. \tag{19}$$

(v) *For all positive* f*, the linear operator* $g \mapsto q^l(f, g)$ *is a non-positive self-adjoint operator in the following sense:*

$$\langle q^l(f, g), h \rangle_{\frac{1}{f}} := \int_{\mathbb{R}^d} \int_{\mathbb{R}^d} q^l(f, g) h \, \frac{dv}{f} = \langle q^l(f, h), g \rangle_{\frac{1}{f}},$$

and $\quad \langle q^l(f, g), g \rangle_{\frac{1}{f}} \leq 0, \quad$ *for all* g *and* h.

We refer to [19] for the proof.

Remark 1. The condition (19) of the proposition is reasonable, since in practice the ratio $\frac{f^{n+1}}{f^n}$ should be close to 1. However, to get a weaker condition, one could consider the following modified scheme

$$\frac{f^{n+1} - f^n}{\Delta t} = Q(f^n) + A q^l(f^n, f^{n+1}),$$

where A is a free positive parameter. This scheme is still consistent and condition (19) turns to

$$\inf_{n \in \mathbb{N}, v \in \mathbb{R}^d} \left(\frac{f^{n+1}}{f^n} \right) \geq \frac{1}{2A}.$$

The last property of the proposition is of practical importance since in that case, the Conjugate Gradient (CG) method could be used in the inversion process of the linear system (see details in Section 4. However, the weight $\frac{1}{f^n}$ which is used to construct an inner product in the CG algorithm should be positive at any time. Unfortunately, we cannot prove that this

property is satisfied. We point out that the two first schemes (contracted and θ-schemes) do not have the self-adjointness property. Therefore, we propose modified versions of the θ and log-linear schemes in which the linear operators are self-adjoint at any time (in particular, the positivity of the weight is guaranteed at any time).

2.4. Equilibrium linearized implicit schemes

We write implicit terms $q(f^n, f^{n+1})$ and $q^l(f^n, f^{n+1})$ of schemes (13) and (16-17) in the following perturbative form: $q(f^n, f^{n+1} - f^n) + q(f^n, f^n)$ and $q^l(f^n, f^{n+1} - f^n)$. The idea is to replace the first argument f^n of the perturbative terms $q(f^n, f^{n+1} - f^n)$ and $q^l(f^n, f^{n+1} - f^n)$ by its associate Maxwellian equilibrium f_{eq} of the form (4) that has the same mass, momentum and energy as f^n:

$$q(f^n, f^{n+1}-f^n) \approx q(f_{eq}, f^{n+1}-f^n) \quad \text{and} \quad q^l(f^n, f^{n+1}-f^n) \approx q^l(f_{eq}, f^{n+1}-f^n).$$

This leads to the following schemes:

- Equilibrium θ-scheme:

$$\frac{f^{n+1} - f^n}{\Delta t} = Q(f^n) + \theta q(f_{eq}, f^{n+1} - f^n). \tag{20}$$

  Because f_{eq} is a Maxwellian, it is well known that the linear operator $g \mapsto q(f_{eq}, g)$ is non-positive self-adjoint for the weight $\frac{1}{f_{eq}} > 0$. Moreover, this scheme is conservative.
- Equilibrium "log" linear scheme:

$$\frac{f^{n+1} - f^n}{\Delta t} = Q(f^n) + q^l(f_{eq}, f^{n+1} - f^n). \tag{21}$$

  Because f_{eq} is positive, from property (v) of Proposition 2.4, the linear operator $g \mapsto q^l(f_{eq}, g)$ is non-positive self-adjoint for the weight $\frac{1}{f_{eq}}$. Moreover, this scheme is conservative.

Remark 2. These schemes are not obtained by a linearization of $Q(f^{n+1})$ near the equilibrium f_{eq}.

3. Velocity Discretizations of the FPL Equation: The Isotropic Case

The previous schemes can naturally be discretized in the velocity variable using standard conservative and entropic discretizations [2, 5, 7, 9, 17].

In this section, we first illustrate this assertion with a simple velocity discretization of the FPL equation in the isotropic case. In a second step, we present a more recent strategy based on wavelet approximation theory.

The isotropic FPL model is equation (1) in which the distribution function f only depends on time t and on the particle kinetic energy $\varepsilon = |v|^2$. In this case, the FPL equation writes

$$\partial_t f(t,\varepsilon) = Q(f) = \frac{1}{\sqrt{\varepsilon}} \frac{\partial}{\partial \varepsilon} \int_0^{+\infty} K(\varepsilon, \varepsilon_*) \left(f(\varepsilon_*) \partial_\varepsilon f(\varepsilon) - f(\varepsilon) \partial_\varepsilon f(\varepsilon_*) \right) d\varepsilon_*, \tag{22}$$

with $K(\varepsilon, \varepsilon_*) = \frac{16}{3} \pi \inf(\varepsilon^{3/2}, \varepsilon_*^{3/2})$ for Coulombian interactions, and $K(\varepsilon, \varepsilon_*) = \frac{16}{3} \pi \varepsilon^{3/2} \varepsilon_*^{3/2}$ for Maxwellian interactions.

For any distribution function f, the collision operator $Q(f)$ satisfies the following weak formulation

$$\begin{aligned} &\int_0^{+\infty} Q(f) \phi(\varepsilon) \sqrt{\varepsilon}\, d\varepsilon \\ &= -\frac{1}{2} \int_0^{+\infty} \int_0^{+\infty} K(\varepsilon, \varepsilon_*) \left(f_* \partial_\varepsilon f - f \partial_\varepsilon f_* \right) \left(\partial_\varepsilon \phi - \partial_\varepsilon \phi_* \right) d\varepsilon d\varepsilon_*, \end{aligned} \tag{23}$$

ϕ being any test function. From this formulation, one can immediately derive the following conservation and entropy properties

$$\int_0^{+\infty} (1, \varepsilon)^T Q(f) \sqrt{\varepsilon}\, d\varepsilon = 0, \qquad \int_0^{+\infty} Q(f) \log(f) \sqrt{\varepsilon}\, d\varepsilon \le 0.$$

3.1. Finite difference schemes for the isotropic FPL equation

We briefly recall here the discretization used by Berezin, Khudic and Pekker [4] (also studied by Buet and Cordier [6]). The energy domain is replaced by a regular energy grid of step $\Delta\varepsilon$ and of nodes $\varepsilon_i = (i-1)\Delta\varepsilon$, with $i = 1$ to N. The case of and irregular discretization is also considered at the end of this section. Here, the length of the grid is $e = (N-1)\Delta\varepsilon$. Any function f of ε is approximated on the grid by values $(f_i)_{i=1}^N$ supposed to be approximations of $(f(\varepsilon_i))_{i=1}^N$.

Integrals on $\mathbb{R}^+$ with respect to the measure $\sqrt{\varepsilon} d\varepsilon$ are approximated by

the following weighted trapezoidal quadrature formula

$$\int_0^{+\infty} \phi(\varepsilon)\sqrt{\varepsilon}\, d\varepsilon \approx \sum_{i=1}^{N} \phi(\varepsilon_i)\omega_i, \tag{24}$$

where $\omega_1 = \frac{1}{2}\int_{\varepsilon_1}^{\varepsilon_2} \sqrt{\varepsilon}\, d\varepsilon$, $\omega_i = \int_{\varepsilon_{i-1}}^{\varepsilon_{i+1}} \sqrt{\varepsilon}\, d\varepsilon$ for $i = 2$ to $N-1$, and $\omega_{N-1} = \frac{1}{2}\int_{\varepsilon_{N-1}}^{\varepsilon_N} \sqrt{\varepsilon}\, d\varepsilon$.

A simple discretization of (22) is $\partial_t f = Q(f)$, where $Q(f)$ now is a N-vector of components

$$Q_i(f) = -\frac{1}{\omega_i}(D^*\mathcal{F})_i, \quad i = 1 : N. \tag{25}$$

The operator D^* is defined by $(D^* f)_i = f_{i-1} - f_i$, and the vector $\mathcal{F}$ is defined by

$$\mathcal{F}_i = \sum_{j=1}^{N-1} K_{ij}\Big(f_j (Df)_i - f_i (Df)_j\Big), \quad i = 1 : N-1, \tag{26}$$

and $\mathcal{F}_0 = \mathcal{F}_N = 0$. The finite difference operator D is the formal adjoint of D^* defined by $(Df)_i = f_{i+1} - f_i$. Approximation (25-26) is in fact a second order approximation (see [6]).

This discretization is constructed so that the discrete collision operator satisfies the following weak formulation

$$\sum_{i=1}^{N} Q_i(f)\phi_i\omega_i = -\frac{1}{2}\sum_{i=1}^{N-1}\sum_{j=1}^{N-1} K_{ij}\Big(f_j (Df)_i - f_i (Df)_j\Big)\Big((D\phi)_i - (D\phi)_j\Big).$$

This implies that conservation and entropy properties are preserved (see the proof in [6]), namely we have

$$\sum_{i=1}^{N} Q_i(f)(1, \sqrt{\varepsilon_i})^T \omega_i = 0, \quad \sum_{i=1}^{N} Q_i(f)(\log f_i)\, \omega_i \le 0.$$

Another discretization is deduced from the "log" form of (22) similarly to (2) (see [4]). In that case the discrete collision operator has the same form as (25) but the vector $\mathcal{F}$ now is

$$\mathcal{F}_i = \sum_{j=1}^{N-1} K_{ij} f_i f_j \Big((D\log f)_i - (D\log f)_j\Big), \quad i = 1 : N-1, \tag{27}$$

and $\mathcal{F}_0 = \mathcal{F}_N = 0$. This scheme has the same properties as the "non-log" scheme.

The main drawback of a regular discretization is the fact that the resolution is not accurate near $\varepsilon = 0$, while there are too many points for large ε. Therefore, it is interesting to consider an irregular discretization with a varying energy step $\Delta\varepsilon_i = \varepsilon_{i+1} - \varepsilon_i$ as in [5]. Following the same procedure as for the regular case, we obtain this discrete collision operator

$$Q_i(f) = \frac{1}{\omega_i}(\mathcal{F}_i - \mathcal{F}_{i-1}) \qquad i = 1 : N, \tag{28}$$

with

$$\mathcal{F}_i = \sum_{j=1}^{N-1} K_{ij}\Big(f_j(Df)_i - f_i(Df)_j\Big)\Delta\varepsilon_j, \quad i = 1 : N-1, \tag{29}$$

and $\mathcal{F}_0 = \mathcal{F}_N = 0$. Here D is the following finite difference operator $(Df)_i = \frac{f_{i+1}-f_i}{\Delta\varepsilon_i}$, for $i = 1 : N-1$. This approximation is conservative, but the associated equilibrium states are only approximations of the Maxwellians, *i.e.*

$$(f_{eq})_i = \beta \prod_{j=1}^{i}(1 + \alpha\Delta\varepsilon_j),$$

β and α being some constants. Maxwellian equilibrium states can be obtained by using an analoguous of the "log" form (27) in the case of a non-regular discretisation.

Now let us check the numerical complexity of the discretisation (25) where the flux is given by (26), (27) or (29). For clarity, we only consider the discretization (26) since the two other discretisations can be handeled in the same way. We first notice that the computational cost of a direct evaluation of fluxes (26) is of the order of N^2. However, in the Maxwellian and Coulombian cases, we shall see that this cost can be strongly reduced (to $O(N)$) without any additional approximations. In the next subsection we show that a similar reduction can be obtained in the general case by using more elborated accelerations techniques.

- *The Maxwellian case:* The kernel K has the following expression

$$K(\varepsilon, \varepsilon_*) = C\varepsilon^{3/2}\varepsilon_*^{3/2}. \tag{30}$$

Then the flux (26) can be written as

$$\mathcal{F}_i = C\varepsilon_i^{3/2}\Big(\sum_{j=1}^{N-1}\varepsilon_j^{3/2}f_j\Big)(Df)_i - C\varepsilon_i^{3/2}f_i\Big(\sum_{j=1}^{N-1}\varepsilon_j^{3/2}(Df)_j\Big),\ i=1:N-1, \tag{31}$$

and straightforwardly, this can be computed in $O(N)$ operations only.

- *The Coulombian case:* The kernel K has the following expression

$$K(\varepsilon,\varepsilon_*) = C\min\Big(\varepsilon^{3/2},\varepsilon_*^{3/2}\Big), \tag{32}$$

which gives the following flux

$$\begin{aligned}\mathcal{F}_i \;=\; & C(Df)_i\Big(\varepsilon_i^{3/2}\sum_{j=1}^{i}f_j + \sum_{j=i+1}^{N-1}\varepsilon_j^{3/2}f_j\Big)\\ & -Cf_i\Big(\varepsilon_i^{3/2}\sum_{j=1}^{i}(Df)_j + \sum_{j=i+1}^{N-1}\varepsilon_j^{3/2}(Df)_j\Big),\quad i=1:N-1,\end{aligned} \tag{33}$$

To evaluate this expression in $O(N)$ operations, we first compute recursively the partial summations $\sum_{j=1}^{i}$ and $\sum_{j=i+1}^{N-1}$ for all $i=1:N-1$, which requires a $O(N)$ effort only. Then we evaluate the flux using these precalculated quantities.

Unfortunately, in the general case, that is for general expressions of the kernel K, these simple techniques cannot be used. In other words, the variables ε and ε_* cannot be separated in general. In the anisotropic situation (ie. the full 3D case) and even for the most physical kernel (that is the Coulombian case), the velocities v and v_* cannot be directly separated and some elaborated techniques based on a multiscale resolution, such as the so called *multipole method* [17], or the *multigrid Monte Carlo* [7] method have to be used. These are accelerated versions of the finite difference schemes and lead to a computational cost of the order of $O(N\log N)$. In the isotropic case, it is shown in [1, 2] that the wavelet approximation theory enables to construct high accurate approximations that have a reduced complexity $O(N)$, and satisfy the conservation and entropy properties. In the following we summarize the principle of this method.

3.2. Multiwavelet approximation of the FPL operator

3.2.1. Interpolating scaling and multiwavelet bases

Let M be an integer and $\{q_i\}_{0\le i\le M}$ an $L^2([0,1])$-orthonormal basis of

polynomial functions of degree less than M. We define the scaling functions $\{\varphi_i\}_{0\le i\le M}$ on $\mathbb{R}$ by: $\varphi_i(x) := q_i(x)$ on $[0,1]$ and 0 otherwise. Let V_0 be the space of functions vanishing outside $[0,1]$ whose restrictions to $[0,1]$ are polynomials of degree less or equal to M. Then we have $V_0 = \mathrm{Span}\{\varphi_i\}_{0\le i\le M}$. We briefly recall the multilevel structure of wavelet-type algorithms. For a level n of resolution, we introduce V_n as the space of piecewise polynomial functions vanishing outside $[0,1]$ and whose restriction to each interval $[2^{-n}l, 2^{-n}(l+1)]$ is a polynomial of degree less or equal to M, for $0 \le l \le 2^n - 1$. We then define the dilated-translated scaling functions $\varphi_I^{(n)}(x) = 2^{n/2}\varphi_i(2^n x - l)$, where $I = (i,l) \in \mathcal{I} = \{(i,l)/0 \le i \le M, 0 \le l \le 2^n - 1\}$. We have $V_n = \mathrm{Span}\{\varphi_I^{(n)}\}_{I\in\mathcal{I}}$ and $V_j \subset V_{j+1}$, for $j \ge 0$. Moreover, $\{\varphi_I^{(n)}\}_{I\in\mathcal{I}}$ is an orthonormalized basis of V_n and $\dim V_n = N = (M+1)2^n$. Let us now introduce the corrective space W_n of V_n as the orthogonal complement of V_n in V_{n+1}: $V_n \oplus W_n = V_{n+1}$. Another construction of spaces W_n can be also described from W_0 as follows. Consider an orthonormal basis $\{\psi_i\}_{0\le i\le M}$ spanning W_0 which is the orthogonal of V_0 in V_1. We define the dilated-translated multiwavelet functions $\left(\psi_I^{(n)}\right)_{I\in\mathcal{I}}$, and get $W_n = \mathrm{Span}\{\psi_I^{(n)}\}_{I\in\mathcal{I}}$, for $n \ge 1$.

Up to now, the orthonormal families $\{\varphi_i\}_{0\le i\le M}$ and $\{\psi_i\}_{0\le i\le M}$ are not specified. We are going to give a suitable basis for our approximation. This basis was recently introduced by Alpert *et al.* [3], and our choice is motivated by the localizing property (see below (35)) of these functions and by their "easy to use" character. Let $\mathbb{P}_{M+1}$ be the Legendre polynomial of degree $M+1$ on $[-1,1]$ and $(x_0,\dots,x_M)$ its roots. Consider the Lagrange interpolating polynomials $\{l_i\}_{0\le i\le M}$ at the Gauss-Legendre points $\{x_i\}_{0\le i\le M}$ and let $\{\omega_i\}_{0\le i\le M}$ be the associated Gauss-Legendre weights, then the following family of scaling functions

$$\varphi_i(x) := \sqrt{\frac{2}{\omega_i}}\, l_i(2x-1), \quad \forall x \in [0,1], 0 \le i \le M. \tag{34}$$

is an orthonormal basis in $L^2([0,1])$. Starting from these functions, we construct the spaces V_n and W_n as previously. The so obtained scaling and multiwavelet functions respectively fulfill the following *concentration* and *vanishing* properties

$$\int_0^1 f(x)\varphi_I^{(n)}(x)dx = \alpha_{n,i} f\left(\frac{y_i + k}{2^n}\right) \quad \text{and} \quad \int_0^1 f(x)\psi_I^{(n)}(x)dx = 0, \tag{35}$$

for every $f \in V_n$, with $I = (i,k) \in \mathcal{I}$, $\alpha_{n,i} = 2^{-n/2}\sqrt{\omega_i/2}$ and $y_i = (x_i+1)/2$. Let $f \in L^2([0,1])$ and $P_n f$ the orthogonal projection of f onto V_n. Then we consider the following Galerkin approximation $\overline{f}$ of $P_n f$

$$P_n f \sim \overline{f} = \sum_{I\in\mathcal{I}} \overline{f}_I \varphi_I^{(n)}, \quad \text{with} \quad \overline{f}_I = \alpha_{n,i} f(\frac{y_i+k}{2^n}) \sim \int_0^1 f(x)\varphi_I^{(n)}(x)dx. \tag{36}$$

To discretize $Q(f)$ given by (22), we first define the product $\overline{f} \otimes \overline{g}$, the logarithm $\overline{\ln}(\overline{f})$ and the exponential $\overline{\exp}(\overline{f})$ of two approximations $\overline{f}$ and $\overline{g}$ in V_n, by their components as follows

$$(\overline{f}\otimes\overline{g})_I = \frac{1}{\alpha_{n,i}}\overline{f}_I\overline{g}_I, \quad \overline{\ln}(\overline{f})_I = \alpha_{n,i}\ln(\frac{\overline{f}_I}{\alpha_{n,i}}) \quad \text{and } \overline{\exp}(\overline{f})_I = \alpha_{n,i}\exp(\frac{\overline{f}_I}{\alpha_{n,i}}). \tag{37}$$

Of course, all these approximations are exact for polynomials of degree less than M. To approximate $Q(f)$, we also need to define some discrete derivative operators on V_n. Following [3] we introduce the right decentered derivative operator D defined by the relations

$$[D\overline{f}]_k = R_0[\overline{f}]_k - R_1^T[\overline{f}]_{k+1}, \quad 0 \le k \le 2^n-2, \quad \text{and} \quad \left[D\overline{f}\right]_{2^n-1} = R_{0,r}[\overline{f}]_{2^n-1}, \tag{38}$$

for a function $\overline{f} \in V_n$ whose components in the scaling basis are stored by blocks of length $M+1$, that is $\overline{f} = ([\overline{f}]_k)_{k=0}^{2^n-1} \in V_n$ with $[\overline{f}]_k = (\overline{f}_{i,k})_{i=0}^M$. Blocks R_0, R_1 and $R_{0,r}$ are $(M+1)\times(M+1)$ matrices given by

$$(R_0)_{i,j} = 2^n[-\varphi_i(1)\varphi_j(1) + \sqrt{\frac{\omega_i}{2}}\varphi_j'(y_i)], \quad (R_1)_{i,j} = -2^n\varphi_i(0)\varphi_j(1)$$

and

$$(R_{0,r})_{i,j} = 2^n\sqrt{\frac{\omega_i}{2}}\varphi_j'(y_i),$$

for $0 \le i,j \le M$, where the derivatives of the scaling functions are exactly computed. We also introduce the adjoint (or transposed) operator D^* of D. Note that the order of accuracy of the above approximations is the number M of scaling functions at level 0. Indeed, all these discretizations are exact for polynomials of degree less that M (see [3] and [1] for details).

To proceed to the approximation of the collision operator, we first restrict the integration domain to $[0,1]$ and keep the same notation for $Q(f)$

$$\sqrt{\varepsilon}Q(f)(\varepsilon) = \frac{\partial}{\partial\varepsilon}\Big[\frac{\partial \ln f}{\partial\varepsilon}(\varepsilon)f(\varepsilon)(Tf)(\varepsilon) - f(\varepsilon)T\Big(\frac{\partial \ln f}{\partial\varepsilon}f\Big)(\varepsilon)\Big], \tag{39}$$

with

$$Tg(\varepsilon) = \int_0^1 K(\varepsilon, \varepsilon')g(\varepsilon')d\varepsilon'. \tag{40}$$

On V_n this operator T is represented by the matrix $T^{(n)}$ whose coefficients are

$$T^{(n)}_{I,J} = \int_0^1 \int_0^1 K(\varepsilon, \varepsilon')\varphi^{(n)}_I(\varepsilon)\varphi^{(n)}_J(\varepsilon')d\varepsilon d\varepsilon'. \tag{41}$$

Using the first formula of (35), we then get an approximation $\mathcal{T} = (\mathcal{T}^{(n)}_{I,J})_{(I,J)\in\mathcal{I}^2}$ of the matrix $T^{(n)}$, whose coefficients are

$$\mathcal{T}^{(n)}_{I,J} = \alpha_{n,i}\alpha_{n,j}K(\frac{y_i + k}{2^n}, \frac{y_j + l}{2^n}). \tag{42}$$

Now, let $\overline{f}$ be the approximation of f defined by (36). Using the previous approximations of the logarithm, the exponential and the product of two elements of V_n, we can derive an approximation (in V_n) of the collision operator (39). This is the subject of the following subsection.

3.2.2. Conservative and entropic multiwavelet approximations for the isotropic FPL operator

Using the previous notations, we state the following approximation for the FPL operator.

Proposition 3.1. *Let $\overline{f}$ be the approximation of the unknown distribution function f defined by* (36)*, $\overline{\ln}$ be the logarithm function on V_n given by* (37) *and consider the approximation $\mathcal{T}$ of the integral operator T defined by* (41)*. Under these notations, we introduce the following operator on the space V_n*

$$\mathcal{Q}(\overline{f}) = -D^*[(D\overline{\ln}(\overline{f})) \otimes \overline{f} \otimes (\mathcal{T}\overline{f}) - \overline{f} \otimes \mathcal{T}((D\overline{\ln}(\overline{f})) \otimes \overline{f})], \tag{43}$$

where D is the derivative operator (38) *and D^* its adjoint operator. The operator* (43) *is an approximation (on V_n) of $\sqrt{\varepsilon}Q(f)$ given by* (39)*, that satisfies the following weak discrete formulation*

$$\begin{aligned}\sum_{I\in\mathcal{I}}[\mathcal{Q}(\overline{f})]_I\Theta_I \;=\; & -\frac{1}{2}\sum_{I,J\in\mathcal{I}^2}\mathcal{T}_{I,J}\overline{f}_I\overline{f}_J\Big[\frac{1}{\alpha_{n,i}}(D\Theta)_I - \frac{1}{\alpha_{n,j}}(D\Theta)_J\Big] \\ & \times\Big[\frac{1}{\alpha_{n,i}}(D\overline{\ln}(\overline{f}))_I - \frac{1}{\alpha_{n,j}}(D\overline{\ln}(\overline{f}))_J\Big],\end{aligned} \tag{44}$$

for any element $\Theta \in V_n$. In particular, it satisfies the conservation of the discrete mass and energy, and the discrete entropy dissipation property respectively given by

$$\sum_{I\in\mathcal{I}}[\mathcal{Q}(\overline{f})]_I\overline{\Theta}_I \;=\; 0, \quad \textit{for} \;\; \Theta = 1, \varepsilon$$

and

$$\sum_{I\in\mathcal{I}}[\mathcal{Q}(\overline{f})]_I(\overline{\ln}(\overline{f}))_I \;\leq\; 0, \quad \forall \overline{f} \in V_n, \;\; \overline{f}_i \geq 0, \tag{45}$$

with equality for this last relation if and only if $\overline{f}$ is a maxwellian: $\overline{f} = \overline{\exp}(A\overline{\varepsilon} + B\overline{1})$, where $(A, B) \in \mathbb{R}^2$.

Scheme (44) can be viewed as a generalization of that obtained in [9]. Indeed, taking $M = 1$ in the definition of the approximation space V_n, (44) coincides with the finite difference scheme developed in [9]. The scheme (44) allows high accuracy on a non-regular grid.

Let us now analyse the numerical complexity of the so obtained discretisation. Of course, a direct evaluation of $\mathcal{T}\overline{f}$ would require $\mathcal{O}(N^2)$ operations, where $N = 2^n(M + 1) = \dim V_n$. However, this cost can be reduced to $\mathcal{O}(N)$ operations using standard wavelet techniques that are based on a multiscale algorithm [13] with suitable successive approximations. More precisely, instead of computing $\mathcal{T}\overline{f}$ in the scaling basis $\left(\varphi_I^{(n)}\right)_{I\in\mathcal{I}}$, we perform this calculation on a basis of $V_{n-1} \oplus W_{n-1}$ that mixes the scaling functions and the multiwavelets $\left(\psi_I^{(n-1)}\right)_{I\in\mathcal{I}}$. Indeed, many coefficients of the matrix $\mathcal{T}$, when written in this new basis, can be neglected. This is due to the vanishing property (35) of the multiwavelets ψ. We then repeat recursively the same procedure for V_{n-1}, V_{n-2}, ... ,V_1. In other words, a suitable choice of the basis of V_n mixing the scaling functions with the multiwavelets at successive levels of resolution $m = n, n-1, ..., 0$ yields a new matrix representation of T that can be well approximated by a sparse one. We refer to [1] for a detailed presentation of this algorithm and for some error estimates. We also note that such a procedure does not affect the conservation property of the original discretisation since it preserves the symmetry of the matrix $\mathcal{T}$. However, we are not able to prove whether the entropy dissipation is still satisfied or not, even if the numerical tests show that this property remains true.

Now, we point out that a 'non-log' multiwavelet approximation can be obtained similarly. Using the same notations as previously the 'non-log'

analogous of (43) writes

$$\mathcal{Q}_{nonlog}(\overline{f}) = -D^*[(D\overline{f} \otimes (\mathcal{T}\overline{f}) - \overline{f} \otimes (\mathcal{T}D\overline{f})]. \tag{46}$$

This approximation satisfies the following weak formulation

$$\begin{aligned}\sum_{I\in\mathcal{I}}[\mathcal{Q}_{nonlog}(\overline{f})]_I\Theta_I \;=\; & -\frac{1}{2}\sum_{I,J\in\mathcal{I}^2}\mathcal{T}_{I,J}[\frac{1}{\alpha_{n,i}}(D\Theta)_I - \frac{1}{\alpha_{n,j}}(D\Theta)_J] \\ & \times[\frac{1}{\alpha_{n,i}}\overline{f}_J(D\overline{f})_I - \frac{1}{\alpha_{n,j}}\overline{f}_I(D\overline{f})_J],\end{aligned} \tag{47}$$

which, in particular, proves the conservation of mass and energy. Again, the entropy property is only checked numerically.

Consequently, we have a discrete collision operator that possesses all desired properties (both in "log" and "non-log" forms). Our implicit schemes can now be derived exactly as in the continuous case, as it is illustrated in the next section.

3.3. Completely discretized implicit schemes

In this section, we give completely discretized implicit schemes in both time and velocity variables.

3.3.1. Using finite difference discretisation

For the sake of simplicity we only present the case of a regular discretization. The case of an irregular grid can be treated in the same way. The different implicit schemes are given by equations (7, 13, 16, 20, 21), where the discrete collision operators are the N-vectors defined as follows:

- contracted scheme: $q_i^c(f,g) = -\frac{1}{\omega_i}(D^*\mathcal{F}^c(f,g))_i$, for $i = 1:N$, where the flux $\mathcal{F}^c$ is

$$\mathcal{F}_i^c(f,g) = \sum_{j=1}^{N-1} K_{ij}\Big(f_j(Dg)_i - f_i(Dg)_j\Big), \quad i = 1:N-1.$$

- θ-scheme: $q_i(f,g) = q_i^c(f,g) + q_i^c(g,f)$, $i = 1:N$.
- log-linear scheme: $q_i^l(f,g) = -\frac{1}{\omega_i}(D^*\mathcal{F}^l(f,g))_i$, for $i = 1:N$, where the

flux $\mathcal{F}^l$ is

$$\mathcal{F}_i^l(f,g) = \sum_{j=1}^{N-1} K_{ij} f_i f_j \Big[\Big(D\left(\frac{g}{f}\right)\Big)_i - \Big(D\left(\frac{g}{f}\right)\Big)_j \Big], \quad i = 1 : N-1.$$

For all these definitions, the numerical fluxes are zero for $i = 0$ and $i = N$, and D and D^* are the finite difference operators defined in Section 3.1. Following the same strategy as in the continuous case, we can write the discrete versions of equilibrium schemes (20) and (21). In that case, the discrete equilibrium is the discrete Maxwellian $f_{eq,i} = \exp(\alpha + \beta\varepsilon_i)$ whose coefficients α and β are computed so as f_{eq} has the same discrete mass and energy as f.

3.3.2. Using multiwavelet approximations

The different implicit schemes are given by equations (7, 13, 16, 20, 21), where a function f is replaced by its approximation $\overline{f} \in V_n$, and where the discrete collision operators are the N-vectors defined as follows:

- contracted scheme: The contracted operator q^c is now replaced by its following approximation on V_n, according to the scheme and the notations presented in section 3.2,

$$\overline{q}^c(\overline{f},\overline{g}) = -\overline{\Big(\frac{1}{\sqrt{\varepsilon}}\Big)} \otimes D^* \mathcal{W}^c(\overline{f},\overline{g}),$$

  where the flux $\mathcal{W}^c$ is

$$\mathcal{W}^c(\overline{f},\overline{g}) = D\overline{g} \otimes (\mathcal{T}\overline{f}) - \overline{f} \otimes (\mathcal{T} D\overline{g}) \tag{48}$$

- θ-scheme: the operator q is now replaced by its approximation on V_n

$$\overline{q}(\overline{f},\overline{g}) = \overline{q}^c(\overline{f},\overline{g}) + \overline{q}^c(\overline{g},\overline{f})$$

.

(1) log-linear scheme: In V_n, the operator ql is approximated by

$$\overline{q}^l(\overline{f},\overline{g}) = -\overline{\Big(\frac{1}{\sqrt{\varepsilon}}\Big)} \otimes D^* \mathcal{W}^l(\overline{f},\overline{g}),$$

where the flux $\mathcal{W}^l$ is

$$\mathcal{W}^l(\overline{f},\overline{g}) = D\overline{\Big(\frac{g}{f}\Big)} \otimes \overline{f} \otimes \mathcal{T}\overline{f} - \overline{f} \otimes \mathcal{T}\Big(\overline{f} \otimes D\overline{\Big(\frac{g}{f}\Big)}\Big). \tag{49}$$

It is now an easy matter to prove that these discrete collision operators satisfy weak formulations similar to those satisfied by the continuous model. Hence the discrete analogous properties of Propositions 2.1, 2.2 and 2.4 are satisfied by all these discrete schemes.

4. Linear Solvers

In this section, we present a strategy to solve the linear implicit schemes (7, 13, 16, 20, and 21), the unknown being f^{n+1}. The present method obeys the conservation properties as soon as the approximation of the collision operator Q is conservative. Here, we deal with both the FPL equation (1) with $d = 2$ or 3 and the isotropic case (22). In this section, these models only differ by the dimension of the integration domain: $d = 2$ or 3 for (1) and $d = 1$ for (22). We only develop the strategy for the finite difference discretisations of the collision operator since the multiwavelet approach can be used in a similar way.

We first assume that the collision operator is discretized with N velocity points and write the schemes (7, 13, 16) in the following matrix-forms:

contracted scheme (7)	$L^c(f^n)f^{n+1} = f^n,$
θ-scheme (13)	$L_\theta(f^n)f^{n+1} = f^n + \Delta t(1-2\theta)Q(f^n),$
log-linear scheme (16)	$L^l(f^n)f^{n+1} = f^n + \Delta t Q(f^n),$

where $L^c(f)$, $L_\theta(f)$ and $L^l(f)$ are the $N \times N$-matrices defined by

$$L^c(f)g = g - \Delta t\, q^c(f,g), \tag{50}$$

$$L_\theta(f)g = g - \Delta t\, \theta q(f,g), \tag{51}$$

$$L^l(f)g = g - \Delta t\, q^l(f,g), \tag{52}$$

for every vector g in $\mathbb{R}^N$.

We can also consider the following equivalent (and more convenient) δ-form

$$L^c(f^n)\delta f^n = \Delta t Q(f^n), \tag{53}$$

$$L_\theta(f^n)\delta f^n = \Delta t Q(f^n), \tag{54}$$

$$L^l(f^n)\delta f^n = \Delta t Q(f^n), \tag{55}$$

where $\delta f^n = f^{n+1} - f^n$. Equilibrium implicit schemes (20) and (21) are defined in δ-form by

$$L_\theta(f_{eq})\delta f^n = \Delta t Q(f^n), \tag{56}$$

$$L^l(f_{eq})\delta f^n = \Delta t Q(f^n). \tag{57}$$

Before going to the resolution of these linear systems, we give some of their important algebraic properties.

Proposition 4.1.

- *For all $\Delta t > 0$ and all $f > 0$, $L^l(f)$ is a positive definite self-adjoint matrix for the inner product with weight $\frac{1}{f}$, and, in particular, it is invertible.*
- *For all $\Delta t > 0$, $\theta \in [0,1]$, and all discrete Maxwellian f_{eq}, $L_\theta(f_{eq})$ is a positive definite self-adjoint matrix for the inner product with weight $\frac{1}{f_{eq}}$, and, in particular, it is invertible.*
- *For all $f > 0$, $L^c(f)$ and $L_\theta(f)$ are invertible if Δt is small enough.*

Proof. We do not give the proof of this proposition, since it is a direct consequence of the discrete versions of the properties of q^c, q, and q^l given in Propositions 2.1, 2.2 and 2.4. □

These linear systems are non sparse, with generally large dimension. For a numerical resolution, one can mainly investigate three classes of methods: direct (as LU), iterative non-stationary (as Krylov methods), iterative stationary (as relaxation methods).

Direct methods have been used by Epperlein [10] for the isotropic FPL equation with LU factorization. However, it is clear that such solvers are not usable in multidimensional cases. The complexity of the algorithms is $O(N^3)$, and the memory storage for the matrices is $O(N^2)$. Even in the one dimensional case as isotropic equation, the total complexity for a given simulation time is asymptotically the same as the simple explicit scheme (see Section 5).

Consequently, it is clear that iterative methods must be considered. At this stage, we want to point out a crucial fact: since iterative methods generally construct an approximate solution to the linear system, *it should be investigated whether conservation properties of the implicit schemes are preserved or not.* Indeed, it is questionable to construct perfectly conservative schemes if that property is destroyed by the linear solvers.

To make this point more precise, we define the $(d+2)\times N$ matrix M that associates to every vector $f \in \mathbb{R}^N$ its corresponding $(d+2)$-vector of moments Mf (*i.e* an approximation of $\int_{\mathbb{R}^d}(1, v, \frac{1}{2}|v|^2)^T f\, dv$). For instance, we can set $Mf = \sum_{i=1}^N (1, v_i, \frac{1}{2}|v_i|^2)^T f_i\, \Delta v^d$ for a regular discretization and $d = 2$ or 3. For the isotropic case, a possible definition is $Mf = \sum_{i=1}^N (1, \varepsilon_i)^T f_i \sqrt{\varepsilon_i}\Delta\varepsilon$. Since we assume the discrete collision operator Q to be conservative, the following relations are satisfied for every f and g in $\mathbb{R}^N$:

$$Mq^c(f,g) = 0, \qquad Mq(f,g) = 0 \quad \text{and} \quad Mq^l(f,g) = 0.$$

This implies the following relations for the matrices (50,51,52)

$$ML^c(f) = M, \qquad ML_\theta(f) = M \quad \text{and} \quad ML^l(f) = M.$$

Multiplying by M the δ-forms (53-57) of our implicit schemes is another way to check the conservation property $Mf^{n+1} = Mf^n$.

Consequently, the implicit schemes written under the δ-forms (53-57) are of the same type as the following general linear system in $\mathbb{R}^N$

$$Ax = b,$$

where the $N \times N$-matrix A and N-vector b satisfy

$$MA = M \text{ and } Mb = 0. \tag{58}$$

Then we consider the following problem: find an iterative solver such that if we start with an initial vector $x^{(0)}$ satisfying $Mx^{(0)} = 0$, then any iterate $x^{(k)}$ also satisfies $Mx^{(k)} = 0$. Such solvers will be called *conservative iterative linear solvers*. They guarantee that the implicit scheme is still conservative even if convergence to the exact solution of the linear system is not achieved.

In the sequel, we claim that Krylov subspace methods are conservative iterative linear solvers. In our study, these methods can be set in the following general frame (see [23] for an introduction to Krylov solvers).

Algorithm 4.1.

1. *give* $x^{(0)}$ *such that* $Mx^{(0)} = 0$ *and set* $r^{(0)} = b - Ax^{(0)}$*;*
2. *for* $k = 1$ *to* K*, find* $x^{(k)}$ *in the affine subspace* $x^{(0)} + \mathcal{K}_k$*, where*

$$\mathcal{K}_k = \{r^{(0)}, Ar^{(0)}, \ldots, A^{k-1}r^{(0)}\}.$$

The different versions of Krylov methods arise from different choices of $x^{(k)}$ in $\mathcal{K}_k$. We now state the following

Proposition 4.2. *All iterative methods that can be set under the form of Algorithm* 4.1 *are conservative. This means that we have* $Mx^{(k)} = 0$ *for every* k.

It is remarkable that this conservation property holds even if the linear system is not exactly solved. Another advantage of these methods is well known: they are "matrix-free", i.e. the matrix A only appears in matrix-vector products Ay in the solver. Thus we do not need to form and store the matrix A. Moreover in our case, if the quantities $q^c(f,g)$, $q(f,g)$ and $q^l(f,g)$ can be computed in $O(N)$ operations, then this is also possible for products Ay in Krylov solvers, since $A = L^c(f^n), L_\theta(f^n)$, or $L^l(f^n)$ which are given by (50,51,52). Finally, we just emphasize that all iterative solvers not necessarily conservative, and refer to [19] for some remarks on this subject.

Among the above Krylov methods, we mainly use the GMRES and the Conjugate Gradient (CG) methods. The CG method is used to solve linear systems whose matrices are positive definite self-adjoint. As stated in Proposition 4.1, this is the case for the log-linear and equilibrium log-linear schemes (16, 21), and the equilibrium θ-scheme (20). This is a real advantage, since the CG method is one of the most efficient Krylov solver (in terms of CPU cost and memory storage). For the other schemes, we simply use the GMRES method.

5. Complexity of the Algorithms

In this section, we assume that for an arbitrary number of dimensions $d \geq 1$, the number of operations for computing $q^c(f,g), q(f,g)$, and $q^l(f,g)$ is $O(N)$. This is true for $d = 1$ in the case of isotropic equation with Coulomb or Maxwell potential using finite difference approximations, and for other potentials using multiwavelet algorithms [2]. For $d \geq 2$, cost reductions to $O(N \log N)$ can be obtained through rapid algorithms as Multipole methods [17] and spectral methods [21]. The cost $O(N)$ could be reached by an extension of the isotropic wavelet methods [2] to the multi-dimensional case. We also assume that the velocity domain is discretized with a step Δv or $\Delta\varepsilon = \frac{1}{n}$ in each directions and a total number of points of $N = n^d$. For a problem with a time scale τ, we want to compare the complexity of the usual Euler explicit scheme (5) and our linear implicit schemes for a given simulation time T.

For the explicit scheme, one iteration requires only one evaluation of $Q(f)$, which costs $O(N)$. Assume that the CFL condition imposes a time step $\Delta t = O(\frac{1}{n^2})$ (this seems to be true from numerical experiments and has been proved in the isotropic case [5]). Then the number of iterations is $\frac{T}{\Delta t} = O(n^2)$. Consequently, the total complexity is $O(N^{1+2/d})$.

For linear implicit schemes with Krylov solvers, the cost of one Krylov iteration is $O(N)$ (since only evaluations of $q^c(f,g), q(f,g)$, and $q^l(f,g)$ are needed). We assume that K iterations of the Krylov solver are necessary to get a correct approximation of the time iterate. Thus the cost of one time iteration is $KO(N)$. Finally, if the time step can be set to τ, the number of time iterations is $\frac{T}{\tau}$, and therefore, the total complexity is $KO(N)$.

In Table 1, we give a comparison of these complexities for $d = 1, 2, 3$.

Table 1. Complexity of explicit and linear implicit schemes for $d = 1, 2, 3$.

	$d=1$	$d=2$	$d=3$
explicit	$O(N^3)$	$O(N^2)$	$O(N^{5/3})$
implicit	$KO(N)$	$KO(N)$	$KO(N)$

We can see that for $d = 1$, there is an important gain even if $K = N$. For $d = 2$, there is still a gain if $K << N$. If $K = O(N)$, the explicit and implicit schemes have the same asymptotic cost $O(N^2)$. For $d = 3$, we have a significant gain only if $K << N^{2/3}$. Therefore, the efficiency of the implicit schemes decreases as the dimension increases. Consequently, the reduction of the number K of Krylov iterations becomes necessary in multi-dimensional cases. This could be done by using adapted preconditioning techniques. As noted by [8], such techniques can render K virtually independent of N. In that case, our implicit schemes would always be advantageous as compared to the explicit scheme. This is the subject of a future work.

6. Numerical Tests

In order to check the properties of the implicit schemes that we introduced in this work, we present various numerical tests with both the Maxwellian and Coulombian potentials.

First, in the case of Maxwellian potential, it is well known that the FPL operator reduces to the so-called Ornstein-Uhlenbeck operator [11] and that

the homogenous FPL equation can be exactly solved for any initial data. In particular, an explicit and practical example of solution is given in [16] by

$$f(t,\varepsilon) = \frac{\rho}{(2\pi T)^{3/2}} \exp(-\frac{\varepsilon}{2T}) \left(1 + \frac{11}{120}\left(\left(\frac{\varepsilon}{T}\right)^2 - 10\frac{\varepsilon}{T} + 15\right)\exp(-8\rho t)\right).$$

In Figure 1 and 2, we compare the numerical solution obtained with the second-order θ-scheme (13) to this exact solution, using finite difference approximation. The energy domain is $[0, 2]$ discretized with 500 points. The parameters of the exact solution are $\rho = 2$ and $T = 0.01$. The time step used with the implicit scheme is about 300 times the time step required by an explicit computation. In Figure 1, we plot f as a function of $v = \sqrt{\varepsilon}$ at different time steps. We observe that the numerical solution is very close to the exact one, and the equilibrium is reached in only ten time steps. This corresponds to a final physical time equal to $t_{max} = 10$. This shows that the dynamics described by the exact FPL equation can efficiently be simulated with a much larger time step than those of usual explicit schemes. This is also clear on Figure 2 where the kinetic entropy is plotted. The slight difference that can be observed in the stationary regime is due to the velocity discretization itself. In fact the discrete Maxwellian is different from the exact one.

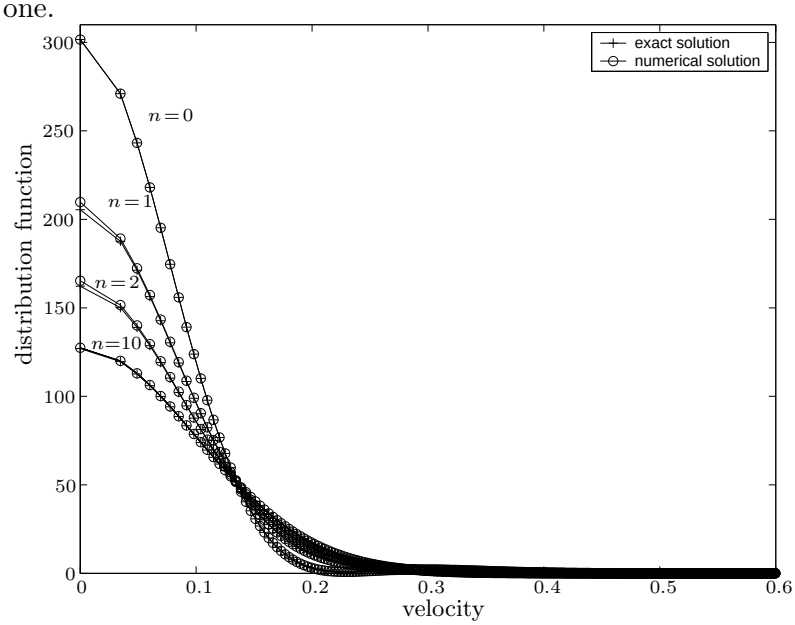

Figure 1. Exact solution and implicit scheme for Maxwellian potential: distribution function at different time steps (finite difference approximation).

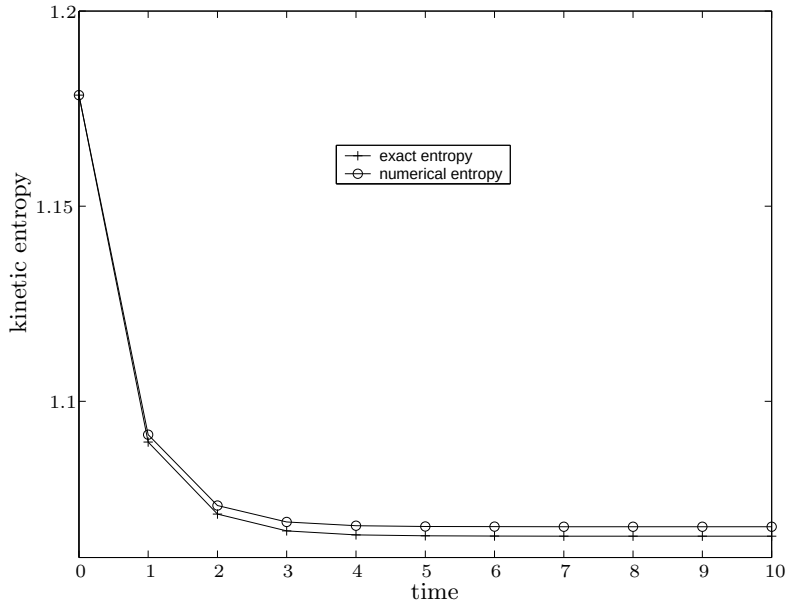

Figure 2. Exact solution and implicit scheme for Maxwellian potential: entropy.

The second test case uses Coulombian potential with the so called Rosenbluth initial data:

$$f^0(\varepsilon) = 0.01 \exp(-10((\sqrt{\varepsilon} - 0.3)/0.3)^2).$$

On Figure 3, we plot the kinetic entropy obtained by the explicit scheme (5) with time step Δt_{exp}, which is the largest step ensuring the stability of the scheme. This entropy is compared with that obtained by the contracted implicit scheme (7). For this last scheme, we take a time step $\Delta t_{imp} = 50\Delta t_{exp}$. We use a regular energy discretization of $[0, 2]$ with $N = 100$ points. Tolerance for GMRES algorithm is 10^{-6}. The dynamics is well described by the implicit scheme even if the time step is much larger. However the gain in terms of CPU time is not obtained unless the number of energy points is sufficiently large. Indeed, the implicit scheme is really advantageous for $N \geq 500$ only. This is clearly shown by Figure 4 where the CPU time versus N is plotted for explicit and implicit schemes. According to what we explained in Section 5, the CPU time of the explicit scheme behaves as $O(N^3)$. A contrario, the implicit scheme only requires $O(n_K N)$ operations, where n_K is the number of iterations in the Krylov procedure. It is known that $n_K \leq N$, and we observe that on this test case, n_K is much smaller

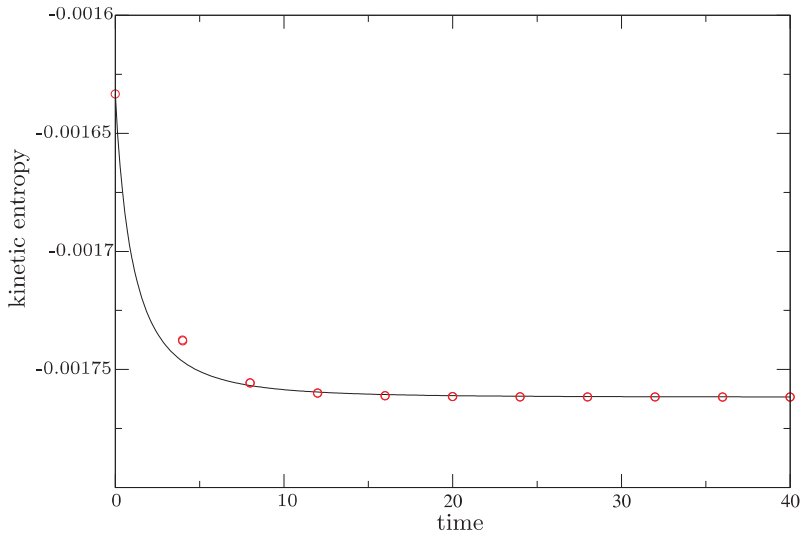

Figure 3. Kinetic entropy for explicit (-) and contracted implicit scheme (7) (o) for Coulombian potential. Case of a regular grid.

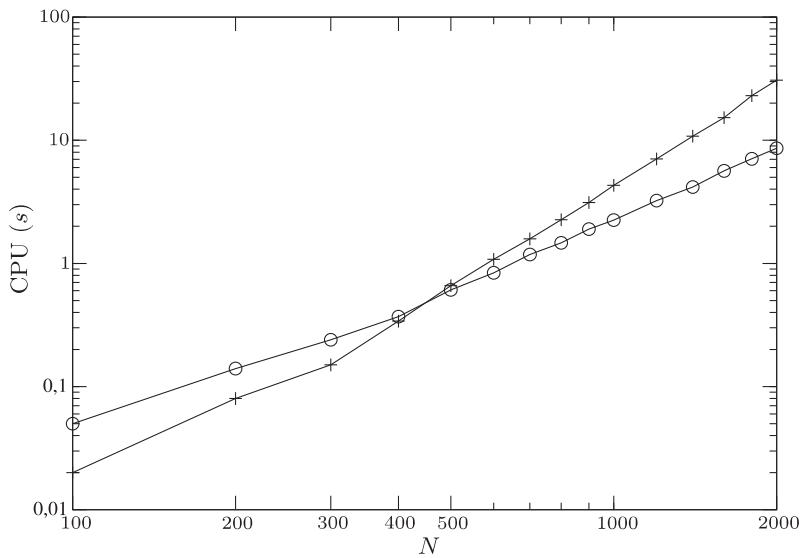

Figure 4. CPU cost of explicit (+) and contracted implicit scheme (7) (o) versus the number N of energy points in a logarithmic scale (Coulombian potential). Tolerance for GMRES is 10^{-6}, time step in implicit scheme is 0.3.

than N. The test on Figure 4 confirms that the numerical complexity of the implicit scheme behaves like $O(N^2)$. Indeed, for $N \geq 500$, the slope of the curve is 1.95 for the implicit scheme whereas it is 2.8 for the explicit scheme.

Then, we perform the same test case with an irregular energy discretization where the discrete energy points are $\varepsilon_i = 2(\frac{i}{100})^3$ for $i = 1$ to 100. First, we observe that the entropy is almost the same as for the regular grid (see Figure 5). However, it is remarkable that even with only 100 points, the implicit scheme is less expensive in terms of CPU time (it requires 0.12 s) than the explicit scheme (which requires 0.5 s).

For the same initial data, we compare the different implicit schemes studied in this work to the explicit one: contracted (7), θ-scheme (13), log-linear implicit scheme (16), equilibrium θ-scheme (20), equilibrium log-linear scheme (21). On Figure 6 we plot the fourth order moment $\int f(\varepsilon)\varepsilon^2\sqrt{\varepsilon}\,d\varepsilon$ as a function of time. We only compare first order (in time) schemes with $\Delta t_{imp} = 50\Delta t_{exp}$ and $N = 100$ on a regular grid. We observe that the equilibrium θ-scheme with $\theta = 1$ is the most accurate in this case. Despite its good mathematical properties, the log-linear scheme is not sufficiently accurate. This is probably due to the linearization of the log function.

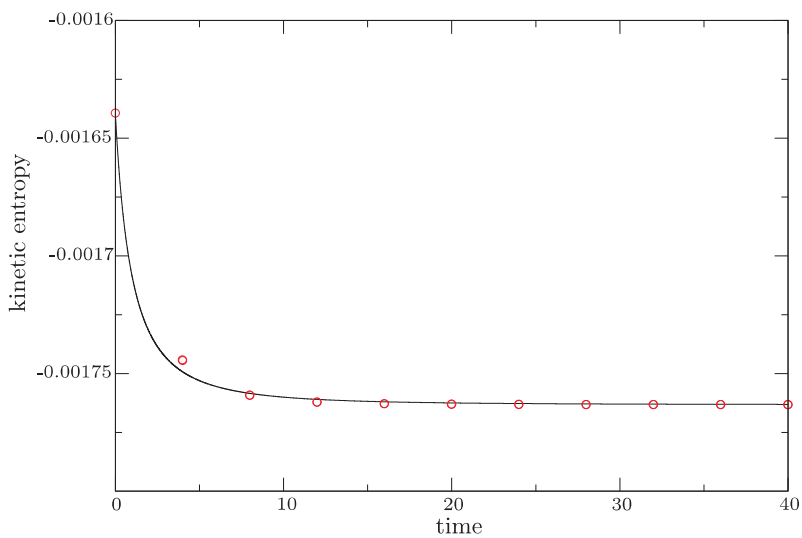

Figure 5. Kinetic entropy for explicit (-) and implicit scheme (7) (o) for Coulombian potential. Case of an irregular grid.

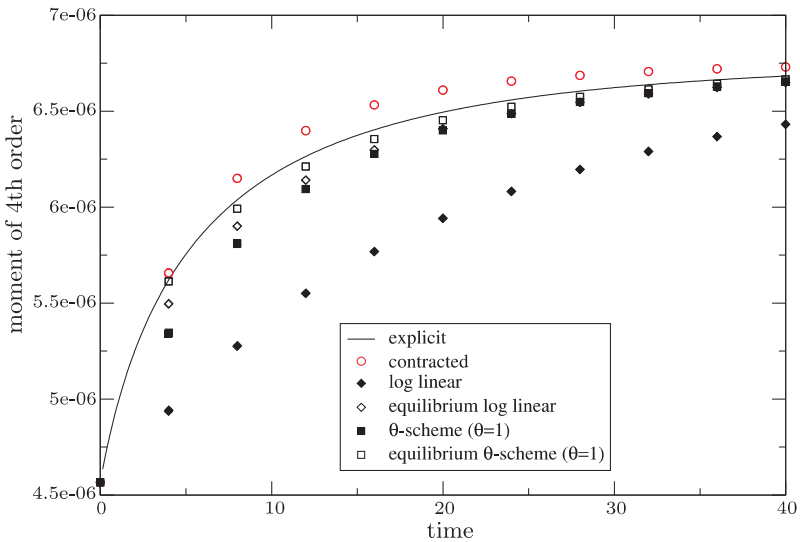

Figure 6. Fourth order moment for different implicit schemes.

On Figure 7, we compare our second order (in time) θ-scheme (with $\theta = 0.5$) to a second order explicit scheme (Runge-Kutta scheme). We use the same parameters as on Figure 6. We can see that the two results are in a very good agreement.

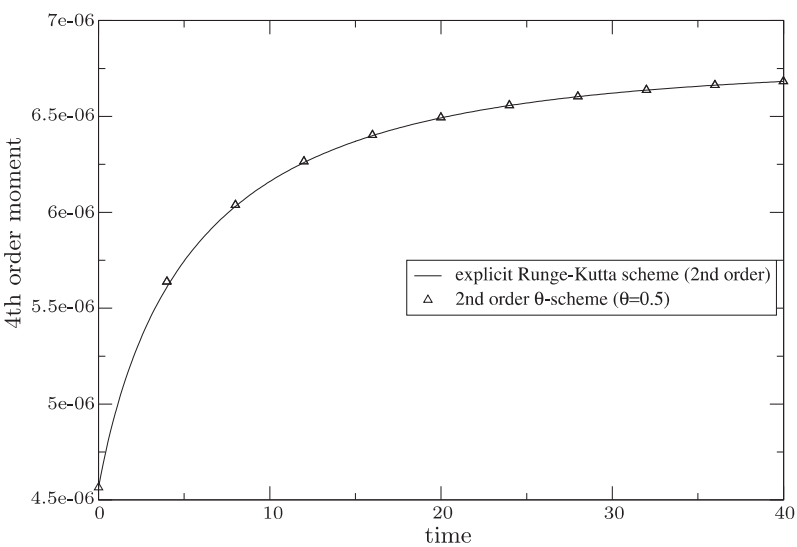

Figure 7. Fourth order moment for second order (in time) schemes.

As a prototype for the multidimensional case, in which simple acceler-

ation techniques as (31) or (33) cannot be used, we consider the isotropic FPL equation derived from (1) for a general parameter γ. In that case, the FPL equation still has the form (22), but the kernel K has now the following extended normalized expression:

$$K(\varepsilon,\varepsilon') = \frac{-1}{(\gamma+2)(\gamma+4)(\gamma+6)}\Big((\varepsilon^{1/2}+\varepsilon'^{1/2})^{\gamma+4}(\varepsilon-(\gamma+4)\varepsilon^{1/2}\varepsilon'^{1/2}+\varepsilon') - |\varepsilon^{1/2}-\varepsilon'^{1/2}|^{\gamma+4}(\varepsilon+(\gamma+4)\varepsilon^{1/2}\varepsilon'^{1/2}+\varepsilon')\Big). \quad (59)$$

In this case, we use a more elaborated technique to accelerate the computation of the collision operator. The method uses the wavelet approximation theory and is summarized in section 3.2. To illustrate this strategy, we give a numerical test and plot the entropy on Figure 8. Again, the implicit scheme gives the same accuracy as the explicit scheme with a cost reduction comparable to the previous cases.

Finally, we point out that all the schemes proposed in this paper are perfectly conservative, up to the machine error. For instance, on the second test case, the variation amplitude of density and energy during the time evolution is about 10^{-16}, as for the explicit scheme. This confirms that the approximate resolution of the linear systems does not affect the conservation properties.

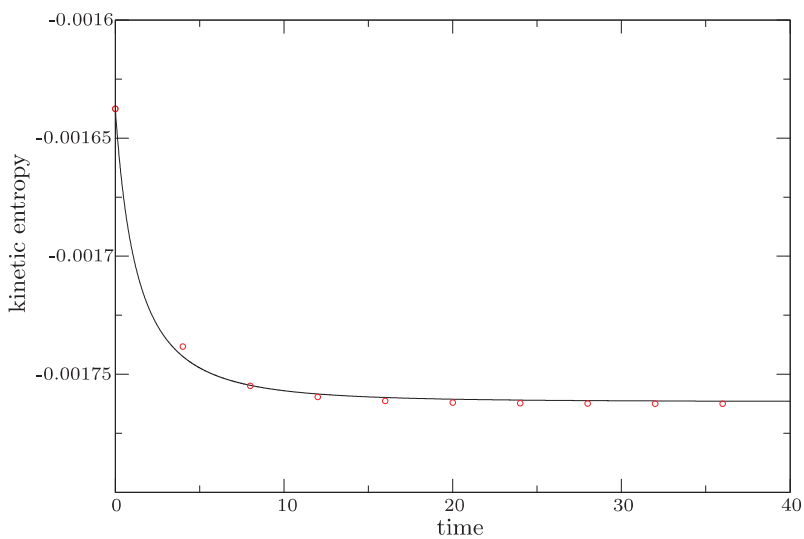

Figure 8. Kinetic entropy for explicit (-) and implicit scheme (7) (o) for Coulombian potential, with wavelet approximation presented in Section 3.2.

7. Conclusion

In a first part of this paper, we have constructed various linear time-implicit schemes to solve the homogeneous FPL equation, reviewing the results in [18, 19]. These strategies are shown to satisafy important physical properties of conservation and entropy for arbitrary dimension of the space variable. Then, in a second part, we present some discretisations and develop some fast algorithms to evaluate the collision part in the isotropic case, preserving the above physical properties. Moreover, numerical tests in the isotropic case have shown a significant gain in terms of CPU time with the same accuracy as that obtained by usual explicit schemes.

As we pointed out in the introduction, the time-implicitation strategy applies to multidimensional collision operators, but still an important point must be investigated in that case. Indeed, the involved linear systems are much larger, and suitable preconditioners are required. We recall that, in a Krylov method with a good preconditioner, the number of iterations is independent of N. Thus the complexity of our implicit schemes could be reduced to $O(N)$. Moreover, the rapid matrix-vector product (partial summations techniques or multiwavelets) that has been used in the isotropic case cannot be directly applied to the multidimensional cases. Therefore, to get a fast implicit solver in several dimensions, more adapted acceleration techniques are required. A first step will be to use some existent fast algorithms such as the multipole method [17]. This task is under investigation. The second step is to extend the multiwavelt strategy to the 3D case and find systematic and suitable preconditionners. This will be the subject of a futur work.

Note that for the inhomogeneous (space dependent) case, our implicit schemes should be more efficient. Indeed, in many inhomogeneous situations the transition phase is rapid and does not need to be accurately described. In such cases, our implicit schemes allow to directly reach the hydrodynamic behavior with a few time steps.

Finally, the extension of our strategy to other collision operators (including the relativistic and quantum effects for instance) is the subject of a future work.

Acknowledgments

Support by the European network HYKE (EC contract HPRN-CT-2002-00282) is acknowledged.

References

1. X. Antoine and M. Lemou, Multiwavelet approximations of a collision operator in kinetic theory, Preprint.

2. X. Antoine and M. Lemou, Wavelet approximation of a collision operator in kinetic theory, *C. R. Acad. Sci. Ser.* 1, (337), 2003.

3. D. Gines B. Alpert, G. Beylkin and L. Vozovoi, Adaptive solution of partial differential equations in multiwavelet bases, *J. Comput. Phys.*, **182**(2002), no.1, 149-190.

4. Yu. A. Berezin, V. N. Khudick and M. S. Pekker, Conservative finite-difference schemes for the Fokker-Planck equation not violating the law of an increasing entropy, *J. Comput. Phys.*, **69**(1987), no. 1, 163-174.

5. C. Buet and S. Cordier, Conservative and entropy decaying numerical scheme for the isotropic Fokker-Planck-Landau equation, *J. Comput. Phys.*, **145**(1998), no.1, 228-245.

6. C. Buet and S. Cordier, Numerical analysis of the isotropic Fokker-Planck-Landau equation, *J. Comput. Phys.*, **179**(2002), no.1, 43-67.

7. C. Buet, S. Cordier, P. Degond, and M. Lemou, Fast algorithms for numerical, conservative, and entropy approximations of the Fokker-Planck-Landau equation, *J. Comput. Phys.*, **133**(1997), no.2, 310-322.

8. L. Chacón, D. C. Barnes, D. A. Knoll, and G. H. Miley, An implicit energy-conservative 2D Fokker-Planck algorithm, II. Jacobian-free Newton-Krylov solver. *J. Comput. Phys.*, **157**(2000), no.2, 654-682.

9. P. Degond and B. Lucquin-Desreux, An entropy scheme for the Fokker-Planck collision operator of plasma kinetic theory, *Numer. Math.*, **68**(1994), 239-262.

10. E. M. Epperlein, Implicit and conservative difference scheme for the Fokker-Planck equation, *J. Comput. Phys.*, **112**(1994), no.2, 291-297.

11. William Feller, *An Introduction To Probability Theory And Its Applications*, Vol. II, second edition, John Wiley & Sons Inc., New York, 1971.

12. F. Filbet and L. Pareschi, A numerical method for the accurate solution of the Fokker-Planck-Landau equation in the nonhomogeneous case, *J. Comput. Phys.*, **179** (2002), no.1, 1-26.

13. R. Coifman G. Beylkin and V. Rokhlin, Fast wavelet transforms and numerical algorithms. i, *Comm. Pure Appl. Math.*, **44**(1991), no.2, 141-183.

14. A. Girard, C. Lécot, and K. Serebrennikov, Numerical simulation of the plasma of an electron cyclotron resonance ion source, *J. Comput. Phys.*, **191**(2003), no.1, 228-248.

15. R. J. Kingham and A. R. Bell, An implicit Vlasov-Fokker-Planck code to model non-local electron transport in 2-D with magnetic fields, *J. Comput. Phys.*, **194**(2004), no.1, 1-34.

16. M. Lemou, Exact solutions of the fokker-planck equation, *C. R. Acad. Sci. Série* 1, **319**(1994), 579-583.

17. M. Lemou, Multipole expansions for the Fokker-Planck-Landau operator, *Numer. Math.*, **78**(1998), no.4, 597-618.

18. M. Lemou and L. Mieussens, Fast implicit schemes for the Fokker-Planck-Landau equation, *C. R. Acad. Sci. Paris, Ser. I*, **338**(2004), 809-814.

19. M. Lemou and L. Mieussens, Implicit schemes for the Fokker-Planck-Landau equation, *SIAM J. Sci. Comput.*, **27**(2005), no.3, 809-830.

20. E. M. Lifchitz and L. P. Petaevski, *Kinetic Theory*, MIR, Moscow, 1979.

21. L. Pareschi and B. Perthame, A Fourier spectral method for homogeneous Boltzmann equations, In *Proceedings of the Second International Workshop on Nonlinear Kinetic Theories and Mathematical Aspects of Hyperbolic Systems* (Sanremo, 1994), **25**(1996), 369-382.

22. L. Pareschi, G. Russo, and G. Toscani, Fast spectral methods for the Fokker-Planck-Landau collision operator, *J. Comput. Phys.*, **165**(2000), no.1, 216-236.

23. Y. Saad, *Iterative methods for sparse linear systems*, available online at http://www-users.cs.umn.edu/~saad, 2000.

MIP (UMR CNRS 5640), UFR MIG, Université Paul Sabatier, 118 Route de Narbonne, 31062 Toulouse Cedex, France.
E-mail: lemou@mip.ups-tlse.fr

MIP (UMR CNRS 5640), UFR MIG, Université Paul Sabatier, 118 Route de Narbonne, 31062 Toulouse Cedex, France.
E-mail: mieussens@mip.ups-tlse.fr