HIVLASHEA: High order methods for Vlasov-Poisson model for sheath

immediate

1 Introduction

We consider the following model for $(f_e, f_i)(t, x, v) \in \mathbb{R}^+$ and $\phi(t, x) \in \mathbb{R}$, $t \geq 0, x \in [-1, 1], v \in \mathbb{R}$

$$\partial_t f_i + v \partial_x f_i - \partial_x \phi \partial_v f_i = \nu f_e, \quad \text{(Vlasov for ions)}$$

$$\partial_t f_e + v \partial_x f_e + \frac{1}{\mu} \partial_x \phi \partial_v f_e = 0, \quad \text{(Vlasov for electrons)} \tag{1}$$

$$-\lambda^2 \partial_x^2 \phi = \rho_i - \rho_e, \quad \rho_{i,e} = \int_{\mathbb{R}} f_{i,e} dv, \quad \text{(Poisson)}$$

with the following boundary and initial conditions

$$f_{i,e}(t, x = -1, v > 0) = 0, \quad f_{i,e}(t, x = 1, v < 0) = 0,$$
$$-\partial_x \phi(t, 1) = E(t, 1) = C_L(t), \quad -\partial_x \phi(t, -1) = E(t, -1) = C_R(t),$$
$$\phi(t, 0) = 0 \text{ (reference potential)},$$
$$f_{i,e}(0, x, v) = f_{i,e}^0(x, v), \tag{2}$$
$$-\lambda^2 \partial_x^2 \phi(0, x) = \int_{\mathbb{R}} f_i^0(x, v) dv - \int_{\mathbb{R}} f_e^0(x, v) dv,$$

where $f_{i,e}^0$ are given and $C_L(t), C_R(t)$ will be defined in (5) and (6).

We look for solution that are symmetric with respect to $x = 0$:

$$f_{i,e}(t, x, v) = f_{i,e}(t, -x, -v) \ \forall x \in [-1, 1], \ \forall v \in \mathbb{R}, \quad \phi(t, x) = \phi(t, -x), \ \forall x \in [-1, 1].$$

This implies

$$\rho_{i,e}(t, x) = \rho_{i,e}(t, -x), \quad J_{i,e}(t, x) = -J_{i,e}(t, -x),$$
$$E(t, x) = -\partial_x \phi(t, x) = \partial_x \phi(t, -x) = -E(t, -x).$$

We call E the electric field which derives from the electric potential ϕ ($E = -\partial_x \phi$), whereas $\rho_{i,e}$ are the ion and electron densities. Moreover, $J_{i,e} = \int_{\mathbb{R}} v f_{i,e} dv$ denote the ion and electron currents.

Looking for $C(t)$

Here we derive the boundary condition for ϕ.

First, we derive with respect to time the Poisson equation

$$-\partial_t(\lambda^2\partial_x^2\phi) = \partial_t(\rho_i - \rho_e),$$

and considering the difference between the v-integration of the Vlasov equations gives

$$\partial_t(\rho_i - \rho_e) + \partial_x(J_i - J_e) = \nu\rho_e,$$

so that, using $E = -\partial_x\phi$, we get

$$\partial_x(\lambda^2\partial_t E + J_i - J_e) = \nu\rho_e, \quad \forall x \in [-1, 1].$$

Integrating now in space leads to

$$\lambda^2\partial_t E(t,1) + (J_i - J_e)(t,1) = \lambda^2\partial_t E(t,-1) + (J_i - J_e)(t,-1) + \nu\int_{-1}^{1}\rho_e(t,x)dx, \quad (3)$$

and using the symmetries highlighted before ($E(t,x) = -E(t,-x)$ and $J(t,x) = -J(t,-x)$), it comes

$$\lambda^2\partial_t E(t,1) + (J_i - J_e)(t,1) = \frac{\nu}{2}\int_{-1}^{1}\rho_e(t,x)dx, \quad (4)$$

or, in terms of ϕ

$$\partial_x\phi(t,1) = \partial_x\phi(0,1) + \frac{1}{\lambda^2}\int_0^t(J_i - J_e)(s,1)ds - \frac{\nu}{2\lambda^2}\int_0^t\int_{-1}^{1}\rho_e(s,x)dxds \equiv C(t). \quad (5)$$

For the condition $x = -1$, (3) gives

$$\lambda^2\partial_t E(t,-1) = J(t,1) - \frac{\nu}{2}\int_{-1}^{1}\rho_e(t,y)dy. \quad (6)$$

Our goal is to propose high order numerical methods to approximate the solution of (1)-(2)-(5). As a starting point, we will use several techniques proposed and explained in [**badsi**]. Below are some areas of works that will be explored and incorporated in the code developed in [**badsi**]:

- to reach high order at the boundaries, we will use [**boutin**, **coulombel**] (code dedicated to the linear transport to test with the semi-Lagrangian method).

- Hermite-Lagrange 1D interpolation (see Section 4) (code linear transport and then

  - code for linear transport (semi-Lagrangian method and periodic boundary conditions)
  - code for Vlasov-Poisson (one species and periodic boundary conditions)
  - code for Vlasov-Poisson (two species and periodic boundary conditions: test on exact solution from [**morel**])

2

- write a solver for the Poisson equation with Dirichlet-Neuman boundary conditions.

- investigate high order time splittings for (1)-(2)-(5).

But also

- Poisson reformulation following ideas in [**degond**] to overcome the stringent condition on the time step.

- construction of the stationary solutions of (1)-(2)-(5).

Energy evolution

Multiplying by $v^2/2$ the ion equation and by $\mu v^2/2$ the electron equation gives, after integration in v and x

$$\frac{d}{dt}\mathcal{E}_{\text{kin}} + \int\int_{-1}^{1} \frac{v^3}{2}\partial_x(f_i + \mu f_e)dxdv - \int_{-1}^{1} E(J_i - J_e)dx = \frac{\nu}{\mu}\mathcal{E}_{\text{kin}}^e,$$

with $\mathcal{E}_{\text{kin}} = \int\int_{-1}^{1} \frac{v^2}{2}(f_i + \mu f_e)dxdv$, $\mathcal{E}_{\text{kin}}^e = \int\int_{-1}^{1} \frac{v^2}{2}\mu f_e dxdv$ and $J_{i,e} = \int v f_{i,e}dv$. But the second term is

$$\int\int_{-1}^{1} \frac{v^3}{2}\partial_x(f_i + \mu f_e)dxdv = M_3(t,1) - M_3(t,-1),$$

with $M_3(t,x) := \int \frac{v^3}{2}(f_i(t,x,v) + \mu f_e(t,x,v))dv$. Using the Ampère equation

$$J(t,x) - J(t,-1) = -\lambda^2\partial_t E(t,x) + \lambda^2\partial_t E(t,-1) + \nu\int_{-1}^{x}\rho_e(t,y)dy,$$

we get for the third term

$$\begin{aligned}
\int_{-1}^{1} EJdx &= \int_{-1}^{1} Edx\left[J(t,-1) + \lambda^2\partial_t E(t,-1)\right] - \lambda^2\int_{-1}^{1} E\partial_t E(t,x)dx + \nu\int_{-1}^{1} E\int_{-1}^{x}\rho_e(t,y)dydx \\
&= -\frac{\lambda^2}{2}\frac{d}{dt}\int_{-1}^{1} E^2 dx - \nu\int_{-1}^{1}(\partial_x\phi)\int_{-1}^{x}\rho_e(t,y)dydx \\
&= -\frac{\lambda^2}{2}\frac{d}{dt}\int_{-1}^{1} E^2 dx - \nu\left[\phi(t,1)\int_{-1}^{1}\rho_e(t,y)dy\right] + \nu\int_{-1}^{1}\phi(t,x)\rho_e(t,x)dx,
\end{aligned}$$

since $\int_{-1}^{1} Edx = -(\phi(t,1) - \phi(t,-1)) = 0$. Hence, we get

$$\frac{d}{dt}\mathcal{E}_{\text{kin}} + \frac{d}{dt}\mathcal{E}_{\text{pot}} + M_3(t,1) - M_3(t,-1) + \nu\phi(t,1)\int_{-1}^{1}\rho_e(t,x)dx$$

$$= \nu\int_{-1}^{1}\phi(t,x)\rho_e(t,x)dx + \frac{\nu}{\mu}\mathcal{E}_{\text{kin}}^e,$$

with $\mathcal{E}_{\text{pot}} = \frac{\lambda^2}{2}\int_{-1}^{1} E^2 dx$.

3

Mass evolution

We integrate in x, v the ion and electron equations. Denoting $m_{i,e}(t) = \int \int_{-1}^{1} f_{i,e} dx dv$ leads to

$$m_i' \quad + \quad 2J_i(t,1) = \nu m_e$$
$$m_e' \quad + \quad 2J_e(t,1) = 0.$$

Thus the difference between the 2 above equations gives $\frac{d}{dt}(m_i - m_e) + 2J(t,1) = \nu m_e$. Using the expression of $J(t,1)$ from (4), we finally obtain

$$\frac{d}{dt}(m_i - m_e) - 2\lambda^2 \partial_t E(t,1) + \nu m_e = \nu m_e,$$

or

$$\frac{d}{dt}(m_i - m_e - 2\lambda^2 E(t,1)) = 0.$$

2 Examples

2.1 Test 1d

We can consider the following 1d test

$$\partial_t + a\partial_x u = 0, \quad x \in [0,1], \quad a > 0,$$

with the following boundary and initial conditions $u(t = 0, x) = u_0(x), u(t,0) = u_L(t)$. We introduce a mesh in space $x_i = i\Delta x, i = 0, \ldots, N$, with $\Delta x > 0$ and $N\Delta x = 1$ and in time $t^n = n\Delta t, \Delta t > 0$. To reconstruct using high order interpolation, one needs to give a sense to the numerical solution at the inflow ghost points $x_{-i}, i = 1, 2, \ldots$ at the outflow ghost points $x_{N+i}, i = 1, 2, \ldots$.

For the inflow boundary, one can investigate two options

- characteristics based: using the characteristics, we have $u(t^n, x_{-1}) = u(t, x_{-1} + a(t - t^n)) = u_L(t)$ with t the time such that $x_{-1} + a(t - t^n) = 0$, or $t = t^n - x_{-1}/a$. Of course, the same strategy can be applied for $x_{-i}, i = 2, 3, \ldots$.

- Inverse Lax-Wendroff (ILW): following the strategy developed in [**boutin**, **coulombel**], we perform a Taylor expansion of inflow ghost points at the boundary $x = 0$

$$u_{-1}^n \approx \sum_{k=0}^{K} \frac{(-\Delta x)^k}{k!} (\partial_x^k u^n)_0 = \sum_{k=0}^{K} \frac{(\Delta x)^k}{a^k k!} (\partial_t^k u^n)_0 = \sum_{k=0}^{K} \frac{(\Delta x)^k}{a^k k!} (\partial_t^k u_L(t^n)).$$

Of course, the same strategy can be applied for $x_{-i}, i = 2, 3, \ldots$.

One can observe and check properly that the ILW is an approximation of the characteristics based approach. The two options can be programmed in combination with a high order semi-Lagrangian method.

For the outflow, one can use the strategy proposed in [**boutin**, **coulombel**] where a condition is imposed to reconstruct $u_{N+i}, i = 1, 2, \ldots$. For a k_b given, they introduce

the finite difference operator $(Du)_j = u_j - u_{j-1}$. We impose $(D^2u)_N = 0$ which means $u_{N-1} - 2u_N + u_{N+1} = 0$ or $u_{N+1} = 2u_N - u_{N-1}$. The next point u_{N+2} is then defined with $(D^2u)_{N+1} = u_N - 2u_{N+1} + u_{N+2} = 0$. Another choice would be $(D^3u)_N = 0$ which imposes u_{N+1} and so on.

Tests can be performed using the exact solution $u(t, x) = \sin(k(x - at))$ which means that the initial condition is $u_0(x) = \sin(kx)$ and the boundary condition is $u_L(t) = -\sin(kat)$ for a given $k \in \mathbb{N}^\star$.

2.2 Test 2d

We suggest to investigate the properties of the above approach in a 2d context using the following model

$$\partial_t u + v\partial_x u - x\partial_v u = 0, \quad x \in [-1, 1], \quad v \in [-1, 1],$$

with the initial condition $u(t = 0, x, v) = u_0(x, v)$ and with the boundary conditions in space

$$u(t, -1, v) = u_L(t, v), v > 0, \quad u(t, 1, v) = u_R(t, v), v < 0,$$

whereas in v, vanishing boundary conditions are considered.

We shall consider the following exact solution

$$u(t, x, v) = u_0(x \cos t - v \sin t, x \sin t + v \cos t),$$

so that the boundary conditions are $u_L(t) = u_0(-\cos t - v \sin t, -x \sin t + v \cos t)$ for $v > 0$ and $u_R(t) = u_0(\cos t - v \sin t, \sin t + v \cos t)$ for $v < 0$. The initial condition can be chosen as $u_0(x, v) = \exp(-100(x - 1)^2 - 100v^2)$.

To use a directional splitting method, one has to modify the boundary conditions to be compatible with the one dimensional space advections.

- solve $\partial_t u + v\partial_x u = 0$ on $\Delta t/2$ with $\tilde{u}_L(t^n) = u_0(x_{-1} \cos t^n - v \sin t^n, x_{-1} \sin t^n + v \cos t^n)$

- solve $\partial_t u - x\partial_v u = 0$ on Δt with zero boundary conditions at the boundary

- solve $\partial_t u + v\partial_x u = 0$ on $\Delta t/2$ with $\tilde{u}_L(t) = u_0(x_{-1} \cos t - v \sin t, x_{-1} \sin t + v \cos t)$ with t such that $v(t - t^{n+1}) = -\Delta x$.

2.3 Poisson solver

To solve the Poisson equation

$$\lambda^2 \partial_x E = \rho, \quad x \in [-1, 1],$$

with $E(1)$ given, one integrates successively between x and 1 and then between -1 and x to get

$$E(1) - E(x) = \frac{1}{\lambda^2} \int_x^1 \rho(y) dy, \quad E(x) - E(-1) = \frac{1}{\lambda^2} \int_{-1}^x \rho(y) dy.$$

We obtain the following expression for $E(x)$

$$E(x) = \frac{1}{2}(E(1) + E(-1)) + \frac{1}{2\lambda^2}\Big[\int_{-1}^{x}\rho(y)dy - \int_{x}^{1}\rho(y)dy\Big].$$

We assume that ρ is known on the spatial grid $x_i = i\Delta x$ so that $\rho(x_i) \approx \rho_i$. Hence, an approximation of the electric field at x_i is given by

$$E_i = \frac{1}{2}(E(1) + E(-1)) + \frac{\Delta x}{4\lambda^2}\Big[\sum_{j=0}^{i-1}(\rho_j + \rho_{j+1}) - \sum_{j=i}^{N-1}(\rho_j + \rho_{j+1})\Big].$$

This expression ensures the symmetry of E at the discrete level.

3   Codes

For testing purposes, codes are developed and validated on examples described below.

3.1   Two-species Vlasov-Poisson with periodic boundary conditions

A code solving the two-species Vlasov-Poisson with periodic boundary conditions is available in Julia and Python. In particular, exact stationary solutions are tested. Ask to Pierre Navaro.

3.2   Two-species Vlasov-Poisson with boundary conditions: version 1

The Mehdi's code solves the full model (1)-(2) with an upwind scheme + explicit Euler time integrator + Poisson solver described in 2.3. It gives a reference numerical solution for the sheath development. The initial condition is

$$f_i^0(x,v) = \frac{1}{\sqrt{2\pi}}e^{-v^2/2}, \quad f_e^0(x,v) = \frac{\sqrt{\mu}}{\sqrt{2\pi}}e^{-\mu v^2/2}, \quad \phi(0,x) = E(0,x) = 0.$$

3.3   Two-species Vlasov-Poisson with boundary conditions: version 2

The goal would be to integrate the numerical methods developed and validated for the problems of Section 2 in the Michel's code.

4   Hermite interpolation

Let us consider the 3rd order interpolation on the interval $[x_i, x_{i+1}]$ such that $x_{i+1} - x_i = h > 0$. We look for the polynomial function $P \in \mathbb{R}^2[X]$ such that

$$P(x_i) = f(x_i), P(x_{i+1}) = f(x_{i+1}), P'(x_i) = f'(x_i), P'(x_{i+1}) = f'(x_{i+1}). \tag{7}$$

On the one side, we define the two 1st order Lagrange polynomial

$$L_i(x) = \frac{x - x_{i+1}}{x_i - x_{i+1}} = \frac{x_{i+1} - x}{h} \text{ and } L_{i+1}(x) = \frac{x - x_i}{x_{i+1} - x_i} = \frac{x - x_i}{h},$$

and following the Peyré's tweet, the Hermite polynomial satisfying (7) is

$$P(x) = f(x_i)H_i(x) + f(x_{i+1})H_{i+1}(x) + f'(x_i)K_i(x) + f'(x_{i+1})K_{i+1}(x), \tag{8}$$

with

$$K_i(x) = L_i(x)^2(x - x_i), \quad K_{i+1}(x) = L_{i+1}(x)^2(x - x_{i+1}), \tag{9}$$
$$H_i(x) = L_i(x)^2(1 - 2L_i'(x_i)(x - x_i)), \quad H_{i+1}(x) = L_{i+1}(x)^2(1 - 2L_{i+1}'(x_{i+1})(x - x_i)) \tag{10}$$

From the properties of H_i, K_i, we can check that (8) satisfies (7).

We can generalize to arbitrary (odd) order in the following way. From a Lagrange polynomial of degree $(2d + 1)$, we can reconstruct using Peyré's tweet a polynomial of degree $2(2d + 1) + 1$. In this method, we need to reconstruct $f'(x_i)$ which is not known since we only know $f(x_i)$. To do so, we can use finite difference formula with the correct order (to not destroy the order of the Hermite reconstruction).

The strategy is the following. First we define the Lagrange polynomial of order $2d+1$ ($d \in \mathbb{N}$ given)

$$L_i(x) = \Pi_{j \neq i} \frac{x - x_j}{x_i - x_j}, \quad \text{for } i = -d, \dots, d + 1.$$

We also define the derivative of the Lagrange polynomial. We only need $L_i'(x_i)$ for which we have

$$L_i'(i) = \sum_{j=-d,\dots,d+1/j \neq i} \frac{1}{i - j}.$$

Then, we define $(K_i)_{i=-d,\dots,d+1}$ and $(H_i)_{i=-d,\dots,d+1}$ from (9) and we have the following polynomial

$$P(x) = \sum_{i=-d}^{d} f(x_i)H_i(x) + \sum_{i=-d}^{d} f'(x_i)K_i(x).$$

An approximation of $f'(x_i)$ is required. Different options can be envisaged according to stability. The first one is the uncentered one as in [caim]

$$P(x) = \sum_{i=-d}^{d} f(x_i)H_i(x) + \sum_{i=-d}^{0} f'_{i,+}K_i(x) + \sum_{i=1}^{d} f'_{i,-}K_i(x).$$

where the derivatives are approximated with

$$f'(x_i) \approx f'_{i,+} = \sum_{\ell=r^+}^{s^+} b_\ell^+ f_{i+\ell} \text{ for } i = -d, \dots, 0,$$

$$f'(x_i) \approx f'_{1,-} = \sum_{\ell=r^-}^{s^-} b_\ell^- f_{i+\ell} \text{ for } i = 1, \dots, d.$$

The integers $r^{\pm}, s^{\pm}$ are computed according to a degree $p \in \mathbb{N}^{\star}$: $r^{+} = -\left\lfloor \dfrac{p}{2} \right\rfloor$, $s^{+} = \left\lfloor \dfrac{p+1}{2} \right\rfloor$, $r^{-} = -s^{+}$, $s^{-} = -r^{+}$. Below are the coefficients for different values of p (general formulas can be found in [**caim**]

$$
\begin{aligned}
p &= 1 & b^{+} &= (-1, 1) \\
p &= 2 & b^{+} &= (-1/2, 0, 1/2) \\
p &= 3 & b^{+} &= (-1/3, -1/2, 1, -1/6) \\
p &= 4 & b^{+} &= (1/12, -2/3, 0, 2/3, -1/12) \\
p &= 5 & b^{+} &= (1/20, -1/2, -1/3, 1, -1/4, 1/30) \\
p &= 6 & b^{+} &= (-1/60, 3/20, -3/4, 0, 3/4, -3/20, 1/60),
\end{aligned}
$$

with $b^{-}_{-\ell} = -b^{+}_{\ell}$ for $\ell = r^{+}, \ldots, s^{+}, \ell \neq 0$ and $b^{-}_{0} = -b^{+}_{0}$.

More details of the method in the maple code of Michel !

```
d:=1:r:=-d:s:=d+1:
dmax:=2*(s-r+1):
for k from r to s do
        \% Lagrange polynomials of odd degree
        L[k]:=product((x-j)/(k-j),j=r..k-1)*product((x-j)/(k-j),j=k+1..s):
        \% derivative of Lagrange
        dL[k]:=diff(L[k],x):
od:
for k from r to s do
        \% Compute H and K from the Peyr\'e formulas
        H[k]:=L[k]**2*(1-2*subs(x=k,dL[k])*(x-k)):#f
        K[k]:=L[k]**2*(x-k):   #fprime
od:
for k from 0 to dmax do
        \%test on a given function
        f:=x**k:df:=diff(f,x):
        g:=simplify(x**k-add(H[j]*subs(x=j,f),j=r..s)-add(K[j]*subs(x=j,df),j=r..s)):
        print(k,g);
od:
```

References

[1] M. BADSI, M. MEHRENBERGER, L. NAVORET *Numerical stability of plasma sheath*, ESAIM: Proceedings and Surveys, SMAI 2017, Volume 64 (2018), pp 17-36.

[2] B. BOUTIN, T. H. T. NGUYEN, ABRAHAM SYLLA, S. TRAN-TIEN, J.-F. COULOMBEL, *High order numerical schemes for transport equations on bounded domains*, (2021), ESAIM: Proceedings and Surveys.

[3] J.-F. Coulombel, F. Lagoutière. *The neumann numerical boundary condition for transport equations*. Kinet. Relat. Models, 2020, 13(1), pp. 1-32.

[4] P. Degond, F. Deluzet, L. Navoret, A.-B. Sun, M.-H. Vignal, *Asymptotic-Preserving Particle-In-Cell method for the Vlasov-Poisson system near quasi-neutrality*, J. Comp. Phys., vol. 229, 16, pp. 5630-5652 (2010)

[5] A. Hamiaz, M. Mehrenberger, H. Sellama, E. Sonnendrucker, *The semi-Lagrangian method on curvilinear grids*, Communications in Applied and Industrial Mathematics, 2016, Volume 7, Issue 3, pp. 96-134.

[6] C. Morel, *A well-balanced scheme using exact solutions to the two species Vlasov-Poisson system*, Intern notes.

[7] E. Sonnendrucker *Numerical Methods for the Vlasov-Maxwell equations*.