Monte-Carlo/PIC methods to solve Vlasov-Boltzmann equation

David Pfefferlé\textsuperscript{1}

\textsuperscript{1}The University of Western Australia, 35 Stirling Highway, Crawley WA 6009, Australia

Mini-course/workshop on the application of computational mathematics to plasma physics

June 24-27, 2019 - Canberra, Australia
How to model a plasma?
A plasma is a globally neutral soup of charged ions and electrons.

Brute-force 2D model:
- take $N$ electrons and $N$ ions
- uniform distribution in space
- Gaussian distribution in velocity with variance $v_{th} = \sqrt{T/m}$
- Coulomb force over all pairs

$$\dot{x}_i = v_i$$

$$\dot{v}_i = \sum_{j \neq i} \frac{e_i e_j (x_i - x_j)}{m_i |x_i - x_j|^3}$$

- $2N \times 4D$ system of ODEs
Brute-force model of plasma in a box

plasma box \([0, 1]^2\) with 20 ions and 20 electrons, \(m_i/m_e = 5\), \(T = 20\), mirror boundary conditions.
Issues with brute-force model are serious

- stiffness due to singular Coulomb force
- numerical accuracy and stability (poor energy conservation, crashes)
- improvement at larger temperature, lower Coulomb coupling
- computational time grows like $N^2$ at best, more like $N^3$ (accuracy)
- real plasma has $N \sim 10^{16} \Rightarrow$ intractable even on largest clusters
- boundary conditions? realistic mass ratios? ...

We need a reduced model ⇒ Vlasov equation [Balescu, 1988]

- most of the time, particles are free-flying (especially at high temperature)
- Coulomb force yields small deflections (except rare face-on events)
- with $N \sim 10^{16} \Rightarrow$ mean field theory should work
Liouville equation for phase-space density

- system of ODEs is a flow on $6 \times N \sim 10^{16}$ dimensional manifold
- as a Hamiltonian system (reversible dynamics), phase-space volume
  $\prod_{i=1}^{N} dx_i dv_i$ is conserved
- Liouville equation holds for PDF $F(x_1, v_1, \ldots, x_N, v_N, t)$

\[
\frac{dF}{dt} = \partial_t F + \sum_{i=1}^{N} \left( \dot{x}_i \cdot \partial_{x_i} F + \dot{v}_i \cdot \partial_{v_i} F \right) = 0
\]

where $\dot{x}_i = v_i$ and $\dot{v}_i = \sum_{j=1}^{N} a_{ij} = \frac{1}{m_i} \partial_{x_i} \sum_{j=1}^{N} \frac{e_i e_j}{\left| x_i - x_j \right|} V_{ij}, \text{ Coulomb}$

- notice $a_{ij} = -a_{ji}$
Statistical average of microscopic events

- averaging over $1, 2, \ldots, N - 2, N - 1$ particles: collection of reduced distribution functions

\[
f^{\alpha\beta}(x, v, x_2, v_2, t) = N_{\alpha}N_{\beta} \int F(x, v, x_2, v_2, \ldots, x_N, v_N, t) dx_3 dv_3 \ldots
\]

\[
f^{\alpha}(x, v, t) = N_{\alpha} \int F(x, v, x_2, v_2, \ldots, x_N, v_N, t) dx_2 dv_2 \ldots dx
\]

- averaging Liouville equation yields BBGKY hierarchy

\[
\partial_t f^{\alpha} + v \cdot \partial_x f^{\alpha} + \sum_{\beta} \int d x_2 d v_2 \, a_{12}^{\alpha\beta} \cdot \partial_v f^{\alpha\beta}(x, v, x_2, v_2, t) = 0
\]
Vlasov via Markovian assumption

\[ f^{\alpha\beta}(x, v, x_2, v_2, t) = f^{\alpha}(x, v, t) f^{\beta}(x_2, v_2, t) + g^{\alpha\beta}(x, v, x_2, v_2, t) \]

product (independence) and “binary correlation function” \( g^{\alpha\beta} \)
similarly, \( g^{\alpha\beta\gamma} \) “three-body correlations”, \( g^{\alpha\beta\gamma\delta} \) “four-body”, . . .

Closure

1. retain only “binary correlations” (truncate hierarchy)
2. Markovian assumption
   \[ f^{\alpha\beta} \sim f^{\alpha} f^{\beta} \text{ and } g^{\alpha\beta} = C[f^{\alpha}, f^{\beta}] \Rightarrow \text{irreversibility} \]

Vlasov-Boltzmann equation

\[ \frac{df^{\alpha}}{dt} = \partial_t f^{\alpha}(x, v, t) + v \cdot \partial_x f^{\alpha} + \frac{e^{\alpha}}{m^{\alpha}} E \cdot \partial_v f^{\alpha} = \sum_\beta C[f^{\alpha}, f^{\beta}] \]

where \( E \) is the mean electric field
What is the mean electric field?

Recall \( f^{\alpha\beta} = f^{\alpha} f^{\beta} \), \( a^{\alpha\beta}_{12} = \frac{1}{m_\alpha} \partial_x \frac{e_\alpha e_\beta}{|x - x_\beta|} \),

\[
\sum_\beta \int d\mathbf{x}_2 d\mathbf{v}_2 \ a^{\alpha\beta}_{12} \cdot \partial_v f^\alpha(x, v, t) f^\beta(x_2, v_2, t) = \frac{e_\alpha}{m_\alpha} \left( \partial_x \int d\mathbf{x}_2 \frac{\rho(x_2, t)}{|x - x_2|} \right) \cdot \partial_v f^\alpha
\]

where \( \rho(x, t) = \sum_\beta e_\beta n_\beta(x, t) \) is the charge density
and \( n_\beta(x, t) = \int d\mathbf{v} f^\beta(x, \mathbf{v}, t) \) is the particle density of species \( \beta \)

- \( \frac{-1}{4\pi|x_2 - x|} \) is the Green’s function of the Laplacian

\[
\Phi(x, t) = \int d\mathbf{x}_2 \frac{\rho(x_2, t)}{|x - x_2|} \iff \nabla^2 \Phi = -4\pi \rho
\]

- mean electric field \( \mathbf{E} = -\nabla \Phi \), \( \nabla \cdot \mathbf{E} = 4\pi \rho \)
  (is curl-free \( \nabla \times \mathbf{E} = 0 \))
Vlasov-Poisson model

\[
\begin{align*}
\partial_t f_i(x, v, t) + v \cdot \partial_x f_i + \frac{e}{m_i} E \cdot \partial_v f_i &= C[f_i, f_i] + C[f_i, f_e] \\
\partial_t f_e(x, v, t) + v \cdot \partial_x f_e - \frac{e}{m_e} E \cdot \partial_v f_e &= C[f_e, f_e] - C[f_i, f_e] \\
E &= -\nabla \Phi, \quad \nabla^2 \Phi = -4\pi e (n_i - n_e) \\
n_{i,e}(x, t) &= \int d\mathbf{v} f_{i,e}(x, \mathbf{v}, t), \quad \int d\mathbf{x} (n_i - n_e) = 0
\end{align*}
\]
Approximate reduced (single) particle distribution function via “macro-particles”

\[ f(x, v, t) = \sum_{l=1}^{M} w_l \delta(x - x_l(t)) \delta(v - v_l(t)) \]

where \( \dot{x}_l = v_l, \dot{v}_l = \frac{e_l}{m_l} E(x_l(t), t) \) and \( w_l \) is the “weight” (phase-space volume)

1. “particle push”: advect \( f(x, v, t) \) via Lagrangian method
   - Hamiltonian flow to ensure conservation laws
   - no mesh in velocity
   - drawback: noise \( \sim 1/\sqrt{N} \) requires lots of markers

2. “field solve”: deposit “charge” on Eulerian mesh and solve \( E \)
   - standard methods (finite difference, FEM, spectral)
Particle push using leap-frog scheme

\[ \begin{align*}
\dot{x} &= v \\
\dot{v} &= \frac{e}{m} E(x, t)
\end{align*} \]  \Rightarrow

\[ \begin{align*}
\mathbf{v}_{n+1/2} &= \mathbf{v}_n + \frac{dt}{2} \frac{q}{m} \mathbf{E}(\mathbf{x}_n, t_n) \\
\mathbf{x}_{n+1} &= \mathbf{x}_n + dt \mathbf{v}_{n+1/2} \\
\mathbf{v}_{n+1} &= \mathbf{v}_{n+1/2} + \frac{dt}{2} \frac{q}{m} \mathbf{E}(\mathbf{x}_{n+1}, t_n)
\end{align*} \]

- 2nd order explicit symplectic integrator
- simple, fast, robust

Python code:

```python
y[:,1]+=0.5*dt*q/m*efield.eval_field(y[:,0])
y[:,0]+=dt*y[:,1]
y[:,1]+=0.5*dt*q/m*efield.eval_field(y[:,0])
```

D. Pfefferlé (UWA)  
github.com/viper2642/pic-vlasov-plasma  
ANU/MSI mini-course  
12 / 16
Particle-in-Cell charge deposition

\[ x - x_i \]

\[ \Delta x \]

\[ q, w \]

\[ x_i \]

\[ x_{i+1} \]
In 1D

\[ \nabla^2 \Phi = \rho \Rightarrow \Phi''(x) = \rho(x) \]

Finite difference discretisation

\[ \frac{\Phi_{i+1} - 2\Phi_i + \Phi_{i-1}}{(\Delta x)^2} = \rho_i \]

Dirichlet boundary conditions

\[
\begin{pmatrix}
-2 & 1 & 0 & \cdots & 0 \\
1 & -2 & 1 & \ddots & \vdots \\
0 & \ddots & \ddots & \ddots & 0 \\
\vdots & \ddots & \ddots & -2 & 1 \\
0 & \cdots & 0 & 1 & -2
\end{pmatrix}
\begin{pmatrix}
\Phi_1 \\
\vdots \\
\Phi_{N-1}
\end{pmatrix}
=
\begin{pmatrix}
(\Delta x)^2 \rho_1 - \Phi_0 \\
(\Delta x)^2 \rho_2 \\
\vdots \\
(\Delta x)^2 \rho_{N-2} \\
(\Delta x)^2 \rho_{N-1} - \Phi_N
\end{pmatrix}
\]
Thomas algorithm for tridiagonal matrix

Our system is

\[
\begin{pmatrix}
g & u & 0 \\
l & \ddots & \ddots \\
0 & \ddots & \ddots \\
0 & \ddots & \ddots \\
\end{pmatrix} \Phi = \rho,
\]

where diagonal $N-1$-vector $g = -2$, upper and lower diagonal $N-2$-vectors $u = l = 1$

Thomas algorithm is a clever Gauss-Jordan elimination:

1. **forward upper sweep** with $u'_1 = u_1/g_1$ and

   \[
   u'_i = \frac{u_i}{g_i - l_i u'_{i-1}}, \quad i = 2, 3, \ldots, N - 2
   \]

2. **forward sweep** on rhs with $\rho'_1 = \rho_1/g_1$ and

   \[
   \rho'_i = \frac{\rho_i - l_i \rho'_{i-1}}{g_i - l_i u'_{i-1}}, \quad i = 2, 3, \ldots, N - 2
   \]

3. **backward propagation** $\Phi_{N-1} = \rho'_{N-1}$

   \[
   \Phi_i = \rho'_i - u'_i \Phi_{i+1}, \quad i = N - 2, N - 3, \ldots, 1
   \]