An exponential integrator for highly oscillatory Vlasov-Poisson systems

Sever Hirstoaga

Inria Nancy-Grand Est & Université de Strasbourg, FRANCE

Joint work with Emmanuel Frénod, Mathieu Lutz and Eric Sonnendrücker

NumKin 2013

Garching, September 5, 2013
OUTLINE

1. Equations of interest - Motivations
2. Stiff ODEs
3. Numerical schemes and the new algorithm
4. Numerical results
5. Conclusion
Paraxial approximation

For $\varepsilon \to 0$ solve

\[
\begin{aligned}
\frac{\partial f^\varepsilon}{\partial t} + \frac{v}{\varepsilon} \frac{\partial f^\varepsilon}{\partial r} + \left( E^\varepsilon - \frac{r}{\varepsilon} \right) \frac{\partial f^\varepsilon}{\partial v} &= 0, \\
\frac{1}{r} \frac{\partial (r E^\varepsilon)}{\partial r} &= \int f^\varepsilon(t, r, v) \, dv \\
f^\varepsilon(t = 0, r, v) &= f_0(r, v).
\end{aligned}
\]

(1)

where

- $f^\varepsilon = f^\varepsilon(t, r, v)$ particles distribution function
- Time $t \in [0, T]$, Position $r > 0$, Velocity $v \in \mathbb{R}$
- $r \mapsto r/\varepsilon$ focusing external electric field
- $E^\varepsilon(t, r)$ self-consistent electric field
Drift-kinetic regime

For $\varepsilon \to 0$ solve

$$
\begin{cases}
\partial_t f^\varepsilon + \mathbf{v} \cdot \nabla_x f^\varepsilon + \left( E^\varepsilon + \frac{1}{\varepsilon} \mathbf{v}^\perp \right) \cdot \nabla_v f^\varepsilon = 0,

E^\varepsilon (x, t) = -\nabla_x \phi^\varepsilon, \quad -\Delta_x \phi^\varepsilon = \int_{\mathbb{R}^2} f^\varepsilon d\mathbf{v} - n_i,

f^\varepsilon (x, \mathbf{v}, t = 0) = f_0 (x, \mathbf{v}),
\end{cases}
$$

(2)

where

- $f^\varepsilon = f^\varepsilon (t, x, \mathbf{v})$ Particles distribution function
- Position $x = (x_1, x_2)$, Velocity $\mathbf{v} = (\nu_1, \nu_2)$, and $\mathbf{v}^\perp = (-\nu_2, \nu_1)$

- **Strong** and constant magnetic field in the $x_3$ direction
- $E^\varepsilon(x, t)$ evolves in the plane $\perp$ to the magnetic field.
Particle-In-Cell method

Dirac sum approximation for $f^\varepsilon$:

$$f^\varepsilon_{N_p}(t, r, v) = \sum_{k=1}^{N_p} \omega_k \delta(r - R_k(t)) \delta(v - V_k(t))$$

where $N_p$ is the number of **macroparticles** and $(R_k(t), V_k(t))$ is the macroparticle $k$ moving along a characteristic curve of Vlasov eq.

$$R'(t) = \frac{1}{\varepsilon} V(t), \quad R(0) = r_0$$

$$V'(t) = -\frac{1}{\varepsilon} R(t) + E(t, R(t)), \quad V(0) = v_0$$

The same thing for $(X_k(t), V_k(t))$

$$X'(t) = V(t), \quad X(0) = x_0$$

$$V'(t) = \frac{1}{\varepsilon} V^\perp(t) + E^\varepsilon(t, X(t)), \quad V(0) = v_0$$
Highly oscillatory solutions

- When $E \equiv 0$, the solution is
  
  $\begin{pmatrix} R(t) \\ V(t) \end{pmatrix} = \mathcal{R} \left( \frac{t}{\varepsilon} \right) \begin{pmatrix} r_0 \\ v_0 \end{pmatrix}$

  where $\mathcal{R}(\tau) = \begin{pmatrix} \cos \tau & \sin \tau \\ -\sin \tau & \cos \tau \end{pmatrix}$.

- When $E \equiv 0$, the solution is
  
  $\begin{align*}
  X(t) &= x_0 + \varepsilon v_0^\perp - \varepsilon \mathcal{R} \left( \frac{t}{\varepsilon} \right) v_0^\perp \\
  V(t) &= \mathcal{R} \left( \frac{t}{\varepsilon} \right) v_0
  \end{align*}$

  $x_0 + \varepsilon v_0^\perp$ is the guiding center.

- When the electric field not zero $\Rightarrow$ stiff solutions (i.e. evolving on two disparate time scales)
Homogenization – The two-scale limit - First model


As $\varepsilon \to 0$, $f^\varepsilon$ two-scale converges to $F$, i.e. $f^\varepsilon(t, r, v) \sim F(t, \frac{t}{\varepsilon}, r, v)$

where $\frac{\partial F}{\partial \tau} + v \frac{\partial F}{\partial r} - r \frac{\partial F}{\partial v} = 0$, meaning that

$$F(t, \tau, r, v) = G(t, R^\tau(r, v)),$$

where $R^\tau$ is a rotation in $\mathbb{R}^2$ and $G = G(t, q, u)$ is the solution to

$$\begin{aligned}
\frac{\partial G}{\partial t}(t, q, u) + \frac{1}{2\pi} \int_0^{2\pi} R^\tau(0, E^0(t, \tau, R_r^{-\tau}(q, u))) d\tau \cdot \nabla_{q,u} G(t, q, u) = 0 \\
G(t = 0, q, u) = \frac{1}{2\pi} f_0(q, u) \quad \text{and} \quad \frac{1}{r} \frac{\partial (r E^0)}{\partial r} = \int G(t, R^\tau(r, v)) dv
\end{aligned}$$

Gain: larger $\Delta t$ may be used in a numerical scheme for $G$. 

Sever Hirstoaga – An exponential integrator for highly oscillatory Vlasov-Poisson systems
As $\varepsilon \to 0$, $f^\varepsilon$ two-scale converges to $F$, i.e. 

$$f^\varepsilon(t, r, v) \sim F\left(t, \frac{t}{\varepsilon}, r, v\right)$$

where

$$\frac{\partial F}{\partial \tau} + v \frac{\partial F}{\partial r} - r \frac{\partial F}{\partial v} = 0,$$

meaning that

$$F\left(t, \tau, r, v\right) = G\left(t, \mathcal{R}^\tau (r, v)\right),$$

where $\mathcal{R}^\tau$ is a rotation in $\mathbb{R}^2$ and $G = G(t, q, u)$ is the solution to

$$\begin{cases}
\frac{\partial G}{\partial t}(t, q, u) + \frac{1}{2\pi} \int_0^{2\pi} \mathcal{R}^\tau (0, E^0(t, \tau, \mathcal{R}^{-\tau}_r (q, u))) d\tau \cdot \nabla_{q,u} G(t, q, u) = 0 \\
G(t = 0, q, u) = \frac{1}{2\pi} f_0(q, u) \quad \text{and} \quad \frac{1}{r} \frac{\partial (r E^0)}{\partial r} = \int G\left(t, \mathcal{R}^\tau (r, v)\right) dv
\end{cases}$$

**Gain:** larger $\Delta t$ may be used in a numerical scheme for $G$. 

**Sever Hirstoaga**

An exponential integrator for highly oscillatory Vlasov-Poisson systems

As \( \varepsilon \to 0 \), \( f^\varepsilon \) two-scale converges to \( F \), where \( \frac{\partial F}{\partial \tau} + v^\perp \cdot \nabla_v F = 0 \) i.e.

\[
F(t, \tau, x, v) = G\left(t, x, R^\tau(v)\right),
\]

where \( G = G(t, x, u) \) is the solution to

\[
\begin{cases}
\frac{\partial G}{\partial t} = 0 \\
G(0, x, u) = \frac{1}{2\pi} f_0(x, u)
\end{cases}
\]

and some limit Poisson equation.
Aim

1. Perform simulation of the models (Vlasov-Poisson for $f^\varepsilon$) with large time steps with respect to the oscillation ($2\pi\varepsilon$).
2. The scheme to be uniformly accurate when $\varepsilon$ goes to zero.

General Problem: Solve stiff ODEs where stiffness arises from the linear term

$$y'(t) = \frac{1}{\varepsilon}Ly(t) + F(t, y(t)).$$

Difficulties: We look for a numerical scheme to be stable and accurate for any initial condition and during both phases (fast and slow).

Drawbacks:
- explicit methods need very small time step (for stability)
- fully implicit methods are costly (slow).
Aim

1. Perform simulation of the models (Vlasov-Poisson for $f^\varepsilon$) with large time steps with respect to the oscillation ($2\pi\varepsilon$).
2. The scheme to be uniformly accurate when $\varepsilon$ goes to zero.

General Problem: Solve **stiff** ODEs where stiffness arises from the linear term

$$y'(t) = \frac{1}{\varepsilon}Ly(t) + F(t, y(t)).$$

Difficulties: We look for a numerical scheme to be stable and accurate for any initial condition and during both phases (fast and slow).

Drawbacks:
- explicit methods need very small time step (for stability)
- fully implicit methods are costly (slow).
Stiff ODEs
When $\varepsilon \ll 1$ solve $y'(t) = -\frac{1}{\varepsilon} y(t) + F(t, y(t))$.

1. In $\mathbb{R}$: $y'(t) = -\frac{1}{\varepsilon} y(t) - \sin t$. The solution is

$$y(t) = (y_0 - \frac{\varepsilon^2}{\varepsilon^2 + 1}) e^{-t/\varepsilon} - \frac{\varepsilon}{\varepsilon^2 + 1} (\sin t - \varepsilon \cos t).$$

Thus, ON the slow manifold

$$y(t) = \frac{-\varepsilon}{\varepsilon^2 + 1} \sin t + \frac{\varepsilon^2}{\varepsilon^2 + 1} \cos t.$$

2. In $\mathbb{C}$: $y'(t) = \frac{i}{\varepsilon} y(t) + e^{it}$. The solution is

$$y(t) = e^{it/\varepsilon} (y_0 + \frac{i}{1 - 1/\varepsilon}) - \frac{i}{1 - 1/\varepsilon} e^{it}.$$
Huge literature


Consider 3 methods avoiding the small time step:

- *ImEx* (or Linearly Implicit method),
- *Integrating Factor* (Lawson 1967)
- *Exponential time differencing* (Certaine 1960).
Huge literature


Consider 3 methods avoiding the small time step:

- *ImEx* (or Linearly Implicit method),
- *Integrating Factor* (Lawson 1967)
- *Exponential time differencing* (Certaine 1960).
Exponential time differencing - History

*ImEx* and *Integrating Factor* methods are frequently used for solving stiff PDE’s. *ETD* is less common but has been re-invented many times over the years.

- The term comes from “computational electrodynamics” (Holland 1994, Taflove 1995): *ETD1*
- Certaine 1960: ideas + multistep *ETD* methods of any order
- Nørsett 1969: arbitrary order A-stable exponential integrator
- ... 
- Cox & Matthews 2002: formulas for *ETD Runge-Kutta* methods of order up to 4. *ETD* is superior over *ImEx* and *Integrating Factor*! 
- Kassam & Trefethen 2005: *ETDRK4* is tested against 5 other 4th order schemes on several PDEs.
- ...
**ImEx** and **Integrating Factor** methods

**ImEx** : implicit formula to advance the linear part
  explicit formula to advance the nonlinear part.
  - works well on the slow manifold.
  - fails to capture the stiff behaviour.
  - for A-stability, cannot extend beyond 2nd order.

**Integrating Factor** : multiply the ODE by $e^{-tL/\varepsilon}$:

$$(e^{-tL/\varepsilon} y)' = e^{-tL/\varepsilon} F(t, y(t))$$

or
$$u' = e^{-tL/\varepsilon} F(t, e^{tL/\varepsilon} u)$$
  and use an explicit scheme.
  - inaccurate for $F$ slowly varying
  - the stiff part is solved exactly.
  - can extend to any order.
Exact solution to

\[(e^{-tL/\varepsilon}y)' = e^{-tL/\varepsilon}F(t, y(t))\]

is

\[y_{n+1} = e^{(\Delta t/\varepsilon) L}y_n + e^{(\Delta t/\varepsilon) L} \int_{t_n}^{t_{n+1}} e^{(t_n-\tau)/\varepsilon} L F(\tau, y(\tau)) \, d\tau\]

- smaller errors than IF on the slow manifold.
- the stiff part is solved exactly.
- can extend to any order.
Examples of *Exponential time differencing* schemes

- by Multi-step methods

**ETD1**: approx $F$ on $[t_n; t_{n+1}]$ by $F_n$

$$y_{n+1} = e^{(\Delta t/\varepsilon)L}y_n + F_n \varepsilon(e^{(\Delta t/\varepsilon)L} - 1)$$

**ETD2**: approx $F$ on $[t_n; t_{n+1}]$ by $\tau \mapsto F_n + (\tau - t_n)(F_n - F_{n-1})/\Delta t$

... 

- by Runge-Kutta methods

**ETD2RK**: approx $F$ on $[t_n; t_{n+1}]$ by $\tau \mapsto F_n + (\tau - t_n)(\tilde{F}_{n+1} - F_n)/\Delta t$, where

$$\tilde{F}_{n+1} = F(t_{n+1}, e^{(\Delta t/\varepsilon)L}y_n + F_n \varepsilon(e^{(\Delta t/\varepsilon)L} - 1))$$

**ETD3RK**: approx $F$ on $[t_n; t_{n+1}]$ by a quadratic interpolant ...

Sever Hirstoaga

An exponential integrator for highly oscillatory Vlasov-Poisson systems
Particular cases

1) $L = -1$ : Solve

$$y'(t) = -\frac{1}{\varepsilon}y(t) + F(t, y(t))$$

- ETD is used only for computing precisely the rapid decay (when the initial condition is OFF the slow manifold).
- Implicit Euler works very well (out of the fast phase) even with big time steps w.r.t. $\varepsilon$.

2) $L = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}$ : Solve

$$\begin{cases} R'(t) = \frac{1}{\varepsilon} V(t) \\ V'(t) = -\frac{1}{\varepsilon} R(t) + E(t, R(t)) \end{cases}$$

ETD computes exactly the fast oscillations. Implicit Euler drifts inward.
Fast oscillations case

\[ L = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \Rightarrow \ e^{tL} = \begin{pmatrix} \cos t & \sin t \\ -\sin t & \cos t \end{pmatrix} =: \mathcal{R}(t). \]

\[ E(t, r) \text{ is either given by Poisson or by } E(t, r) = -r^3. \]

The ETD is

\[
\begin{pmatrix} R_{n+1} \\ V_{n+1} \end{pmatrix} = \mathcal{R}\left(\frac{\Delta t}{\varepsilon}\right) \begin{pmatrix} R_n \\ V_n \end{pmatrix} + \mathcal{R}\left(\frac{\Delta t}{\varepsilon}\right) \int_{t_n}^{t_{n+1}} \mathcal{R}\left(\frac{t_n - \tau}{\varepsilon}\right) \begin{pmatrix} 0 \\ E(\tau, R(\tau)) \end{pmatrix} d\tau
\]

Tests with the approximation \textit{ETD2RK} : linear interpolation of the slow integrand through \( t_n \) and \( t_{n+1} \) by using \textit{ETD1} for the prediction at \( t_{n+1} \).

\[ \Rightarrow \text{inaccurate results} \]

since for \( \Delta t \geq 2\varepsilon \) errors are significant (especially for particles off the slow manifold). The beam do not spiral at the good rate.
Fast oscillations case

- \( L = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \Rightarrow e^{tL} = \begin{pmatrix} \cos t & \sin t \\ -\sin t & \cos t \end{pmatrix} =: R(t). \)

- \( E(t, r) \) is either given by Poisson or by \( E(t, r) = -r^3. \)

The ETD is

\[
\begin{pmatrix} R_{n+1} \\ V_{n+1} \end{pmatrix} = \mathcal{R}\left(\frac{\Delta t}{\varepsilon}\right) \begin{pmatrix} R_n \\ V_n \end{pmatrix} + \mathcal{R}\left(\frac{\Delta t}{\varepsilon}\right) \int_{t_n}^{t_{n+1}} \mathcal{R}\left(\frac{t_n - \tau}{\varepsilon}\right) \begin{pmatrix} 0 \\ E(\tau, R(\tau)) \end{pmatrix} d\tau
\]

Tests with the approximation \( ETD2RK \): linear interpolation of the slow integrand through \( t_n \) and \( t_{n+1} \) by using \( ETD1 \) for the prediction at \( t_{n+1} \).

\( \Rightarrow \) inaccurate results

since for \( \Delta t \geq 2\varepsilon \) errors are significant (especially for particles off the slow manifold). The beam do not spiral at the good rate.
Fast oscillations case

\[ L = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix} \Rightarrow e^{tL} = \begin{pmatrix} \cos t & \sin t \\ -\sin t & \cos t \end{pmatrix} =: \mathcal{R}(t). \]

- \( E(t, r) \) is either given by Poisson or by \( E(t, r) = -r^3 \).

The ETD is

\[
\begin{pmatrix} R_{n+1} \\ V_{n+1} \end{pmatrix} = \mathcal{R}\left(\frac{\Delta t}{\varepsilon}\right) \begin{pmatrix} R_n \\ V_n \end{pmatrix} + \mathcal{R}\left(\frac{\Delta t}{\varepsilon}\right) \int_{t_n}^{t_{n+1}} \mathcal{R}\left(\frac{t_n - \tau}{\varepsilon}\right) \begin{pmatrix} 0 \\ E(\tau, R(\tau)) \end{pmatrix} d\tau
\]

Tests with the approximation \textit{ETD2RK} : linear interpolation of the slow integrand through \( t_n \) and \( t_{n+1} \) by using \textit{ETD1} for the prediction at \( t_{n+1} \).

\[ \Rightarrow \text{inaccurate results} \]

since for \( \Delta t \geq 2\varepsilon \) errors are significant (especially for particles off the slow manifold). The beam do not spiral at the good rate.
Global errors: the case $E(t, r) = -r^3$

$\epsilon = 10^{-2}$, final time $= \pi$, size of the beam $\approx 1$. Starting with a particle **on** (left) and **off** (right) the slow manifold.

Similar results in the coupling with Poisson case.
The new ETD algorithm – big time steps
If we want $\Delta t \gg \varepsilon$ then find the integer $N$ and the real $o$ s.t.

$$\Delta t = N \cdot (2\pi \varepsilon) + o$$

$2\pi \varepsilon$ is an approximation for the fast time for one grand tour.

Thus the integral term in the exact ETD writes

$$\int_{t_n}^{t_{n+1}} d\tau = \sum_{j=0}^{N-1} \int_{t_n+2\pi \varepsilon j}^{t_n+2\pi \varepsilon (j+1)} d\tau + \int_{t_{n+1} - o}^{t_{n+1}} d\tau$$

that we approximate by

$$N \int_{t_n}^{t_n+2\pi \varepsilon} d\tau + \int_{t_{n+1} - o}^{t_{n+1}} d\tau$$
By the exact ETD we have

\[ \mathcal{I}_1 = \int_{t_n}^{t_n + 2\pi \varepsilon} d\tau = \left( \begin{array}{c} R(t_n + 2\pi \varepsilon) - R(t_n) \\ V(t_n + 2\pi \varepsilon) - V(t_n) \end{array} \right). \]

we have to compute \( \left( R(t_n + N \cdot 2\pi \varepsilon), V(t_n + N \cdot 2\pi \varepsilon) \right) \) since needed in the 3rd step. By the exact ETD we have

\[ \left( \begin{array}{c} R(t_n + N \cdot 2\pi \varepsilon) \\ V(t_n + N \cdot 2\pi \varepsilon) \end{array} \right) \circa{\approx} \left( \begin{array}{c} R_n \\ V_n \end{array} \right) + N \cdot \mathcal{I}_1 \]

likewise, \( \mathcal{I}_2 = \int_{t_{n+1} - \varepsilon}^{t_{n+1}} d\tau \) is

\[ \mathcal{I}_2 = \mathcal{R} \left( - \frac{\varepsilon}{\varepsilon} \right) \left( \begin{array}{c} \tilde{R}(t_{n+1}) \\ \tilde{V}(t_{n+1}) \end{array} \right) - \left( \begin{array}{c} R(t_n + N \cdot 2\pi \varepsilon) \\ V(t_n + N \cdot 2\pi \varepsilon) \end{array} \right). \]

where \( R(t_n + 2\pi \varepsilon) \) and \( \tilde{R}(t_{n+1}) \) need to be calculated by RK4 with small step. Replacing these formulae in the exact ETD leads to

\[ R_{n+1} = \tilde{R}(t_{n+1}) \quad \text{and} \quad V_{n+1} = \tilde{V}(t_{n+1}) \]
The ODE’s solution \((R_n, V_n)\) at time \(t_n\) is given. Then

1. compute \((R, V)\) at time \(t_n + 2\pi\varepsilon\) by using a fine Runge-Kutta solver with initial condition \((R_n, V_n)\).

2. compute \((R, V)\) at time \(t_n + N \cdot 2\pi\varepsilon\) by the following rule

\[
\begin{pmatrix}
R(t_n + N \cdot 2\pi\varepsilon) \\
V(t_n + N \cdot 2\pi\varepsilon)
\end{pmatrix}
= \begin{pmatrix}
R_n \\
V_n
\end{pmatrix}
+ N \begin{pmatrix}
R(t_n + 2\pi\varepsilon) - R_n \\
V(t_n + 2\pi\varepsilon) - V_n
\end{pmatrix}.
\]

3. compute \((R, V)\) at time \(t_{n+1}\) by using a fine Runge-Kutta solver with initial condition \((R, V)\) obtained at the previous step.

Assumption:

\[
\int_{t_n}^{t_n+N(2\pi\varepsilon)} \mathcal{R}\left(\frac{t_n - \tau}{\varepsilon}\right) \begin{pmatrix}
0 \\
E(\tau, R(\tau))
\end{pmatrix} d\tau \simeq N \cdot \int_{t_n}^{t_n+2\pi\varepsilon} \ldots d\tau
\]
Test case (2D phase space)

- initial condition for Vlasov:
  \[ f_0(r, \nu) = \frac{1}{\sqrt{2\pi} \nu_{th}} e^{-\nu^2/(2\nu_{th}^2)} 1_{[-R,R]}(r) \]

  where \( \nu_{th} = 0.0727 \) and \( R = 0.75 \).

- \( N_p = 10000 \) particles.

- \( E(t, r) = -r^3 \) or \( -r \).

- coupling with Poisson equation
  \[ \frac{1}{r} \frac{\partial(r E)}{\partial r} = \int f(t, r, \nu) \, d\nu \]

  trapezoidal rule with 128 cells.
Global errors at final time 3.5

$E = -r^3$

\begin{align*}
\text{ONsm i.c.} & \quad \text{OFFsm i.c.} \\
\text{Sever Hirstoaga} & \quad \text{An exponential integrator for highly oscillatory Vlasov-Poisson systems}
\end{align*}
A more accurate (mean) period

Use of an inaccurate period in the Algorithm can lead to instability

Example:

\[
\begin{align*}
R'(t) &= \frac{1}{\varepsilon} V(t) \\
V'(t) &= -\frac{1}{\varepsilon} R(t) - R(t).
\end{align*}
\]

- A phase space trajectory is an ellipse
- The rapid period is \( T = \frac{2\pi\varepsilon}{\sqrt{1+\varepsilon}} \) for all the initial particles => not spiraling beam.
- The slow manifold \( \{(0,0)\} \)
- ONsm i.c. \( r_0 \sim 0.306, \; v_0 \sim 7 \cdot 10^{-6} \) and
  - OFFsm i.c. \( r_0 \sim 0.748, \; v_0 \sim 0.142 \)
- Using \( 2\pi\varepsilon \) instead of \( T \) drifts particles outward in the phase space
A more accurate (mean) period

Use of an inaccurate period in the Algorithm can lead to instability

Example:

\[
\begin{align*}
R'(t) &= \frac{1}{\varepsilon} V(t) \\
V'(t) &= -\frac{1}{\varepsilon} R(t) - R(t).
\end{align*}
\]

- a phase space trajectory is an ellipse
- the rapid period is \( T = \frac{2\pi\varepsilon}{\sqrt{1+\varepsilon}} \) for all the initial particles \( \Rightarrow \) not spiraling beam.
- the slow manifold \( \{(0,0)\} \)
- ONsm i.c. \( r_0 \sim 0.306, \ v_0 \sim 7 \cdot 10^{-6} \) and
  OFFsm i.c. \( r_0 \sim 0.748, \ v_0 \sim 0.142 \)
- using \( 2\pi\varepsilon \) instead of \( T \) drifts particles outward in the phase space
Global errors at final time 3.5 with the mean period

$E = -r^3$

ONsm i.c. $\leftarrow$ Poisson $\downarrow$ OFFsm i.c. $\rightarrow$

An exponential integrator for highly oscillatory Vlasov-Poisson systems
The case $E(t, r) = -r^3$

$\varepsilon = 10^{-4}$, time step $= 8750 \varepsilon$, final time $= 3.5$, using for particles period $2\pi \varepsilon$ (at left) and the mean period (at right).
Vlasov-Poisson case

\[ \varepsilon = 10^{-4}, \ \text{time step} = 8750 \varepsilon, \ \text{final time} = 3.5, \ \text{using for particles period} \ 2\pi \varepsilon \ (\text{at left}) \ \text{and the mean period} \ (\text{at right}). \]
Second model – 4D phase space

\[ X'(t) = V(t), \quad X(0) = x_0 \]

\[ V'(t) = \frac{1}{\varepsilon} V^\perp(t) + E^\varepsilon(t, X(t)), \quad V(0) = v_0 \]

where \( E^\varepsilon(x, t) = \begin{pmatrix} 2x_1 + x_2 \\ x_1 + 2x_2 \end{pmatrix} \) or coupling with Poisson eq.

The exponential integrator in velocity:

\[ V(t) = e^{\frac{t-s}{\varepsilon}L} V(s) + e^{\frac{t-s}{\varepsilon}L} \int_s^t e^{\frac{s-\tau}{\varepsilon}L} E^\varepsilon(X(\tau), \tau) \, d\tau. \]

\[ X(t) = X(s) + \int_s^t V(\tau) \, d\tau. \]
The algorithm (ETD-PIC scheme)

Find the integer $N$ and the real $o$ s.t.

$$\Delta t = N \cdot (2\pi \varepsilon) + o$$

$2\pi \varepsilon$ is an approximation for the fast time for one grand tour.

Assumption:

$$\int_{t_n}^{t_n+N(2\pi \varepsilon)} \mathcal{R} \left( \frac{t_n - \tau}{\varepsilon} \right) E^\varepsilon(X(\tau), \tau) d\tau \simeq N \cdot \int_{t_n}^{t_n+2\pi \varepsilon} \ldots d\tau$$

The ODEs solution $(X_n, V_n)$ at time $t_n$ is given. Then

1. compute $(X, V)$ at time $t_n + 2\pi \varepsilon$ by using a fine Runge-Kutta solver with initial condition $(X_n, V_n)$.

2. compute $(X, V)$ at time $t_n + N \cdot 2\pi \varepsilon$ by the following rule

$$\begin{pmatrix}
X(t_n + N \cdot 2\pi \varepsilon) \\
V(t_n + N \cdot 2\pi \varepsilon)
\end{pmatrix} = \begin{pmatrix}
X_n \\
V_n
\end{pmatrix} + N \begin{pmatrix}
X(t_n + 2\pi \varepsilon) - X_n \\
V(t_n + 2\pi \varepsilon) - V_n
\end{pmatrix}.$$  

3. compute $(X, V)$ at time $t_{n+1}$ by using a fine Runge-Kutta solver with initial condition $(X, V)$ obtained at the previous step.
The linear case

\[ X_1^\varepsilon (t; x_0, v_0) = K_1^\varepsilon \left( \cos (a_{\varepsilon} t) - \frac{a_{\varepsilon}}{\varepsilon} \sin (a_{\varepsilon} t) \right) + K_2^\varepsilon \left( \sin (a_{\varepsilon} t) + \frac{a_{\varepsilon}}{\varepsilon} \cos (a_{\varepsilon} t) \right) + K_3^\varepsilon \left( \cos (b_{\varepsilon} t) - \frac{b_{\varepsilon}}{\varepsilon} \sin (b_{\varepsilon} t) \right) + K_4^\varepsilon \left( \sin (b_{\varepsilon} t) + \frac{b_{\varepsilon}}{\varepsilon} \cos (b_{\varepsilon} t) \right), \]

\[ X_2^\varepsilon (t; x_0, v_0) = -K_1^\varepsilon u_{\varepsilon} \cos (a_{\varepsilon} t) - K_2^\varepsilon u_{\varepsilon} \sin (a_{\varepsilon} t) - K_3^\varepsilon v_{\varepsilon} \cos (b_{\varepsilon} t) - K_4^\varepsilon v_{\varepsilon} \sin (b_{\varepsilon} t), \]

\[ V_1^\varepsilon (t; x_0, v_0) = -K_1^\varepsilon a_{\varepsilon} \left( \frac{a_{\varepsilon}}{\varepsilon} \cos (a_{\varepsilon} t) + \sin (a_{\varepsilon} t) \right) + K_2^\varepsilon a_{\varepsilon} \left( \cos (a_{\varepsilon} t) - \frac{a_{\varepsilon}}{\varepsilon} \sin (a_{\varepsilon} t) \right) - K_3^\varepsilon b_{\varepsilon} \left( \frac{b_{\varepsilon}}{\varepsilon} \cos (b_{\varepsilon} t) + \sin (b_{\varepsilon} t) \right) + K_4^\varepsilon b_{\varepsilon} \left( \cos (b_{\varepsilon} t) - \frac{b_{\varepsilon}}{\varepsilon} \sin (b_{\varepsilon} t) \right), \]

\[ V_2^\varepsilon (t; x_0, v_0) = K_1^\varepsilon a_{\varepsilon} u_{\varepsilon} \sin (a_{\varepsilon} t) - K_2^\varepsilon a_{\varepsilon} u_{\varepsilon} \cos (a_{\varepsilon} t) + K_3^\varepsilon b_{\varepsilon} v_{\varepsilon} \sin (b_{\varepsilon} t) - K_4^\varepsilon b_{\varepsilon} v_{\varepsilon} \cos (b_{\varepsilon} t), \]

where \( a_{\varepsilon} \sim \sqrt{3\varepsilon} \) and \( b_{\varepsilon} \sim 1/\varepsilon \) and \( (K_i^\varepsilon)_i \) depend on the i.c. \((x_0, v_0)\).

The slow manifold obtained when \( K_3^\varepsilon (x_0, v_0) = 0 \) and \( K_4^\varepsilon (x_0, v_0) = 0 \) \( \Rightarrow \) a two dimensional space in \( \mathbb{R}^4 \).
Different initial particles

- **Tests with 3 initial conditions**

\[
(x_0^1, v_0^1) = \left(1, 0, \varepsilon, -\frac{2\varepsilon u_\varepsilon}{2 - \varepsilon^2} + \frac{\varepsilon^3 u_\varepsilon}{2 - \varepsilon^2}\right) \quad \text{on the slow manifold}
\]

\[
(x_0^2, v_0^2) = \left(1, -\frac{u_\varepsilon}{w_\varepsilon}, \varepsilon \frac{w_\varepsilon}{u_\varepsilon}, -\frac{2\varepsilon u_\varepsilon}{2 - \varepsilon^2} + \varepsilon \frac{w_\varepsilon}{u_\varepsilon} + \frac{\varepsilon^3 u_\varepsilon}{2 - \varepsilon^2}\right) \quad \text{close to the slow manifold}
\]

\[
(x_0^3, v_0^3) = (1, 1, 1, 1) \quad \text{far from the slow manifold}
\]

- **Tests with a beam**

\[
f_0(x, v) = \frac{1}{8\pi^2 v_{th}^2} (1 + \eta \cos (k \cdot x)) \chi(x) \exp \left(-\frac{v_1^2 + v_2^2}{2v_{th}^2}\right),
\]

with \(k_1 = 0.5, \ k_2 = 0, \ v_{th} = 0.1, \ \eta = 0.1, \ \text{and}\)

\[
\chi(x) = \chi_{[0,4\pi]}(x_1) \chi_{[0,1]}(x_2).
\]
Different initial particles

- Tests with 3 initial conditions

\[(x_0^1, v_0^1) = \left( 1, 0, \varepsilon, -\frac{2\varepsilon u_\varepsilon}{2 - \varepsilon^2} + \frac{\varepsilon^3 u_\varepsilon}{2 - \varepsilon^2} \right) \quad \text{on the slow manifold}\]

\[(x_0^2, v_0^2) = \left( 1, -\frac{u_\varepsilon}{w_\varepsilon}, \varepsilon \frac{w_\varepsilon}{u_\varepsilon}, -\frac{2\varepsilon u_\varepsilon}{2 - \varepsilon^2} + \varepsilon \frac{w_\varepsilon}{u_\varepsilon} + \frac{\varepsilon^3 u_\varepsilon}{2 - \varepsilon^2} \right) \quad \text{close to the slow manifold}\]

\[(x_0^3, v_0^3) = (1, 1, 1, 1) \quad \text{far from the slow manifold}\]

- Tests with a beam

\[f_0(x, v) = \frac{1}{8\pi^2 v_{th}^2} (1 + \eta \cos(k \cdot x)) \chi(x) \exp \left( -\frac{v_1^2 + v_2^2}{2v_{th}^2} \right) ,\]

with \(k_1 = 0.5, \ k_2 = 0, \ v_{th} = 0.1, \ \eta = 0.1, \ \text{and} \)

\[\chi(x) = \chi[0, 4\pi](x_1) \chi[0, 1](x_2).\]
Global errors at final time 10

An initial condition close to the slow manifold
Global errors at final time $10: \varepsilon = 10^{-2}, 10^{-3}, 5 \cdot 10^{-4}, 10^{-4}$
Landau damping: periodic boundary conditions on $\Omega_x = [0; 4\pi] \times [0; 1]$

$$f_0(x, v) = \frac{1}{2\pi v_{th}^2} (1 + \eta \cos (k \cdot x)) \exp \left( -\frac{v_1^2 + v_2^2}{2v_{th}^2} \right),$$

with $k_1 = 0.5$, $k_2 = 0$, $v_{th} = 0.1$, $\eta = 0.1$. 

An exponential integrator for highly oscillatory Vlasov-Poisson systems
numerical scheme to solve highly oscillatory ODEs – need to know the (numerical) fast period.

allows big time steps with respect to this period.

Outlook

long time behaviour of the scheme

numerical comparisons with the two-scale limit model and the guiding center model

Finite Larmor Radius test case

THANK YOU!
Conclusions

- numerical scheme to solve highly oscillatory ODEs – need to know the (numerical) fast period.
- allows big time steps with respect to this period.

Outlook

- long time behaviour of the scheme
- numerical comparisons with the two-scale limit model and the guiding center model
- Finite Larmor Radius test case

THANK YOU!