Screw-pinch simulations in SeLaLib: from reduced drift-kinetics to gyrokinetics

Yaman Güçlü ¹  Edoardo Zoni ¹  Michel Mehrenberger ²
Eric Sonnendrücker ¹

¹Division of Numerical Methods in Plasma Physics, Max-Planck-Institut für Plasmaphysik, Garching bei München, Germany

²Institut de Recherche Mathématique Avancée, University of Strasbourg, France

Collaborators:
Guillaume Latu, Maurizio Ottaviani, Selalib team, Gysela team

NumKin 2016, Strasbourg, France, 17-21 October 2016
# Introduction

## Motivation
- Study turbulent transport in magnetic confinement devices
- Expensive gyrokinetic simulations in complex geometries

## Observation: exploit anisotropy
- Linear regime: most unstable modes have large parallel wavelengths ($\lambda_\parallel \gg \rho_i$) along magnetic field $B$
- Non-linear regime: turbulence structures stretched along $B$

## Field-aligned interpolation approach $[^a]$
- Discretize poloidal planes with favourite (fine) mesh
- Between poloidal planes interpolate solution along $B$
- Allows reduction of grid points along toroidal direction $\varphi$

---

Introduction

Field-aligned semi-Lagrangian method

- Vlasov-like kinetic equation:

\[
\frac{d}{dt} f(t, Z(t)) = 0, \quad \frac{dZ}{dt} = \ldots
\]

- Method of characteristics:
  1. Trace characteristic trajectories backward in time
  2. Interpolate solution at foot of characteristics

- Use field-aligned interpolation at step (2) above
Introduction

Standard interpolation
- centered rectangular stencil
- requires fine mesh in $\varphi$

\begin{align*}
\theta & \quad 2\pi \\
\varphi & \quad 2\pi
\end{align*}
Field-aligned interpolation

- stencil adapts to $B$ field line
- allows for coarser mesh in $\varphi$
Introduction

Previous results [\textsuperscript{a}]

- Analysis: stability proof for 2D advection, error estimates
- Numerical tests: ITG instability in cylinder and torus
- Memory reduction: $\lambda_{\parallel} / \lambda_{\phi} \approx 0.1 / \rho^* \gg 1$

Screw-pinch simulation in Selalib

- Selalib library is collection of algorithms for building semi-Lagrangian and particle-in-cell simulations
- As of June 2016, screw-pinch simulation in Selalib implemented \textit{reduced drift-kinetic equations}
- This talk: new simulation implements full drift kinetic model

\textsuperscript{a} G. LATU, M. MEHRENBERGER, Y. G"UL"U, M. OTTAVIANI, AND E. SONNENDR"UCKER. \url{https://hal.archives-ouvertes.fr/hal-01315889} (2016).
Outline

- Gyrokinetic Vlasov’s equation
- Predictor-corrector time-stepping
- Field-aligned splitting
- Field-aligned advection solver
- Screw-pinch model (full / reduced)
- Numerical results
Gyrokinetic Vlasov’s equation

Gyrocenter distribution function $f(t, \mathbf{x}, v_\parallel, \mu)$

- $\mathbf{x}$ is gyrocenter position
- $v_\parallel$ is parallel velocity
- $\mu$ is modified magnetic moment (invariant)
Gyrokinetic Vlasov’s equation

3D-1V, parametric in $\mu$, may be formulated in different forms.

- Eulerian advective form:
  $$\left( \frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla_{\mathbf{x}} + a_{\parallel} \frac{\partial}{\partial v_{\parallel}} \right) f(t, \mathbf{x}, v_{\parallel}, \mu) = 0$$

- Lagrangian form:
  $$\frac{d}{dt} f(t, \mathbf{X}(t), V_{\parallel}(t), \mu) = 0, \quad \frac{d}{dt} \begin{bmatrix} \mathbf{X} \\ V_{\parallel} \end{bmatrix} = \begin{bmatrix} \mathbf{u}(t, \mathbf{X}, V_{\parallel}, \mu) \\ a_{\parallel}(t, \mathbf{X}, V_{\parallel}, \mu) \end{bmatrix}$$

- Eulerian conservative form:
  $$\frac{\partial}{\partial t} \left( B_{\parallel}^* f \right) + \nabla_{\mathbf{x}} \cdot \left( \mathbf{u} B_{\parallel}^* f \right) + \frac{\partial}{\partial v_{\parallel}} \left( a_{\parallel} B_{\parallel}^* f \right) = 0$$

$[B_{\parallel}^*(\mathbf{x}, v_{\parallel})$ is Jacobian determinant of coordinate transformation$]
Gyrokinetic Vlasov’s equation

- Phase-space flow field:

\[
\begin{align*}
\mathbf{u}(t, \mathbf{x}, v_\parallel, \mu) &= \frac{1}{B_\parallel^*} \left( \frac{1}{m} \frac{\partial H}{\partial v_\parallel} \mathbf{B}^* + \frac{1}{q} \mathbf{b} \times \nabla_x H \right) \\
\mathbf{a}_\parallel(t, \mathbf{x}, v_\parallel, \mu) &= \frac{1}{B_\parallel^*} \left( -\frac{1}{m} \mathbf{B}^* \cdot \nabla_x H \right)
\end{align*}
\]

- Modified magnetic field:

\[
\mathbf{B}^*(\mathbf{x}, v_\parallel) = \mathbf{B} + \frac{mv_\parallel}{q} \nabla_x \times \mathbf{b} \\
B_\parallel^*(\mathbf{x}, v_\parallel) = \mathbf{B}^* \cdot \mathbf{b}
\]

- \(H(t, \mathbf{x}, v_\parallel, \mu)\) is single-particle gyrocenter Hamiltonian
- Phase-space flow field \((\mathbf{u}, \mathbf{a}_\parallel)\) is divergence-free, because

\[
\frac{1}{B_\parallel^*} \left[ \nabla_x \cdot (B_\parallel^* \mathbf{u}) + \frac{\partial}{\partial v_\parallel} \left( B_\parallel^* \mathbf{a}_\parallel \right) \right] = 0
\]
Gyrokinetic Vlasov’s equation

Non-linearity

- \((u, a_{||})\) are time-varying through Hamiltonian \(H\)
- \(H\) is time-varying through electromagnetic potentials \((\phi, A)\)
- \((\phi, A)\) depend on instantaneous moments of distribution \(f\)

E.g. electrostatic case with kinetic ions and adiabatic electrons:

\[
H(t, \mathbf{x}, v_{||}, \mu) = \frac{1}{2}mv_{||}^2 + \mu B(\mathbf{x}) + q\langle \phi \rangle_\alpha(t, \mathbf{x}, \mu),
\]

\[
- \left( \nabla_\perp \cdot \frac{\rho_{th,i}^2}{\lambda_{D,i}^2} \nabla_\perp \right) \phi + \frac{\phi - \langle \phi \rangle_f}{\lambda_{D,e}^2} = \frac{1}{\varepsilon_0} \sigma_i[f](t, \mathbf{x}),
\]

with non-local integral operators: \(\langle \cdot \rangle_\alpha\) gyro-average, \(\langle \cdot \rangle_f\) average on magnetic flux-surface, \(\sigma_i[\cdot]\) charge density
### Time-advancing strategy

#### Main idea

Within each time step, choose an approximate **time-invariant** Hamiltonian \( \tilde{H}(x, v_\parallel, \mu) \) and solve the **linear** equation

\[
\left( \frac{\partial}{\partial t} + \tilde{u}(x, v_\parallel, \mu) \cdot \nabla x + \tilde{a}_\parallel(x, v_\parallel, \mu) \frac{\partial}{\partial v_\parallel} \right) f(t, x, v_\parallel, \mu) = 0 \quad (1)
\]

#### Predictor-corrector scheme

- **predictor** sets \( \tilde{H} = H[f_0] \) and solves (1) on \([t_0, t_1/2]\) to obtain \( f^* \)

- **corrector** sets \( \tilde{H} = H[f^*] \) and solves (1) on \([t_0, t_1]\) to obtain \( f_1 \)
Time-advancing strategy

Linear gyrokinetic Vlasov’s equation:

\[
\left( \partial_t + \mathbf{\tilde{u}}(x, v_\parallel, \mu) \cdot \nabla_x + \mathbf{\tilde{a}}_\parallel(x, v_\parallel, \mu) \partial_{v_\parallel} \right) f(t, x, v_\parallel, \mu) = 0
\]

Field-aligned splitting

- Use non-orthogonal vector decomposition \( \mathbf{\tilde{u}} = \mathbf{\tilde{u}}_{\text{pol}} + \mathbf{\tilde{u}}_b \mathbf{b} \), with \( \mathbf{u}_{\text{pol}} \) tangent to local poloidal plane, \( u_b \) scalar
- Decompose kin. eq. into 3 separate advection equations:
  1. Advection on poloidal plane \( (\partial_t + \mathbf{\tilde{u}}_{\text{pol}} \cdot \nabla_x) f = 0 \)
  2. Advection along magnetic field \( (\partial_t + \mathbf{\tilde{u}}_b \mathbf{b} \cdot \nabla_x) f = 0 \)
  3. Advection along \( v_\parallel \) \( (\partial_t + \mathbf{\tilde{a}}_\parallel \partial_{v_\parallel}) f = 0 \)
- Solve each advection eq. with semi-Lagrangian method
Time-advancing strategy

1. Advection on poloidal plane
   \[(\partial_t + \tilde{u}_{\text{pol}} \cdot \nabla_x) f = 0\]
2. Advection along magnetic field
   \[(\partial_t + \tilde{u}_b b \cdot \nabla_x) f = 0\]
3. Advection along \(v_{\parallel}\)
   \[(\partial_t + \tilde{a}_{\parallel} \partial_{v_{\parallel}}) f = 0\]

**Predictor**: Lie splitting
- Compute \(\tilde{H} = H[f_0]\)
- Initial conditions: \(f_0\)
  - Advance (2) by \(\Delta t/2\)
  - Advance (3) by \(\Delta t/2\)
  - Advance (1) by \(\Delta t/2\)
- Final solution: \(f^*\)

**Corrector**: Strang splitting
- Compute \(\tilde{H} = H[f^*]\)
- Initial conditions: \(f_0\)
  - Advance (2) by \(\Delta t/2\)
  - Advance (3) by \(\Delta t/2\)
  - Advance (1) by \(\Delta t\)
  - Advance (3) by \(\Delta t/2\)
  - Advance (2) by \(\Delta t/2\)
- Final solution: \(f_1\)
Choice of Coordinates

- Consider uniform grid in curvilinear coordinates $\eta = (\eta_1, \eta_2, \eta_3)$
  
  $$x = F(\eta)$$

- Always choose $F$ s.t. $\eta_3 = \text{const.}$ on poloidal plane
  $\Rightarrow$ operator (1) is 2D advection in $(\eta_1, \eta_2)$, parametric in $(\eta_3, v_\parallel, \mu)$

- Open choices for $(\eta_1, \eta_2)$:
  - magnetic flux coordinates ($\eta_1 = \text{const.}$ along magnetic field line):
    $\Rightarrow$ operator (2) is 2D advection on flux surface $(\eta_2, \eta_3)$
  - generic coordinates: operator (2) is 3D advection in $(\eta_1, \eta_2, \eta_3)$
Field-aligned advection

\[
(\partial_t + \tilde{\mathbf{u}}_b \mathbf{b} \cdot \nabla_x) f = 0
\]

- B field tangent to $\Psi = \text{const}$ surfaces
- Operator involves 2 (conformal) or 3 (non-conformal) coords.
- “Remapping-based” parallelization: conformal mesh preferred
Field-aligned advection

Magnetic field line $X_L(\varphi)$ parametrized by toroidal angle $\varphi$:

\[
\begin{align*}
\frac{dX_L}{d\varphi} &= b \frac{d\varphi}{ds} \\
\frac{dX_L}{d\varphi} &= \hat{\varphi} \cdot dX_L = b_\varphi ds
\end{align*}
\]

\[
\Rightarrow \quad \frac{dX_L}{d\varphi} = \frac{b}{b_\varphi} \quad (b_\varphi \neq 0)
\]

In **pre-processing**, for each point on $(\varphi, \theta)$ grid:

- Reconstruct field line passing through point
- Store values of $\theta$ and $b_\varphi$ along line
Field-aligned advection

Method of characteristics for field-aligned advection:

\[
\frac{d}{dt} f(t, X(t), v\parallel, \mu) = 0 \quad \frac{dX}{dt} = \tilde{u}_b(X(t), v\parallel, \mu) b(X(t))
\]

Toroidal velocity:

\[
\frac{d\varphi}{dt} = \frac{dX}{dt} \cdot \varphi = \tilde{u}_b(X(t), v\parallel, \mu) b_\varphi(X(t))
\]

Characteristic equation \(\varphi(t)\) along field line \(X_L(\varphi)\):

\[
\frac{d\varphi}{dt} = \tilde{u}_b(X_L(\varphi), v\parallel, \mu) b_\varphi(X_L(\varphi))
\]
Field-aligned advection

Let

\[
F(\phi) = f(t, X_L(\phi), v_\parallel, \mu)
\]

\[
U(\phi) = \tilde{u}_b(X_L(\phi), v_\parallel, \mu) b_\phi(X_L(\phi))
\]

For each grid point on \((\phi, \theta)\) grid:

1. Interpolate \(f\) and \(\tilde{u}_b b_\phi\) along \(\theta\) at \((\phi_k, \theta^*_k)\) with \(k \in [k_{\text{min}}, k_{\text{max}}]\) to obtain \(F(\phi_k)\) and \(U(\phi_k)\)
2. Construct interpolants for \(F(\phi)\) and \(U(\phi)\) on \(\phi \in [\phi_{k_{\text{min}}}, \phi_{k_{\text{max}}}]\)
3. Integrate \(d\phi/dt = U(\phi)\) on \([t, t + \Delta t]\) to obtain foot of characteristic \(\phi^*\)
4. Set \(f(t + \Delta t, x, v_\parallel, \mu) = F(\phi^*)\)
Field-aligned advection

\[ \frac{\partial}{\partial \varphi} \phi \]

\[ U_{-2} \quad U_{-1} \quad U_0 \quad U_1 \quad U_2 \]

\[ F_{-2} \quad F_{-1} \quad F_0 \quad F_1 \quad F_2 \]
Screw-pinch, geometry

Magnetic configuration

Cylindrical coordinates \((r, \theta, z)\), tangent basis \(e_r, e_\theta, e_z\):

- \(B = B_0 \left( \frac{\zeta(r)}{r} e_\theta + e_z \right)\)
- \(\zeta(r) = \iota(r) r / R_0\)
- Rotational transform \(\iota(r)\) given

\[
\begin{align*}
B(r) &= |B_0| \sqrt{1 + \zeta^2} \\
b^\theta(r) &= \frac{\text{sgn}(B_0) \zeta}{r \sqrt{1 + \zeta^2}} \\
b^z(r) &= \frac{\text{sgn}(B_0)}{\sqrt{1 + \zeta^2}}
\end{align*}
\]

Properties

- \(B, b^\theta, b^z\) depend only on radial coordinate \(r\):

- Field lines straight on \((\theta, z)\) plane: \(d\theta / dz = \iota(r) / R_0 = \text{const.}\)
Screw-pinch, model equations

- \( \mu = 0 \), adiabatic electrons, electrostatic:

\[
H(t, x, v_\parallel) = \frac{1}{2}v_\parallel^2 + \phi(t, x) \quad \Rightarrow \quad \begin{cases} \\
\frac{\partial H}{\partial v_\parallel} = v_\parallel \\
\nabla_x H = \nabla_x \phi \end{cases}
\]

- Gyrokinetic Vlasov’s equation for ions:

\[
\left( \frac{\partial}{\partial t} + u_{rpol}' \frac{\partial}{\partial r} + u_{\theta pol}' \frac{\partial}{\partial \theta} + u_b \nabla_\parallel + \frac{\partial}{\partial v_\parallel} \right) f(t, r, \theta, z, v_\parallel) = 0
\]

- Quasi-neutrality equation without zonal flow for \( \phi(r, \theta, z) \):

\[
- \left( \frac{\partial^2 \phi}{\partial r^2} + \left( \frac{1}{r} + \frac{\partial r n_0}{n_0} \right) \frac{\partial \phi}{\partial r} + \frac{1}{r^2} \frac{\partial^2 \phi}{\partial \theta^2} \right) + \frac{1}{T_e} \phi = \frac{1}{n_0} \int (f - f_{eq}) \, dv
\]
Screw-pinch, model equations

\[ u_{\text{pol}}(t, r, \theta, z, v_\parallel) = \frac{1}{B^*_\parallel} \left[ -c_0 \frac{\partial \phi}{\partial \theta} + \zeta \nabla_\parallel \phi \right] e_r + \frac{1}{B^*_\parallel} \left[ c_0 \frac{\partial \phi}{\partial r} - c_2 v_\parallel^2 \right] e_\theta \]

\[ u_b(t, r, \theta, z, v_\parallel) = v_\parallel + \frac{1}{B^*_\parallel} \left[ c_3 v_\parallel^2 - \zeta \frac{\partial \phi}{\partial r} \right] \]

\[ a_\parallel(t, r, \theta, z, v_\parallel) = -\nabla_\parallel \phi + \frac{1}{B^*_\parallel} \left[ c_2 v_\parallel \frac{\partial \phi}{\partial \theta} - c_3 v_\parallel \nabla_\parallel \phi \right] \]

\[ c_0(r) = \text{sgn}(B_0) \frac{\sqrt{1 + \zeta^2}}{r} \quad c_1(r) = \frac{d\zeta}{dr} + \frac{\zeta}{r} \]

\[ c_2(r) = \text{sgn}(B_0) \frac{\zeta^2 / r^2}{\sqrt{1 + \zeta^2}} \quad c_3(r) = \frac{\zeta^3 / r}{\sqrt{1 + \zeta^2}} \]
Screw-pinch, reduced model

- Keep \((e_r, e_\theta, b)\) basis unchanged, and let \(\zeta \to 0\)
- Obtain \(c_0(r) = \text{sgn}(B_0)/r, c_1 = c_2 = c_3 = 0\) and \(B_\|^* = |B_0|:\)
  \[
  \left(\partial_t + [\phi, \cdot] + v_\| \nabla_\| - \nabla_\| \phi \partial_\| \right) f(t, r, \theta, z, v) = 0
  \]
  \[
  [\phi, \cdot] = -\frac{\partial_\theta \phi}{rB_0} \partial_r + \frac{\partial_r \phi}{rB_0} \partial_\theta, \quad \nabla_\| = b \cdot \nabla = b^\theta \partial_\theta + b^z \partial_z
  \]
- Field-aligned splitting:
  \[
  \partial_t f + [\phi, f] = 0 \quad \text{independent of } v_\|
  \]
  \[
  \partial_t f + v_\| \nabla_\| f = 0 \quad \text{constant advection along } b
  \]
  \[
  \partial_t f - \nabla_\| \phi \partial_\| f = 0 \quad \text{constant advection along } v_\|
  \]
- Reduced screw-pinch model in Selalib since 2015
- Full model just implemented and verified (October 2016)
Test Case: Ion Temperature Gradient (ITG) Instability

Computational domain:

\[ r \in [r_{\text{min}}, r_{\text{max}}], \quad \theta \in [0, 2\pi], \quad z \in [0, 2\pi R], \quad v \in [-v_{\text{max}}, v_{\text{max}}] \]

Initial conditions:

\[ f(0, r, \theta, z, v_\parallel) = f_{\text{eq}}(r, v) \left[ 1 + \epsilon \exp \left( -\frac{(r - r_p)^2}{\delta r} \right) \cos \left( m\theta + \frac{n}{R}z \right) \right] \]

Maxwellian equilibrium, changing with radius:

\[ f_{\text{eq}}(r, v) = \frac{n_0(r)}{\sqrt{2\pi T_i(r)}} \exp \left( -\frac{v^2}{2T_i(r)} \right) \]

Radial profiles \( n_0(r), T_e(r), T_i(r) \), and free parameters, given in [^a]

ITG instability, linear phase

Mode: \( m = 15, \ n = -11. \) Aspect ratio: \( R_0/r_{\text{max}} \approx 16.5 \)

Reduced model, numerical: \( W(t) \propto \exp(0.00383t) \)
Full screw-pinch, numerical: \( W(t) \propto \exp(0.00382t) \)

Reduced model, analytical: \( W(t) \propto \exp(0.00383t) \)
Full screw-pinch, analytical: \( W(t) \propto \exp(0.00382t) \)
ITG instability, linear phase

Mode: \( m = 15, n = -11 \). Aspect ratio: \( R_0/r_{\text{max}} = 5 \)

\[
\begin{align*}
\text{Reduced model, numerical} & : W(t) \propto \exp(0.00539t) \\
\text{Reduced model, analytical} & : W(t) \propto \exp(0.00539t) \\
\text{Full screw-pinch, numerical} & : W(t) \propto \exp(0.00413t) \\
\text{Full screw-pinch, analytical} & : W(t) \propto \exp(0.00413t)
\end{align*}
\]
ITG instability, non-linear phase
Poloidal cut at $z = 0$, $v_\parallel = 0$. Reduced model

$t = 4000$

$t = 5000$

$t = 6000$

Full screw-pinch

$t = 4000$

$t = 5000$

$t = 6000$
Parallel Performance

- Strong scaling on Draco (Hydra extension, MPCDF Garching)
- Intel Xeon E5-2698 processors: 32 cores/node, 2.3 GHz
- $N_r = 256$, $N_\theta = 512$, $N_\varphi = 32$, $N_{v\parallel} = 128$
- Efficiency = $\frac{\text{single\_proc\_wall\_time}}{\text{wall\_time} \times N_{\text{proc}}}$

![Graph showing efficiency vs. number of MPI processes for single node and full nodes. Ideal scaling: 100% efficiency. Single node efficiency: 77%, 51% for full nodes.]
## Summary

- Drift-kinetic model in **screw-pinch** geometry, $\mu = 0$
- Implementation in **Selalib**, remap-based domain decomposition
- Main ingredients: **field-aligned splitting** and **interpolation**
- Full model verified against dispersion relation
- High-order terms important when aspect ratio small
- Parallel code NOT optimized, but scaling is OK

## Future work

- Implement $\mu \neq 0$ and multi-$\mu$ models
- Toroidal geometry (circular Tokamak)
- Curvilinear coordinates in poloidal plane