

Implementation of Implicit-Explicit Time Integration for the Kinetic Modeling of Tokamak Plasma Edge

SIAM Conference on Computational Science and Engineering, Atlanta, GA

Debojyoti Ghosh¹ Milo Dorr¹ Mikhail Dorf² Jeffrey Hittinger¹

¹ Center for Applied Scientific Computing

² Physics Division

March 2, 2017



Background and Motivation

Inner edge: Adjacent to the core

- High temperature and density
- Mean free paths comparable to density/temperature gradients
- Weakly collisional

Requires kinetic simulation with collision model

Outer edge: Near tokamak wall

- Low temperature and density
- Short mean free paths compared to density/temperature gradients
- Strongly collisional

Introduces very small time scales

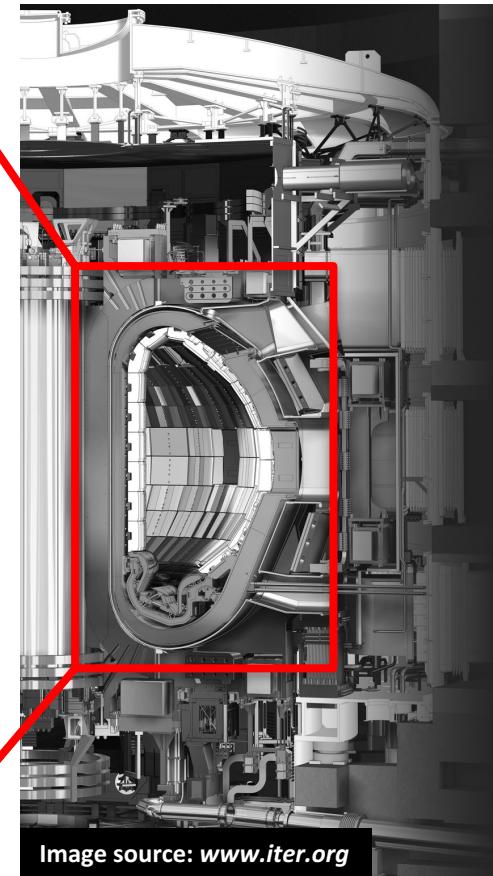
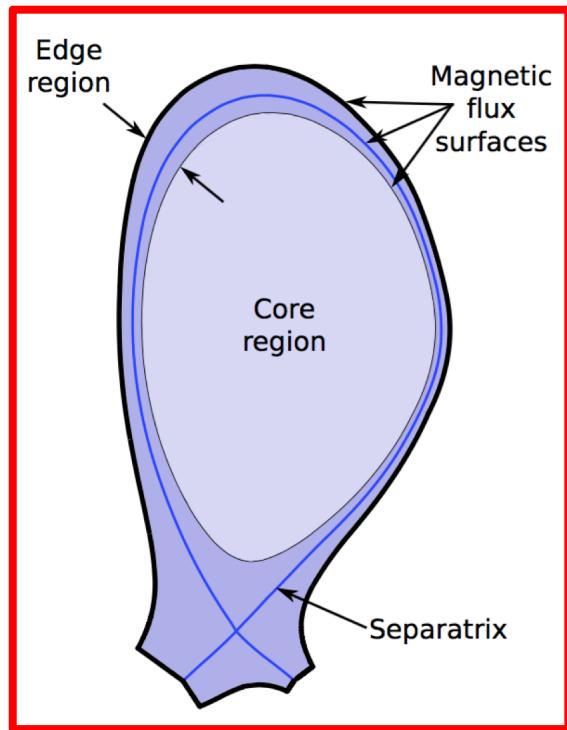


Image source: www.iter.org

ITER tokamak

Plasma dynamics in the edge region is characterized by a large range of temporal scales

Governing Equations

Full- f gyrokinetic Vlasov equation for each ion species

$$\underbrace{\frac{\partial B_{\parallel,\alpha}^* f_\alpha}{\partial t} + \nabla_{\mathbf{R}} \cdot (\dot{\mathbf{R}}_\alpha B_{\parallel,\alpha}^* f_\alpha)}_{\text{Vlasov}} + \underbrace{\frac{\partial}{\partial v_\parallel} (\dot{v}_{\parallel,\alpha} B_{\parallel,\alpha}^* f_\alpha)}_{\text{Collisions}} = \sum_\beta c [f_\alpha, f_\beta]$$

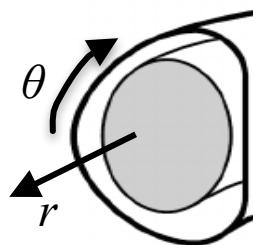
where $\dot{\mathbf{R}}_\alpha \equiv \dot{\mathbf{R}}_\alpha (\mathbf{R}, v_{\parallel}, \mu, t) = \frac{1}{B_{\parallel,\alpha}^*} \left[v_{\parallel} \mathbf{B}_\alpha^* + \frac{1}{Z_\alpha e} \hat{\mathbf{b}} \times (Z_\alpha e \mathbf{E} + \mu \nabla_{\mathbf{R}} B) \right]$ Velocity

$$\dot{v}_{\parallel,\alpha} \equiv \dot{v}_{\parallel,\alpha} (\mathbf{R}, v_{\parallel}, \mu, t) = -\frac{1}{m_\alpha B_{\parallel,\alpha}^*} \mathbf{B}_\alpha^* \cdot (Z_\alpha e \mathbf{E} + \mu \nabla_{\mathbf{R}} B)$$
 Acceleration

4D (2D-2V) phase space

$$\mathbf{R} \equiv \{r, \theta\}$$

$$v_{\parallel}, \mu = \frac{1}{2} \frac{m_{\alpha} v_{\perp}^2}{B}$$



Electric field \mathbf{E} can be specified or computed from f_α using the Poisson equation for electrostatic potential

We consider *single-species* cases in this study.

Fokker-Planck Collision Model

Fokker-Planck-Rosenbluth equation

$$c [f_\alpha, f_\beta] = \lambda_c \left(\frac{4\pi Z_\alpha Z_\beta e^2}{m_\alpha} \right)^2 \nabla_{(v_\parallel, \mu)} \cdot \left[\vec{\gamma}_\beta f_\alpha + \overleftrightarrow{\tau}_\beta \nabla_{(v_\parallel, \mu)} f_\alpha \right]$$

where the advective and diffusive coefficients are given by

$$\vec{\gamma}_\beta = \begin{bmatrix} \frac{\partial \varphi_\beta}{\partial v_\parallel} & 2\mu \frac{m_\beta}{B} \frac{\partial \varphi_\beta}{\partial \mu} \end{bmatrix}, \quad \overleftrightarrow{\tau}_\beta = \begin{bmatrix} -\frac{\partial^2 \varrho_\beta}{\partial v_\parallel^2} & -2\mu \frac{m_\beta}{B} \frac{\partial^2 \varrho_\beta}{\partial v_\parallel \partial \mu} \\ -2\mu \frac{m_\beta}{B} \frac{\partial^2 \varrho_\beta}{\partial v_\parallel \partial \mu} & -2\mu \left(\frac{m_\beta}{B}\right)^2 \left\{ 2\mu \frac{\partial^2 \varrho_\beta}{\partial \mu^2} + \frac{\partial \varrho_\beta}{\partial \mu} \right\} \end{bmatrix}$$

Rosenbluth potentials are related to f_β by the Poisson equations

$$\frac{\partial^2 \varphi_\beta}{\partial v_\parallel^2} + \frac{m_\beta}{B} \frac{\partial}{\partial \mu} \left(2\mu \frac{\partial \varphi_\beta}{\partial \mu} \right) = f_\beta$$

$$\frac{\partial^2 \varrho_\beta}{\partial v_\parallel^2} + \frac{m_\beta}{B} \frac{\partial}{\partial \mu} \left(2\mu \frac{\partial \varrho_\beta}{\partial \mu} \right) = \varphi_\beta$$



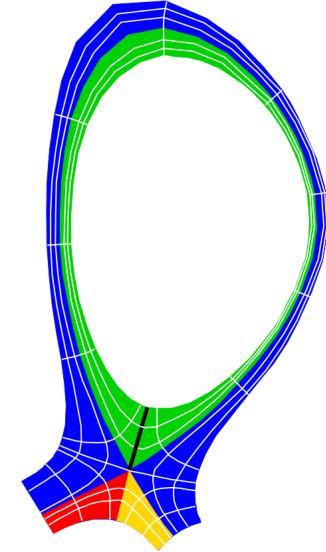
Non-linear, integro-differential term

Each evaluation of the Fokker-Planck term requires Poisson solve in the velocity space

COGENT: Spatial Discretization

Vlasov term

- Mapped, multi-block grids for complex geometries
- 4th order finite-volume discretization
- 5th order WENO scheme for reconstruction at cell faces
- *Colella, et al., J. Comput. Phys., 2011; McCorquodale, et al., J. Comput. Phys., 2015*



Fokker-Planck collision term

- Conservative finite-difference discretization on Cartesian velocity grid
- 5th order upwind discretization for the advective terms; 4th order central discretization for the diffusive terms
- Poisson equations for Rosenbluth potentials discretized using 2nd order central differences (*Dorf, et al., Contrib. Plasma Phys., 2014*)
- Energy-conserving modification implemented (*Taitano, et al., J. Comput. Phys., 2015*)

Time Integration

Spatial discretization
yields **semi-discrete**
ODE in time

Explicit time integration:
Runge-Kutta methods

$$\Delta t \left(\lambda \left[\frac{d\mathcal{R}(\tilde{f})}{d\tilde{f}} \right] \right) \in \{z : |R(z)| \leq 1\}$$

Time step constrained by eigenvalues
(time scales) of *entire RHS*

$$\frac{d\tilde{f}}{dt} = \mathcal{R}(\tilde{f}) \equiv \underbrace{\mathcal{V}(\tilde{f})}_{\text{Spatially-discretized Vlasov terms}} + \underbrace{\mathcal{C}(\tilde{f})}_{\text{collisions terms}}$$

Spatially-discretized **Vlasov** and **collisions** terms

Implicit-Explicit (IMEX) time integration:
Additive Runge-Kutta (ARK) methods

$$\mathcal{R}(\tilde{f}) = \underbrace{\mathcal{R}_{\text{stiff}}(\tilde{f})}_{\text{Implicit}} + \underbrace{\mathcal{R}_{\text{nonstiff}}(\tilde{f})}_{\text{Explicit}}$$


$$\Delta t \left(\lambda \left[\frac{d\mathcal{R}_{\text{nonstiff}}(\tilde{f})}{d\tilde{f}} \right] \right) \in \{z : |R(z)| \leq 1\}$$

IMEX: time step constrained by eigenvalues (time scales) of *stiff component of RHS*

Temporal Scales at Tokamak Edge

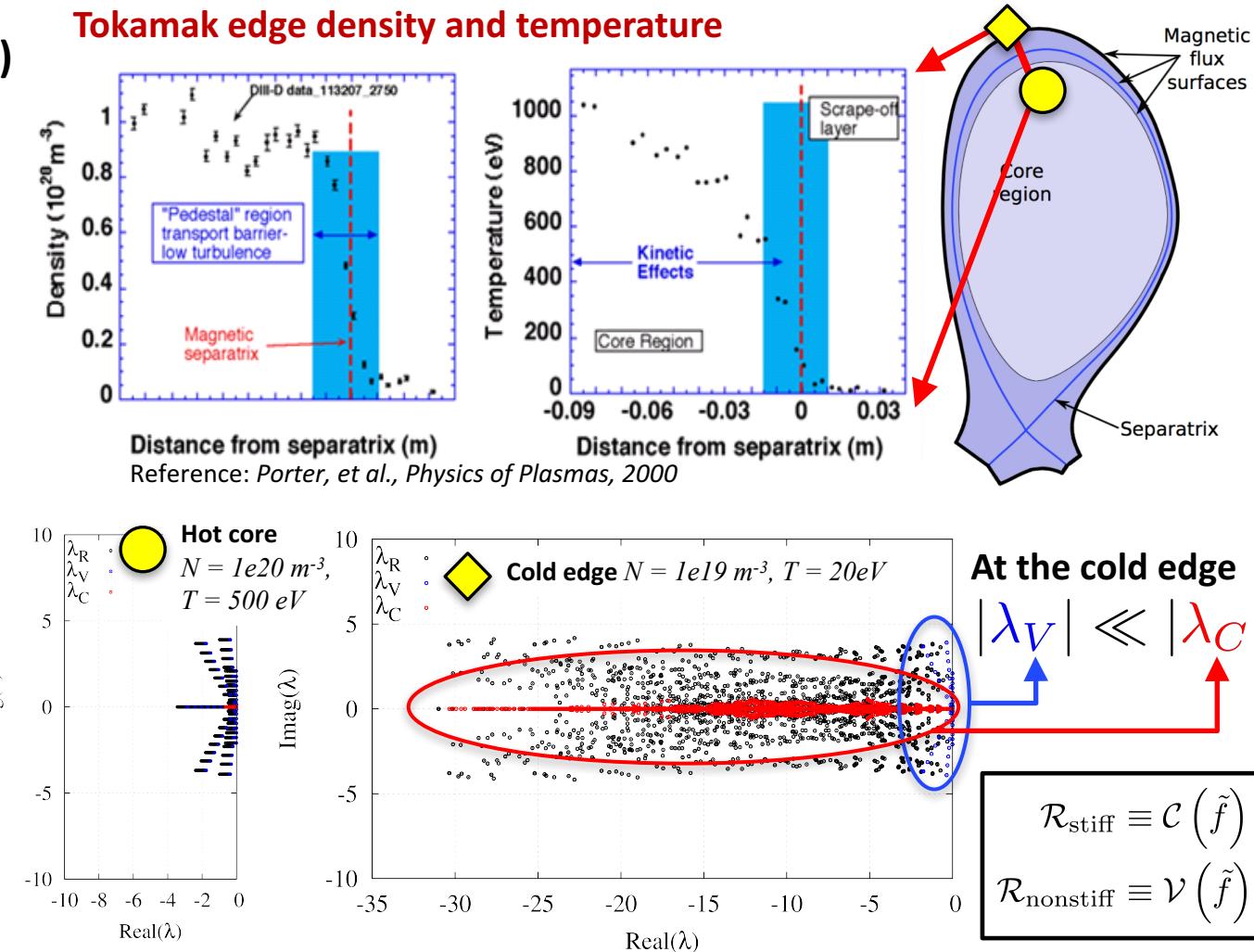
Eigenvalues (time scales)
of the entire RHS, and
the Vlasov and collision
terms separately

$$\lambda_R \equiv \lambda \left[\frac{d\mathcal{R}(\tilde{f})}{d\tilde{f}} \right]$$

$$\lambda_V \equiv \lambda \left[\frac{d\mathcal{V}(\tilde{f})}{d\tilde{f}} \right]$$

$$\lambda_C \equiv \lambda \left[\frac{d\mathcal{C}(\tilde{f})}{d\tilde{f}} \right]$$

Jacobians computed using
finite-differences on a very
small grid



Implicit-Explicit (IMEX) Time Integration

High-order, conservative multistage **Additive Runge-Kutta (ARK)** methods

Explicit method	0	0		
for non-stiff	c_2	a_{21}	0	
component	\vdots	\vdots	\ddots	0
(Explicit RK)	c_s	a_{s1}	\cdots	$a_{s,s-1}$
		b_1	\cdots	\cdots
		b_s		



Implicit method	0	0		
for stiff	\tilde{c}_2	\tilde{a}_{21}	γ	
component	\vdots	\vdots	\ddots	γ
(ESDIRK)	\tilde{c}_s	\tilde{a}_{s1}	\cdots	$\tilde{a}_{s,s-1}$
		b_1	\cdots	\cdots
				b_s

$s \rightarrow$ number of stages

Time step: $t_n \rightarrow t_{n+1} = t_n + \Delta t$

Stage solutions $\longrightarrow \tilde{f}^{(i)} = \tilde{f}_n + \Delta t \sum_{j=1}^{i-1} a_{ij} \mathcal{V}(\tilde{f}^{(j)}) + \Delta t \sum_{j=1}^i \tilde{a}_{ij} \mathcal{C}(\tilde{f}^{(j)}), i = 1, \dots, s$

Step completion $\longrightarrow \tilde{f}_{n+1} = \tilde{f}_n + \Delta t \sum_{i=1}^s b_i \left\{ \mathcal{V}(\tilde{f}^{(i)}) + \mathcal{C}(\tilde{f}^{(i)}) \right\}$

ARK2

- 2nd order, 3 stage
- *Giraldo, et al., SIAM J. Sci. Comput., 2013*

ARK3

- 3rd order, 4 stage
- *Kennedy & Carpenter, J. Comput. Phys., 2003*

ARK4

- 4th order, 6 stage
- *Kennedy & Carpenter, J. Comput. Phys., 2003*

Implicit Stage Solution

Implicit stages require the solution to a *nonlinear system of equations*

$$\underbrace{\frac{1}{\Delta t \tilde{a}_{ii}} \tilde{f}^{(i)} - \mathcal{C}(\tilde{f}^{(i)}) - \left[\tilde{f}_n + \Delta t \sum_{j=1}^{i-1} \left\{ a_{ij} \mathcal{V}(\tilde{f}^{(j)}) + \tilde{a}_{ij} \mathcal{C}(\tilde{f}^{(j)}) \right\} \right]}_{\mathcal{F}(y) = 0 \text{ where } y \equiv \tilde{f}^{(i)}} = 0$$

Jacobian-free Newton-Krylov method (*Knoll & Keyes, J. Comput. Phys., 2004*):

Initial guess: $y_0 \equiv \tilde{f}_0^{(i)} = \tilde{f}^{(i-1)}$

Newton update: $y_{k+1} = y_k - \mathcal{J}(y_k)^{-1} \mathcal{F}(y_k)$

GMRES solver:

$$\mathcal{J}(y_k) \Delta y = \mathcal{F}(y_k)$$

Action of the Jacobian on a vector
approximated by directional derivative

$$\mathcal{J}(y_k) x = \left. \frac{d\mathcal{F}(y)}{dy} \right|_{y_k} x \approx \frac{1}{\epsilon} [\mathcal{F}(y_k + \epsilon x) - \mathcal{F}(y_k)]$$

Preconditioning

Preconditioner

(sparse matrix)

$$\left[\alpha\mathcal{I} - \frac{d\bar{\mathcal{C}}(\tilde{f})}{d\tilde{f}} \right] \approx \left[\alpha\mathcal{I} - \frac{d\mathcal{C}(\tilde{f})}{d\tilde{f}} \right]$$

Exact Jacobian

(never assembled)

- Use lower order finite differences to construct the preconditioning matrix
- More sparse than the actual Jacobian
- Assembled and stored as a sparse matrix

$$\mathcal{C}(\tilde{f})$$

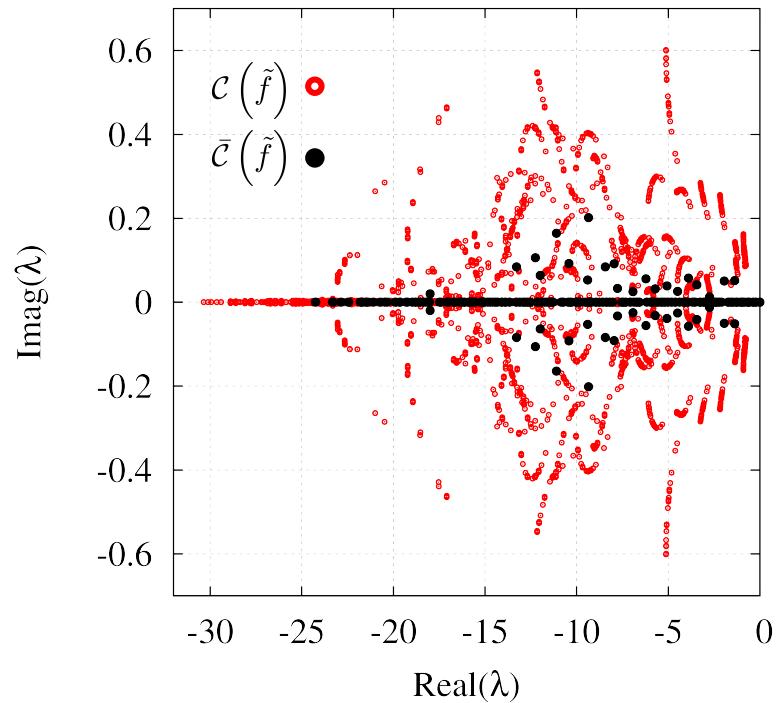
- 5th order upwind for advective terms
- 4th order central for diffusion terms

$$\bar{\mathcal{C}}(\tilde{f})$$

- 1st order upwind for advective terms
- 2nd order central for diffusion terms

→ Results in a 9-banded matrix

Eigenvalues of the Jacobian of the actual collisions term and the approximation for preconditioning



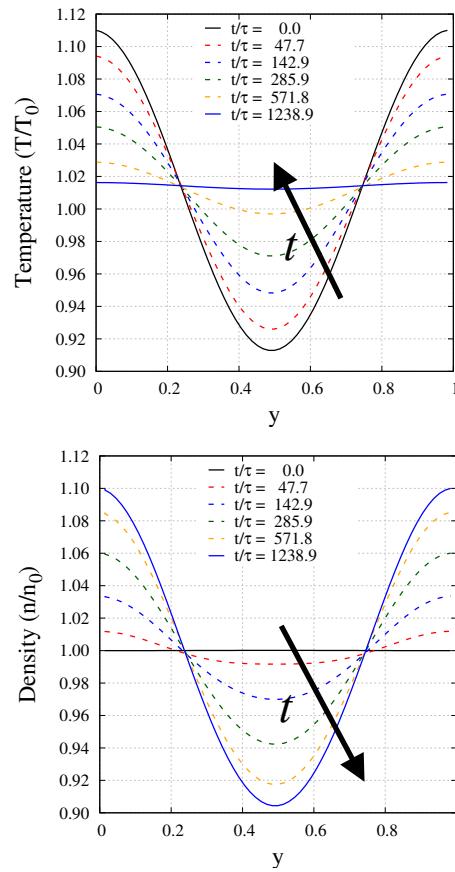
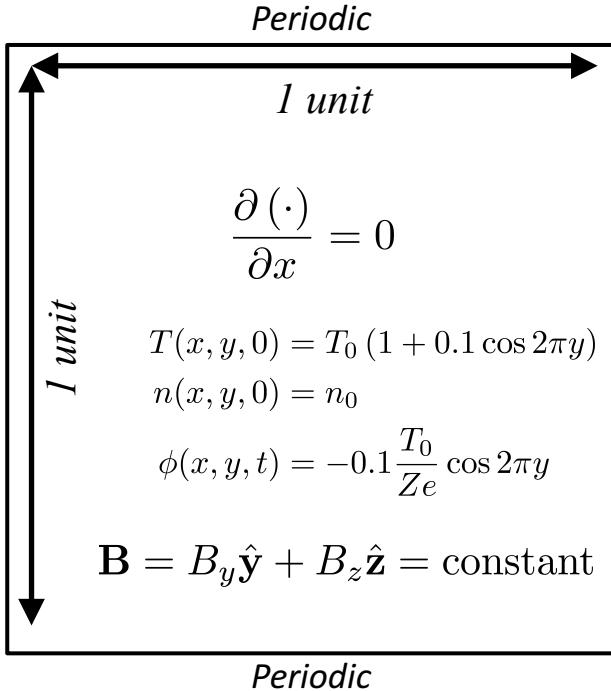
The preconditioner is inverted using the *Gauss-Seidel method* (computationally inexpensive)

Test Problem: Ion Parallel Heat Transport

A 2D slab (in configuration space), *representative of cold edge*

$$\left. \begin{array}{l} n = 10^{20} \text{ m}^{-3} \\ T = 20 \text{ eV} \end{array} \right\} k_{\parallel} \lambda = 0.065$$

Highly collisional

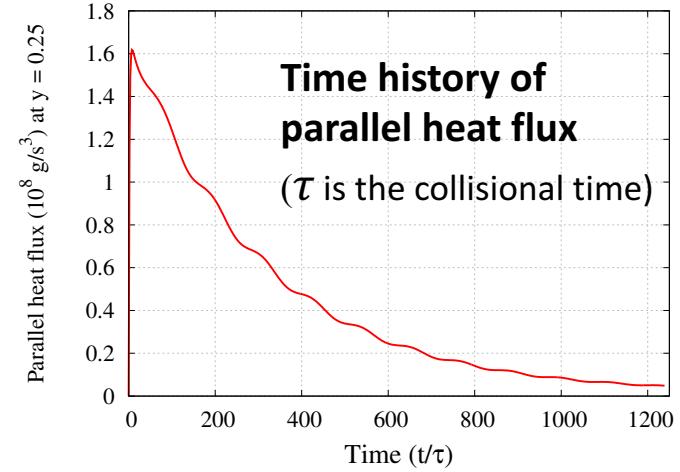


Transport time scale

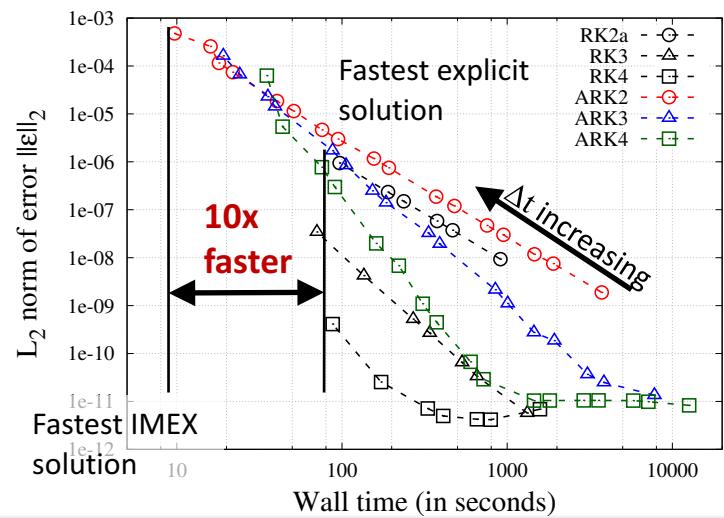
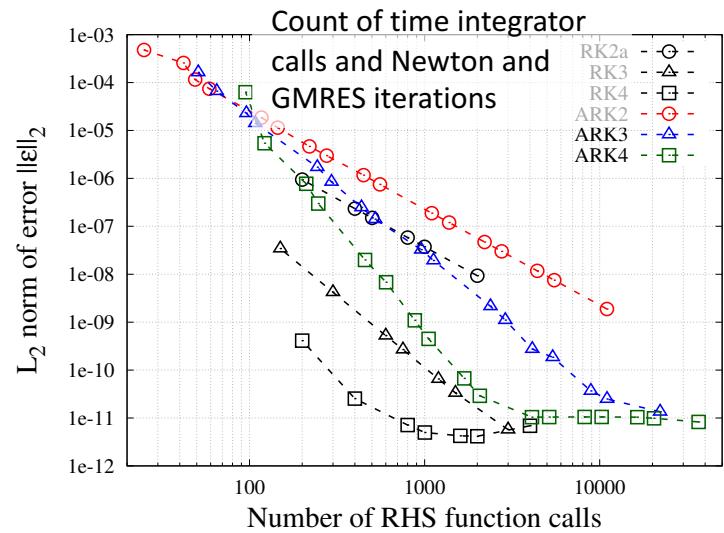
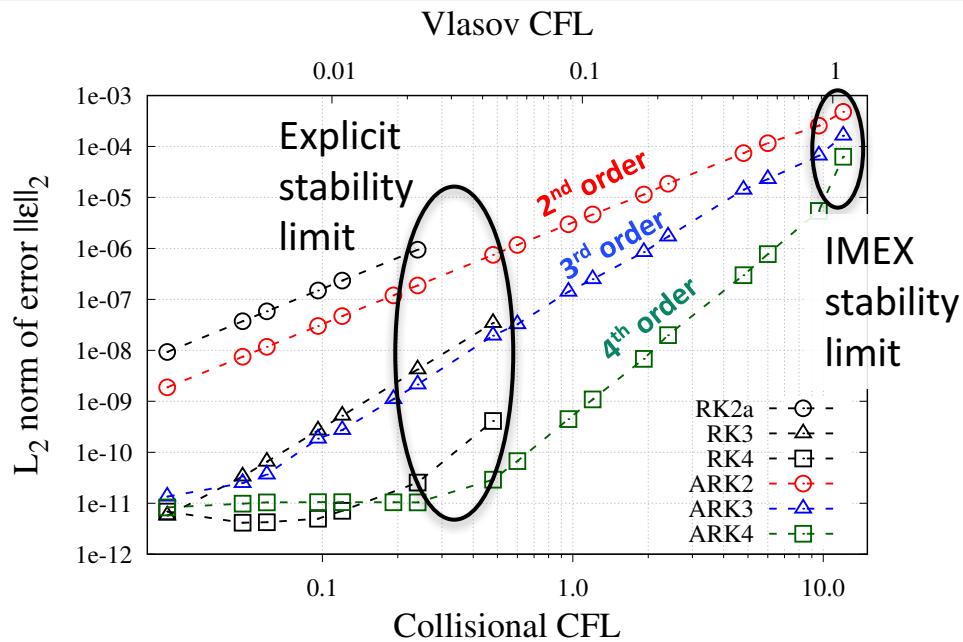
- Temperature equilibrates to constant value
- Density assumes cosine shape to balance electrostatic potential

Collisional time scale

- Heat flux attains values consistent with temperature gradient



Performance of IMEX Time Integrators



- Problem setup:
 - Grid size: $6 (x) \times 64 (y) \times 36 (v_{||}) \times 24 (\mu)$
 - Reference solution computed with RK4 using very low Δt
- IMEX methods achieve theoretical orders of convergence
- **Fastest stable solution:** IMEX schemes are 4x (ARK4) to 10x (ARK2) faster with 25x larger time steps.

Effect of Preconditioner

Computational cost of ARK4 with and without preconditioner

Gauss-Seidel solver with 80 iterations

Vlasov CFL	Collision CFL	Number of Function Calls *			Wall time (seconds)		
		No PC	With PC	Ratio	No PC	With PC	Ratio
0.04	0.5	2164	2067	0.96	711.33	719.1	1.01
0.11	1.2	991	879	0.89	321.5	307.9	0.96
0.22	2.4	586	457	0.78	190.5	161.2	0.85
0.55	6.0	355	211	0.59	123.1	75.5	0.61
1.10	12.0	190	95	0.50	63.3	35.1	0.55

* Number of function calls = Calls from time integrator (time steps × stages) + number of Newton iterations + number of GMRES iterations

- Preconditioner results in significant speed-up
- Overhead of assembling and inverting the preconditioning matrix is relatively small

Conclusions and Future Work

- IMEX approach for highly-collisional tokamak edge plasma
 - ✓ Collisions integrated in time implicitly while Vlasov term integrated in time explicitly
 - ✓ Wall time for fastest stable solution significantly reduced
 - ✓ Low order preconditioning results in lower computational cost at high collision CFL numbers
- Future work
 - More efficient solver for inverting the preconditioning matrix (Gauss-Seidel needs 80 iterations!)
 - Implement IMEX for other fast scales (*electrostatic Alfvén waves, parallel electron transport, ion acoustic modes, parallel ion transport*)

**Thank you.
Questions?**

