Discontinuous Galerkin Algorithms for Edge Gyrokinetics

Greg Hammett, Ammar Hakim, Eric Shi, Ian Abel\textsuperscript{2}, Tim Stoltzfus-Dueck\textsuperscript{3}
Princeton Plasma Physics Laboratory
\textsuperscript{2}Princeton Center For Theoretical Sciences, Princeton University
\textsuperscript{3}Princeton University

Joint Inst. For Fusion Theory Workshop, New Orleans, Oct. 31, 2014

Supported by the MPPC and a PPPL LDRD project.

(with small changes after talk)
Improving Confinement Can Significantly
↓ Size & Construction Cost of Fusion Reactor

Well known that improving confinement & $\beta$ can lower Cost of Electricity / kWh, at fixed power output.

Even stronger effect if consider smaller power: better confinement allows significantly smaller size/cost at same fusion gain $Q (nT\tau_E)$.

Standard H-mode empirical scaling:

$$\tau_E \sim H I_p^{0.93} P^{-0.69} B^{0.15} R^{1.97} \ldots$$

($P = 3 V n T / \tau_E$ & assume fixed $n T \tau_E, q_{95}, \beta_N, n/n_{\text{Greenwald}}$):

$$\$ \sim R^2 \sim 1 / (H^{4.8} B^{3.4})$$

ITER std $H=1$, steady-state $H \sim 1.5$
ARIES-AT $H \sim 1.5$
MIT ARC $H_{89}/2 \sim 1.4$

Need comprehensive simulations to make case for extrapolating improved $H$ to reactor scales.

(Plots assumes cost $\propto R^2$ roughly. Includes constraint on $R$ @ magnet with ARIES-AT 1.16 m blanket/shield, $a/R=0.25$, i.e. $B = B_{\text{mag}} (R-a/a_{BS})/R$. Neglects current drive issues.)
Edge region very difficult

Present core gyrokinetic codes are highly optimized for core, need new codes to handle additional complications of edge region of tokamaks (& stellarators):

open & closed field lines, plasma-wall-interactions, large amplitude fluctuations, (positivity constraints, non-Maxwellian full-F), atomic physics, non-axisymmetric RMP / stellarator coils, magnetic fluctuations near beta limit…

Hard problem: but success of core gyrokinetic codes makes me believe this is tractable, with a major initiative
Progress & Plans for Discontinuous Galerkin Gyrokinetic Code Gkeyll

• Developing new gyrokinetic code using advanced continuum/Eulerian algorithms (Discontinuous Galerkin, DG) that can help with the challenges of the edge region of fusion devices. Want to study edge problems like the height of the pedestal, suppression of ELMs, how much improvement can be made with lithium walls.

• Code or techniques could eventually be applied to a wider range of problems where kinetic effects become important, including astrophysics and non-plasma problems.

• Good progress:
  - Extensive tests in lower dimensions (Hamiltonian properties, parallel & perp dynamics of gyrokinetics, collisions), [http://www.ammar-hakim.org/sj/](http://www.ammar-hakim.org/sj/)
  - Demonstrated ability to handle magnetic fluctuations in an efficient way.
  - Now 2x+2v \((x, y, v_\parallel, v_\perp)\), including Lenard-Bernstein collision operator, logical sheath boundary conditions. Working towards full 3d+2V for gyrokinetics.
General goal: robust (gyro)kinetic code incorporating several advanced algorithms

- Several advanced algorithms (some in planning) to significantly improve efficiency:
  - A version of Discontinuous Galerkin (DG) algorithm can conserve energy exactly for Hamiltonian systems (even with upwinding, if implicit or $\Delta t$ small enough)
  - DG flexibility to handle optimized (Maxwellian-weighted) basis functions (Landreman JCP 2013: just 6 basis functions in v for accurate neoclassical theory)
  - sub-grid models in phase space
  - efficient use of massively parallel computers (GENE continuum code has excellent strong scaling to over 100,000 cores)

- DG: Efficient Gaussian integration --> ~ twice the accuracy / interpolation point:
  - Standard interpolation: $p$ uniformly-spaced points to get $p$ order accuracy
  - DG interpolates $p$ optimally-located points to get $2p-1$ order accuracy

- Kinetic turbulence very challenging, benefits from all tricks we can find. Potentially big win: Factor of 2 reduction in resolution --> 64x speedup in 5D gyrokinetics

Goal: a robust code applicable for a wider range of fusion and non-fusion problems, capable of relatively fast simulations at low velocity resolution, but with qualitatively-good gyro-fluid-like results, or fully converged kinetic results at high velocity resolution w/ massive computing.
Discontinuous Galerkin (DG) Combines Attractive Features of Finite-Volume & Finite Element Methods

Standard finite-volume (FV) methods evolve just average value in each cell (piecewise constant), combined with interpolations.

DG evolves higher-order moments in each cell. I.e. uses higher-order basis functions, like finite-element methods, but, allows discontinuities at boundary like shock-capturing finite-volume methods -->(1) easier flux limiters like shock-capturing finite-volume methods (preserve positivity) (2) calculations local so easier to parallelize.

Discontinuous Galerkin (DG) Combines Attractive Features of Finite-Volume & Finite Element Methods

Don’t get hung up on the word “discontinuous”. Simplest DG is piecewise constant: equivalent to standard finite volume methods that evolve just cell averaged quantities. Can reconstruct smooth interpolations between adjacent cells when needed.

Going to at least piecewise linear allows energy conservation (even with upwinding).

DG has ~ twice the accuracy per point of FV, by optimal spacing of points within cell.
Contributions Made to DG Algorithms

• First to note that a version of DG (based on C.-W. Shu & Liu, 2000) spatial discretization can exactly conserve energy for general Hamiltonian problems, $\frac{\partial f}{\partial t} = \{H, f\}$. (Time discretization considered separately.) Interestingly, does so even with upwind fluxes for $f$ --> limiters to preserve positivity, artificial oscillations.

• While we use DG for $f$, this energy conserving algorithm requires $H$ to be in continuous subspace of $f$ (i.e., standard finite elements for fields). Developed an extension that allows $H$ to be discontinuous also (preserves separability of gyrokinetic Poisson equation into independent 2D problems).

• Discovered improvements for diffusion terms $\frac{\partial^2 f}{\partial x^2}$ in widely used Local DG method. Instead use Recovery method by van Leer. (Not noticed before because it is a very transient initial error in diffusion equations.)

• Discovered a way to efficiently handle Alfven waves in DG, by using smoother basis functions for phi. Discovered a class of self-adjoint filter-projection operators that accomplishes this while avoiding global matrix inversion of a full projection operator.

• Flexibility of DG: plan to implement Maxwellian-weighted basis functions.

• Plan to implement subgrid models in both physical space and phase-space.
Simulated journal with extensive documentation of tests at [http://www.ammar-hakim.org/sj](http://www.ammar-hakim.org/sj)

Tested various features of perpendicular and parallel dynamics of gyrokinetic equations separately, and tested collisions. Now working to integrate together into a full 5D gyrokinetic code.

Results for various test problems:

* Incompressible Euler equations
* Hasegawa-Wakatani equations
* Vlasov-Poisson equations, linear Landau damping and nonlinear trapping
Simulation journal with extensive documentation of tests at http://www.ammar-hakim.org/sj

Hasegawa-Wakatani drift-wave turbulence
Gkeyll uses modern code architecture

• Gkeyll is written in C++ and is inspired by framework efforts like Facets, VORPAL (Tech-X Corporation) and WarpX (U. Washington). Uses structured grids with arbitrary dimension/order nodal basis functions.

• Linear solvers from Petsc\(^1\) are used for inverting stiffness matrices.

• Programming language Lua\(^2\), used as embedded scripting language to drive simulations. (Lua in widely played games like World of Warcraft, some iPhone apps, ...)

• MPI is used for parallelization via the txbase library developed at Tech-X

• Package management and builds are automated via scimake and bilder, both developed at Tech-X.

• (I am beginning to explore Julia / iJulia for postprocessing: [http://julialang.org](http://julialang.org). New high-level language oriented to scientific programming being developed at MIT. Fast, parallelization, ...)

\(^1\) [http://www.mcs.anl.gov/petsc/](http://www.mcs.anl.gov/petsc/)

\(^2\) [http://www.lua.org](http://www.lua.org)
Test Problem Geometry

- ELM crash simulated as a source of plasma at the midplane
- Target plates at edges of symmetric computational domain, midplane in the center
- Evolve plasma and calculate heat flux vs. time at target plates

First done by Pitts (2007), widely used test case (Havlickova, Fundamenski et al. (2012), Omatani & Dudson, 2013, ...)

Use gyrokinetic equations: keeping not just parallel dynamics, but also perpendicular ion polarization in GK quasineutrality / vorticity equation.

Don’t have to resolve Debye length (use sheath boundary conditions), much faster:

\[-\partial_\perp \left( \frac{n_i m_i}{B^2} \partial_\perp \phi \right) = e (n_i - n_e)\]

(using simplified lower bound on $k_\perp^2$ at first.)
Gkeyll can now Model ELM Heat Pulse in 1D SOL

Simulation of ELM pulse to divertor plate on JET agrees well with full PIC and Vlasov codes (Pitts, 2007, Havlickova, Fundamenski et al. 2012). Confirms sheath potential rises to shield divertor from initial electron heat pulse.

30,000x faster than full PIC because gyrokinetics doesn’t have to resolve Debye length.

![Graphs showing comparison between Full PIC, 1D Vlasov, and Gkeyll simulations of ELM heat pulse](image)

(small differences because initial conditions not precisely specified.)
Simplest Alfven Wave in Gyrokinetics

\[ \frac{\partial f_e}{\partial t} + v_\parallel \frac{\partial f_e}{\partial z} + \frac{q_e}{m_e} \left( -\frac{\partial \phi}{\partial z} - \frac{\partial A_\parallel}{\partial t} \right) \frac{\partial f_e}{\partial v_\parallel} = 0 \]

\[ -n_i k_\perp^2 \rho_s^2 \frac{e\phi}{T_{e0}} = \int d v_\parallel f_e - n_i \]

\[ k_\perp^2 A_\parallel = \mu_0 q_e \int d v_\parallel f_e v_\parallel \]

If \( \omega \gg k_\parallel v_{te} \), this gives:

\[ \omega^2 = \frac{k_\parallel^2 v_{te}^2}{k_\parallel^2 \rho_s^2 / \beta_e} \]

where \( \beta_e = (\beta_e/2)(m_i/m_e) \). The electrostatic case \( A_\parallel = 0 \) corresponds to the low \( \beta \) limit, where there is an \( \Omega_H \) mode that is even faster than electrons at low \( k_\perp \):

\[ \omega^2 = \frac{k_\parallel^2 v_{te}^2 / \beta_e}{1 + k_\perp^2 \rho_s^2 / \beta_e} \rightarrow \frac{k_\parallel^2 v_{te}^2}{k_\perp^2 \rho_s^2} \]

It would seem that finite \( \beta \) should be easier because it limits the fastest wave at low \( k_\perp \) to be no faster than the Alfven wave
Handling the $\partial A_{||}/\partial t$ term

$$\frac{\partial f_e}{\partial t} + v_{||} \frac{\partial f_e}{\partial z} + \frac{q_e}{m_e} \left( -\frac{\partial \phi}{\partial z} - \frac{\partial A_{||}}{\partial v_{||}} \right) \frac{\partial f_e}{\partial v_{||}} = 0$$

Codes usually eliminate the $\partial A_{||}/\partial t$ term with the substitution $\delta f_e = g + (q_e/m_e)A_{||}\partial F_{0}/\partial v_{||}$ (or by going to $p_{||} = m v_{||} + q_e A_{||}$ coordinates, which is equivalent linearly). Ampere’s law become:

$$\left( k_{\perp}^2 + C_n \frac{\mu_0 q_e^2}{m_e} \int dp_{||} f_e \right) A_{||} = C_j \mu_0 \frac{q_e}{m_e^2} \int dp_{||} f_e p_{||}$$

$$C_n \omega_{pe}^2/c^2$$

“The Ampere Cancellation Problem”: the ratio of the first to second term is very small, $k_{\perp}^2 \rho_s^2/\hat{\beta} \sim 10^{-5}$, for $k_{\perp} \rho_s = 0.01$ and $\beta_e \sim 1\%$. Small errors (represented by $C_n$ or $C_j \neq 1$) in large terms can have a large effect:

If $\omega \gg k_{||} v_{te}$:

$$\omega^2 = \frac{k_{||}^2 v_{te}^2}{k_{\perp}^2 \rho_s^2} \left[ \frac{k_{||}^2 \rho_s^2 + (C_n - C_j)\beta_e}{k_{||}^2 \rho_s^2 + C_n\beta_e} \right]$$

We were first to note a version of the Discontinuous Galerkin (DG) algorithm can exactly conserve energy for general Hamiltonian problems $\partial f / \partial t = \{ H, f \}$. (Based on algorithm by C.-W. Shu and Liu, 2000.) Requires $H$ (and thus $\phi$ and $A_{||}$) to be in a continuous subspace of $f$.

In the MHD limit, we need

$$E_{||} = -\frac{\partial \phi}{\partial z} - \frac{\partial A_{||}}{\partial t} \approx 0$$

but there is no way for a continuous $A_{||}(z)$ to offset this discontinuous $\partial \phi / \partial z$.

This electrostatic field drives a current that is a square wave, and wants to make a square wave $A_{||}(z)$. But projection of this square wave $A_{||}$ onto a continuous subspace gives $A_{||} = 0$, as if $\beta = 0$. This gives very high frequency mode at grid scale, requiring a very small time step $\Delta t < k_{||, \text{max}} v_{te} / (k_{\perp, \text{min}} \rho_s)$. 

(x in these figures should be z)
Fix for magnetic fluctuations for DG

There are several solutions. One is to project $\phi(z)$ onto a $C_1$ subspace where $\phi$ and $\partial \phi / \partial z$ are continuous. ($\phi$ must be at least piecewise-parabolic in this case.) This allows a $C_0 A_{||}(z, t)$ to better approximate the ideal MHD condition $E_{||} \approx 0 = -\partial \phi / \partial z - \partial A_{||} / \partial t$. Allows Gkeyll to reproduce Alfven wave even at very low $k_{\perp} \rho_s$ with a normal time step.

In order to conserve energy, the projection operator must be self-adjoint. We have found a local filtering/projection operator that is self-adjoint.
Progress & Plans for Discontinuous Galerkin Gyrokinetic Code Gkeyll

- Developing new gyrokinetic code using advanced continuum/Eulerian algorithms (Discontinuous Galerkin, DG) that can help with the challenges of the edge region of fusion devices. Want to study edge problems like the height of the pedestal, suppression of ELMs, how much improvement can be made with lithium walls.

- Code or techniques could eventually be applied to a wider range of problems where kinetic effects become important, including astrophysics and non-plasma problems.

- Good progress:
  - Extensive tests in lower dimensions (Hamiltonian properties, parallel & perp dynamics of gyrokinetics, collisions), [http://www.ammar-hakim.org/sj/](http://www.ammar-hakim.org/sj/)
  - Demonstrated ability to do 1D SOL Test problem of ELM on JET, 30,000x faster than full orbit (non-gyrokinetic) PIC code
  - Demonstrated ability to handle magnetic fluctuations in an efficient way.
  - Now 2x+2v \((x, y, v_\parallel, v_\perp)\), including Lenard-Bernstein collision operator, logical sheath boundary conditions. Working towards full 3d+2V for gyrokinetics.