Multiple timescale calculations of sawteeth and other global macroscopic dynamics of tokamaks

Stephen C. Jardin[1]
N.M. Ferraro[2,3], J. Breslau[1], J. Chen[1],

[1]*Princeton Plasma Physics Laboratory*
[2] *General Atomics*
[3]*Oak Ridge Institute for Science and Education, Oak Ridge TN*

International Conference on Numerical Simulations of Plasmas

Long Branch, NJ
Sept. 9, 2011

This work was performed in close collaboration with M. Shephard, F. Zhang, and F. Delalondre at the SCOREC center at Rensselaer Polytechnic Institute in Troy, NY.

Acknowledgements also to: M. Chance, G. Fu, S. Hudson, H. Strauss, L. Sugiyama

PPPL
PRINCETON PLASMA PHYSICS LABORATORY

Summary and Overview:

- 3D MHD equations are a mixed system: hyperbolic + parabolic
  - This leads to multiple timescales

- The hyperbolic terms are associated with ideal MHD wave propagation and global instabilities.
  - These are the shortest timescales: typically micro-seconds

- The parabolic terms are associated with diffusion and transport of the magnetic field, current, pressures, and densities
  - These are the longest timescales: typically 100s of milliseconds

- To calculate both phenomena in a single simulation requires a highly implicit formulation so that the time step is determined by accuracy requirements only
  - not by numerical stability requirements such as Courant condition

- The implicit solution procedure is complicated by the fact that the multiple timescales present in the physics lead to a very ill-conditioned matrix equation that needs to be solved each time step.
  - Here we describe the techniques we use to deal with this in M3D-C^1

2-Fluid 3D MHD Equations:

$$\frac{\partial n}{\partial t} + \nabla \bullet (n\mathbf{V}) = 0$$ continuity

$$\frac{\partial \mathbf{B}}{\partial t} = -\nabla \times \mathbf{E} \qquad \nabla \bullet \mathbf{B} = 0 \qquad \mu_0 \mathbf{J} = \nabla \times \mathbf{B}$$ Maxwell

$$nM_i\left(\frac{\partial \mathbf{V}}{\partial t} + \mathbf{V} \bullet \nabla \mathbf{V}\right) + \nabla p = \mathbf{J} \times \mathbf{B} - \nabla \bullet \mathbf{\Pi}_{GV} - \nabla \bullet \mathbf{\Pi}_{\mu}$$ momentum

$$\mathbf{E} + \mathbf{V} \times \mathbf{B} = \eta \mathbf{J} + \frac{1}{ne}\left(\mathbf{J} \times \mathbf{B} - \nabla p_e - \nabla \bullet \mathbf{\Pi}_e\right)$$ Ohm's law

$$\frac{3}{2}\frac{\partial p_e}{\partial t} + \nabla \bullet \left(\frac{3}{2} p_e \mathbf{V}\right) = -p_e \nabla \bullet \mathbf{V} + \eta J^2 - \nabla \bullet \mathbf{q}_e + Q_\Delta$$ electron energy

$$\frac{3}{2}\frac{\partial p_i}{\partial t} + \nabla \bullet \left(\frac{3}{2} p_i \mathbf{V}\right) = -p_i \nabla \bullet \mathbf{V} - \mathbf{\Pi}_\mu \bullet \nabla \mathbf{V} - \nabla \bullet \mathbf{q}_i - Q_\Delta$$ ion energy

Ideal MHD

Resistive MHD

2-fluid MHD

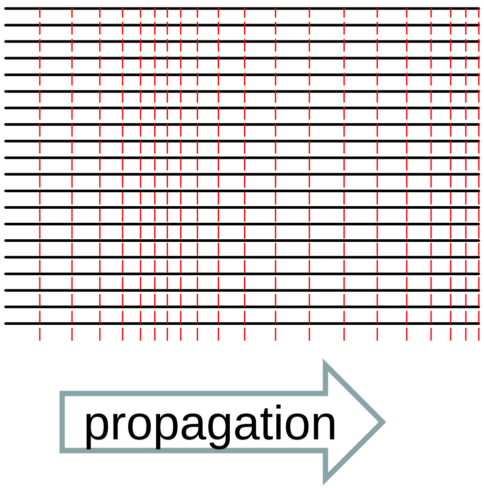
The objective of the **M3D-*C*[1]** project is to solve these equations as accurately as possible in 3D toroidal geometry with realistic B.C. and optimized for a low-β torus with a strong toroidal field.

Contain ideal MHD, reconnection, and transport timescales

$$\tau_I \ll \tau_R \ll \tau_T$$

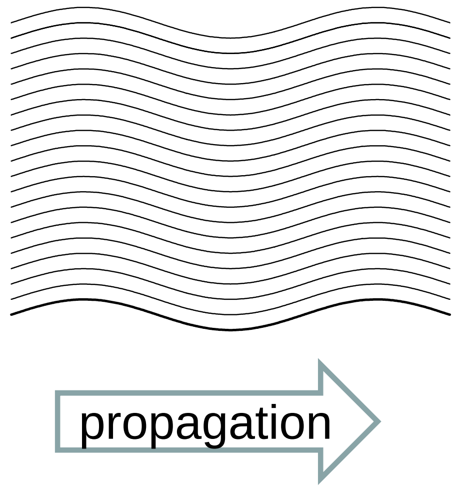
Three types of wave solutions in ideal MHD

Slow Wave



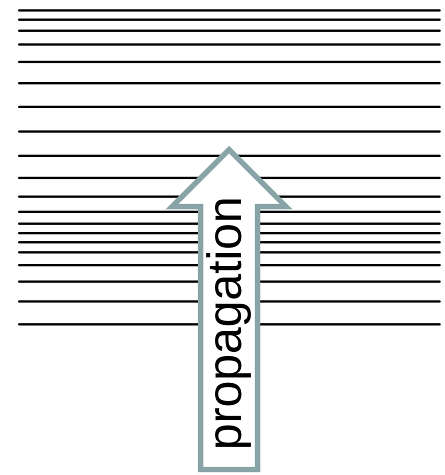
- only propagates parallel to **B**
- only compresses fluid in parallel direction
- does not perturb magnetic field

Alfven Wave



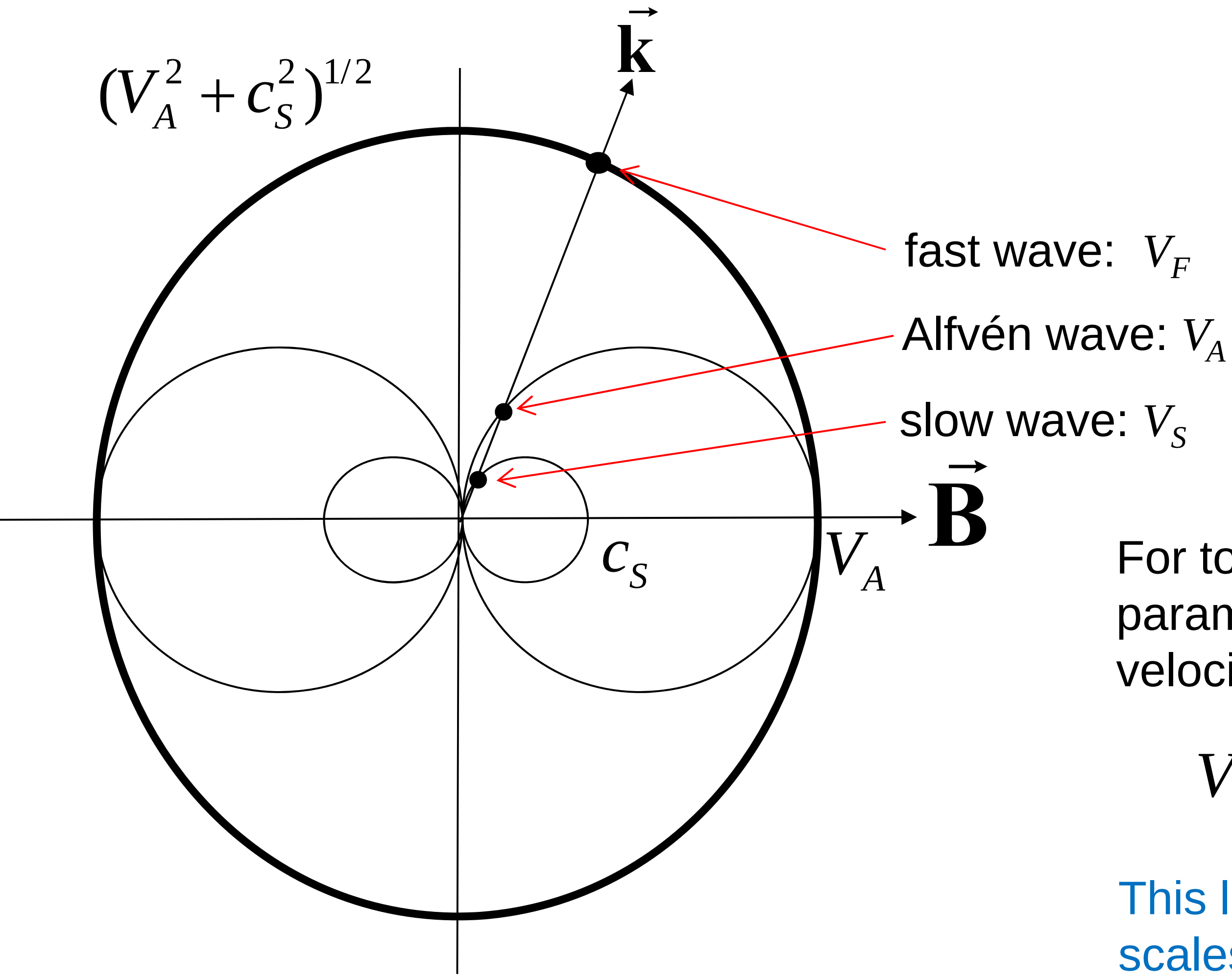
- only propagates parallel to **B**
- incompressible
- only bends the field, does not compress it

Fast Wave



- can propagate perpendicular to **B**
- only compresses fluid in $\perp$ direction
- compresses the magnetic field
- **This is the wave that makes equations stiff!**

The three ideal MHD waves have widely separate velocities for propagation with $\mathbf{k}\bullet\mathbf{B} \sim 0$



Wave speed diagram for ideal MHD. Intersection points show wave velocity for given propagation direction.

For tokamak geometry and parameters, the three wave velocities satisfy the inequalities:

$$V_F \gg V_A \gg V_S$$

This leads to multiple time-scales, even within ideal MHD

Implicit solution requires evaluating the spatial derivatives at the new time level.

The advantage of an implicit solution is that the time step can be very large and still be numerically stable (no Courant condition)

If we discretize in space (finite difference, finite element, or spectral) and linearize the equations about the present time level, the implicit equations take the form:

$$\begin{bmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} & \mathbf{A}_{13} \\ \mathbf{A}_{21} & \mathbf{A}_{22} & \mathbf{A}_{23} \\ \mathbf{A}_{31} & \mathbf{A}_{32} & \mathbf{A}_{33} \end{bmatrix} \bullet \begin{bmatrix} \mathbf{V} \\ \mathbf{B} \\ p \end{bmatrix}^{n+1} = \begin{bmatrix} \mathbf{r}_1 \\ \mathbf{r}_2 \\ \mathbf{r}_3 \end{bmatrix}^{n}$$

Very large, ~ (10^7 x 10^7)
non-diagonally dominant,
non-symmetric, ill-conditioned sparse matrix (contains all MHD waves)

How best to solve this?

3 step physics-based preconditioner greatly improves iterative solve

(1) Split implicit formulation

(2) Apply annihilation operators

(3) Apply block-Jacobi preconditioner by using SuperLU_dist on each poloidal plane independently

Original matrix multiplying
V^{n+1}, **B**$^{n+1}$, p^{n+1}
- non-symmetric,
- non-diagonally dominant &
- large range of eigenvalues

GMRES

Smaller matrix multiplying **V**$^{n+1}$ only,
- nearly symmetric
- closer to diagonal
- still with large range of eigenvalues

Matrix now consists of 3 dominant diagonal blocks, each with narrower range of eigenvalues.

Now, range of eigenvalues in each block is greatly reduced.

Preconditioned system converges in 10’s of iterations

3 step physics-based preconditioner greatly improves iterative solve

(1) Split implicit formulation

(2) Apply annihilation operators

(3) Apply block-Jacobi preconditioner by using SuperLU_dist on each poloidal plane independently

Original matrix multiplying
V^{n+1}, **B**$^{n+1}$, p^{n+1}
- non-symmetric,
- non-diagonally dominant &
- large range of eigenvalues

→ → → → GMRES

Smaller matrix multiplying **V**$^{n+1}$ only,
- nearly symmetric
- closer to diagonal
- still with large range of eigenvalues

Matrix now consists of 3 dominant diagonal blocks, each with narrower range of eigenvalues.

Now, range of eigenvalues in each block is greatly reduced.

Preconditioned system converges in 10's of iterations

(1) Split implicit formulation eliminates $\mathbf{B}^{n+1}$ and p^{n+1} in favor of $\mathbf{V}^{n+1}$

As an example, consider the simple 1D wave equation for velocity V and pressure p

$$\left.\begin{aligned}\frac{\partial V}{\partial t} &= c\frac{\partial p}{\partial x}\\ \frac{\partial p}{\partial t} &= c\frac{\partial V}{\partial x}\end{aligned}\right\}\quad \frac{\partial^2 V}{\partial t^2} - c^2\frac{\partial^2 V}{\partial x^2} = 0$$

Implicit FD time advance evaluates spatial derivatives at the new time level

$$\frac{V_j^{n+1} - V_j^n}{\delta t} = c\left(\frac{p_{j+1/2}^{n+1} - p_{j-1/2}^{n+1}}{\delta x}\right)$$

$$\frac{p_{j+1/2}^{n+1} - p_{j+1/2}^n}{\delta t} = c\left(\frac{V_{j+1}^{n+1} - V_j^{n+1}}{\delta x}\right)$$

(1) Split implicit formulation eliminates $\mathbf{B}^{n+1}$ and p^{n+1} in favor of $\mathbf{V}^{n+1}$

As an example, consider the simple 1D wave equation for velocity V and pressure p

$$\left.\begin{aligned} \frac{\partial V}{\partial t} &= c\frac{\partial p}{\partial x} \\ \frac{\partial p}{\partial t} &= c\frac{\partial V}{\partial x} \end{aligned}\right\} \quad \frac{\partial^2 V}{\partial t^2} - c^2\frac{\partial^2 V}{\partial x^2} = 0$$

Implicit FD time advance evaluates spatial derivatives at the new time level

$$\frac{V_j^{n+1} - V_j^n}{\delta t} = c\left(\frac{\boxed{p_{j+1/2}^{n+1}} - p_{j-1/2}^{n+1}}{\delta x}\right)$$

$$\frac{\boxed{p_{j+1/2}^{n+1}} - p_{j+1/2}^{n}}{\delta t} = c\left(\frac{V_{j+1}^{n+1} - V_j^{n+1}}{\delta x}\right)$$

Now, algebraically eliminate new time pressure in favor of velocity

$$V_j^{n+1} = V_j^n + (\delta t c)^2\left(\frac{V_{j+1}^{n+1} - 2V_j^{n+1} + V_{j-1}^{n+1}}{\delta x^2}\right) + \delta t c\left(\frac{p_{j+1/2}^n - p_{j-1/2}^n}{\delta x}\right) \quad \longleftarrow$$

Symmetric & diagonally dominant!

$$p_{j+1/2}^{n+1} = p_{j+1/2}^n + \frac{\delta t c}{\delta x}\left(V_{j+1}^{n+1} - V_j^{n+1}\right)$$

These equations will give exactly the same answers, but can be solved sequentially!

Schematic of difference in matrices to be inverted after applying split implicit formulation

$$
2N \quad \begin{bmatrix}
1 & -S & & & & & & & & & & \\
S & 1 & -S & & & & & & & & & \\
 & S & 1 & -S & & & & & & & & \\
 & & S & 1 & -S & & & & & & & \\
 & & & S & 1 & -S & & & & & & \\
 & & & & S & 1 & -S & & & & & \\
 & & & & & S & 1 & -S & & & & \\
 & & & & & & S & 1 & -S & & & \\
 & & & & & & & S & 1 & -S & & \\
 & & & & & & & & S & 1 & -S & \\
 & & & & & & & & & S & 1 & -S \\
 & & & & & & & & & & S & 1
\end{bmatrix}
\quad \Rightarrow \quad
N \quad \begin{bmatrix}
1+2S^2 & -S^2 & & & & \\
-S^2 & 1+2S^2 & -S^2 & & & \\
 & -S^2 & 1+2S^2 & -S^2 & & \\
 & & -S^2 & 1+2S^2 & -S^2 & \\
 & & & -S^2 & 1+2S^2 & -S^2 \\
 & & & & -S^2 & 1+2S^2
\end{bmatrix}
$$

$$
N \quad \begin{bmatrix}
1 & & & & & \\
 & 1 & & & & \\
 & & 1 & & & \\
 & & & 1 & & \\
 & & & & 1 & \\
 & & & & & 1
\end{bmatrix}
\qquad S = \frac{c\delta t}{\delta x} \gg 1
$$

Coupled system multiplying V^{n+1} & p^{n+1}

Un-coupled system multiplying V^{n+1} & p^{n+1} separately

Substitution takes us from having to invert a 2N x 2N anti-symmetric system that has large off-diagonal elements to sequentially inverting a N x N symmetric system that is diagonally dominant + the identity matrix.

Mathematically equivalent → same answers! (but much better conditioned)

3 step physics-based preconditioner greatly improves iterative solve

(1) Split implicit formulation

(2) Apply annihilation operators

(3) Apply block-Jacobi preconditioner by using SuperLU_dist on each poloidal plane independently

Original matrix multiplying

$\mathbf{V}^{n+1}$, $\mathbf{B}^{n+1}$, p^{n+1}

- non-symmetric,
- non-diagonally dominant &
- large range of eigenvalues

→ [ ] → [[ ] [ ] [ ]] → [[ ] [ ] [ ]] → GMRES

Smaller matrix multiplying $\mathbf{V}^{n+1}$ only,

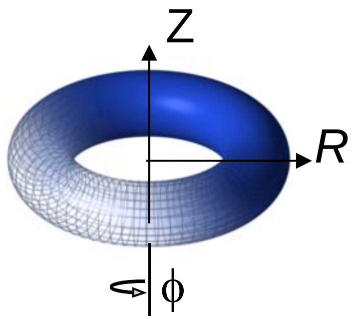
- nearly symmetric
- closer to diagonal
- still with large range of eigenvalues

Matrix now consists of 3 dominant diagonal blocks, each with narrower range of eigenvalues.

Now, range of eigenvalues in each block is greatly reduced.

Preconditioned system converges in 10's of iterations

(2) Apply annihilation operators to separate eigenvalues into diagonal blocks



$$\nabla_\perp \equiv \hat{R}\frac{\partial}{\partial R} + \hat{Z}\frac{\partial}{\partial Z}$$

Velocity vector written in terms of 3 scalar fields:

$$\mathbf{V} = R^2 \nabla U \times \nabla \phi + R^2 \omega \nabla \phi + \frac{1}{R^2} \nabla_\perp \chi$$

Associated mainly with the shear Alfven wave: does not compress the toroidal field

Associated mainly with the slow wave: also does not compress the toroidal field

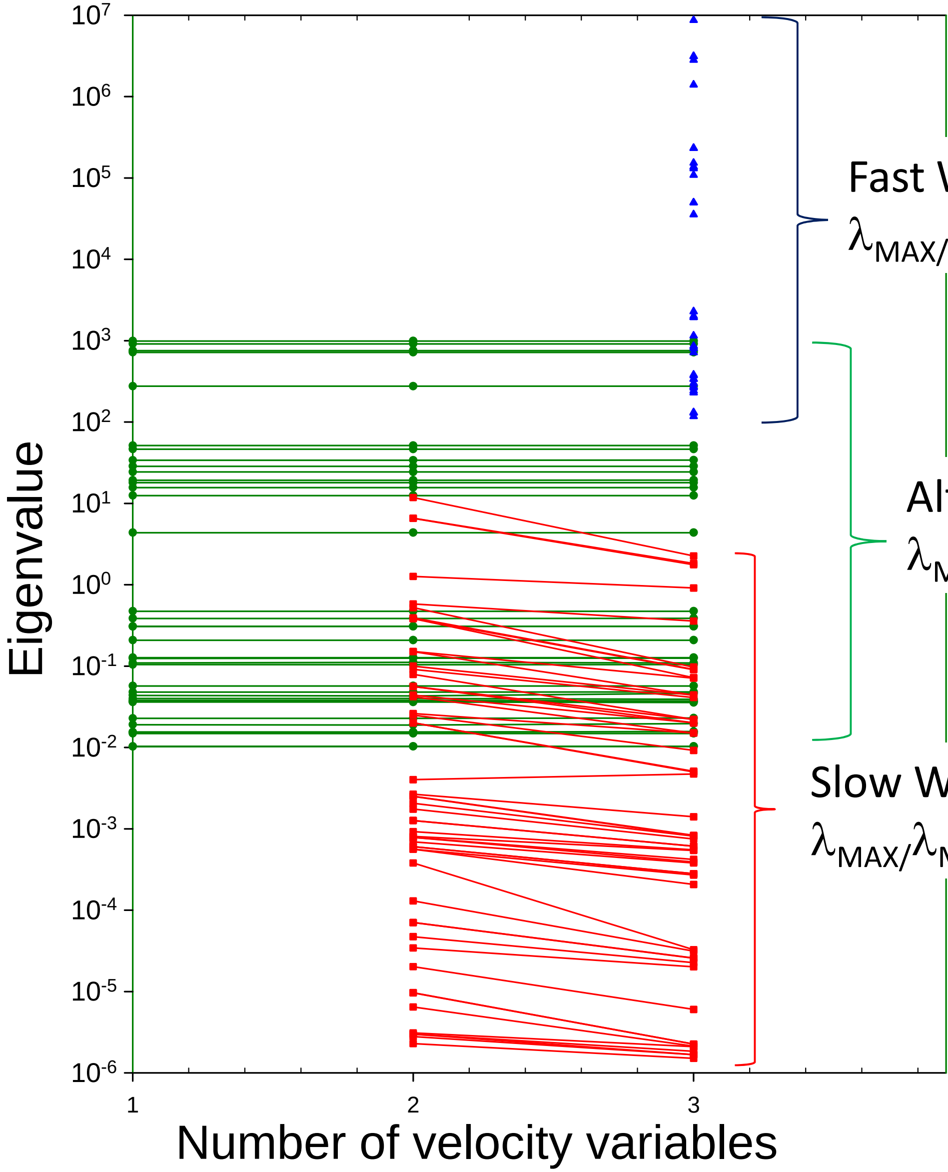
Associated mainly with the fast wave: does compress the toroidal field

To obtain scalar equations, we apply annihilation projections to isolate the physics associated with the different wave types in different blocks in the matrix

$$\left.\begin{array}{ll} \text{Alfven wave:} & \nabla \varphi \bullet \nabla_\perp \times R^2 \\ \text{slow wave:} & R^2 \nabla \varphi \bullet \\ \text{fast wave:} & -\nabla_\perp \bullet R^{-2} \end{array}\right\} \frac{\partial \mathbf{V}}{\partial t} + \mathbf{V} \bullet \nabla \mathbf{V} = \frac{1}{nM_i}\left[-\nabla p + \mathbf{J} \times \mathbf{B} - \nabla \bullet \mathbf{\Pi}_{GV} - \nabla \bullet \mathbf{\Pi}_{\mu}\right]$$

Code can be run with 1,2 (reduced MHD) or 3 (full MHD) velocity variables

M3D-C^1 can be run with 1, 2, or 3 velocity variables. Tracking the eigenvalues shows how they separate into 3 groups

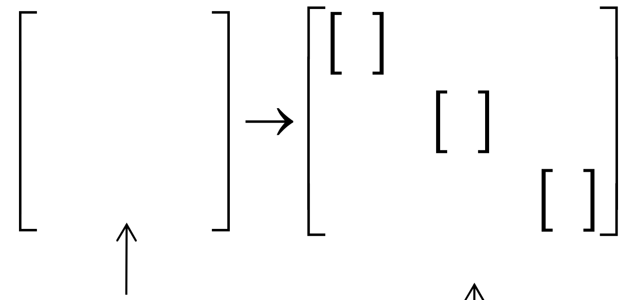


U $\quad$ $U+\omega$ $\quad$ $U+\omega+\chi$

Fast Wave
$\lambda_{MAX}/\lambda_{MIN}$ =7.3 E4

Alfven Wave
$\lambda_{MAX}/\lambda_{MIN}$ =9.9 E4

Slow Wave
$\lambda_{MAX}/\lambda_{MIN}$ =1.46 E6



Original matrix:
$\lambda_{MAX}/\lambda_{MIN} = 10^{13}$

Annihilation operators split this into 3 loosely connected matrices, each with a condition number much less than original matrix

3 step physics-based preconditioner greatly improves iterative solve

(1) Split implicit formulation

(2) Apply annihilation operators

(3) Apply block-Jacobi preconditioner by using SuperLU_dist on each poloidal plane independently

Original matrix multiplying
V^{n+1}, **B**$^{n+1}$, p^{n+1}
- non-symmetric,
- non-diagonally dominant &
- large range of eigenvalues

GMRES

Smaller matrix multiplying **V**$^{n+1}$ only,
- nearly symmetric
- closer to diagonal
- still with large range of eigenvalues

Matrix now consists of 3 dominant diagonal blocks, each with narrower range of eigenvalues.

Now, range of eigenvalues in each block is greatly reduced.

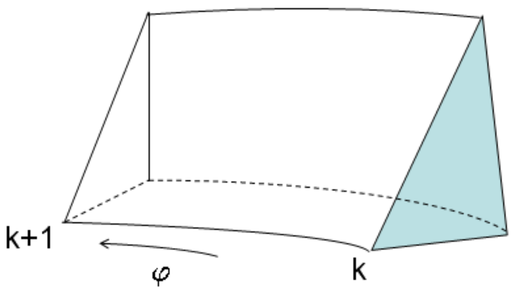
Preconditioned system converges in 10's of iterations

(3) Apply block-Jacobi preconditioner by using SuperLU_dist on each poloidal plane independently

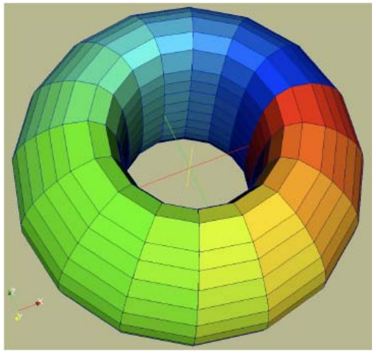
M3D-C^1 uses a triangular wedge high order finite element

- Continuous 1st derivatives in all directions … *C^1 continuity*
- Unstructured triangles in (R,Z) plane
- Structured in φ

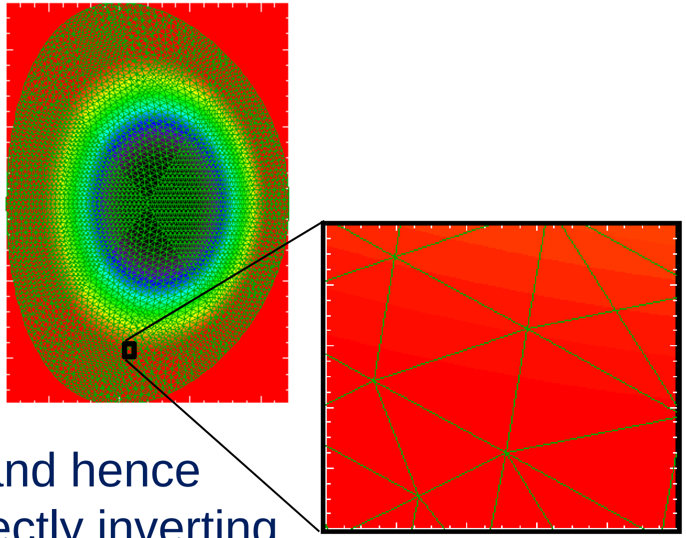
Triangular wedge integration volume



Top view: 16-32 toroidal prisms



Slice view: ~ 10^4 nodes/plane



Because of the small zone size within the plane, and hence strong coupling, we precondition the matrix by directly inverting the components within each poloidal plane simultaneously.

Block Jacobi Preconditioner: reduces condition number by $\left(\frac{\Delta x_{\varphi}}{\Delta x_{R,Z}}\right)^2 \sim 4000$

(3) Apply block-Jacobi preconditioner by using SuperLU_dist on each poloidal plane independently (cont)

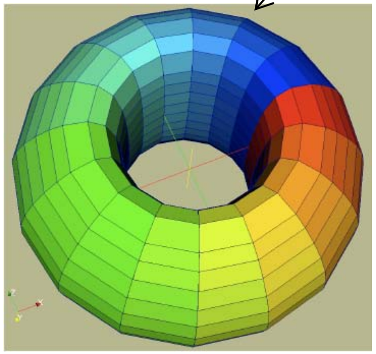
- All the nodes on each poloidal plane are coupled only to their nearest neighbors. This leads to block triangular structure

$$\begin{bmatrix} \mathbf{B}_1 & \mathbf{C}_1 & & & & & & \mathbf{A}_1 \\ \bullet & \bullet & \bullet & & & & & \\ & \bullet & \bullet & \bullet & & & & \\ & & \mathbf{A}_j & \mathbf{B}_j & \mathbf{C}_j & & & \\ & & & \bullet & \bullet & \bullet & & \\ & & & & \bullet & \bullet & \bullet & \\ & & & & & \bullet & \bullet & \bullet \\ \mathbf{C}_N & & & & & & \mathbf{A}_N & \mathbf{B}_N \end{bmatrix} \bullet \begin{bmatrix} \mathbf{V}_1 \\ . \\ \mathbf{V}_{j-1} \\ \mathbf{V}_j \\ \mathbf{V}_{j+1} \\ . \\ . \\ \mathbf{V}_N \end{bmatrix}^{n+1} = \begin{bmatrix} \mathbf{y}_1 \\ . \\ \mathbf{y}_{j-1} \\ \mathbf{y}_j \\ \mathbf{y}_{j+1} \\ . \\ . \\ \mathbf{y}_N \end{bmatrix}^{n}$$

$\mathbf{A}_j, \mathbf{B}_j, \mathbf{C}_j$

are 2D sparse matrices at plane j

$\mathbf{V}_j$ denotes all the velocity variables on plane j



Block Jacobi preconditioner corresponds to multiplying each row by inverse of diagonal block $\mathbf{B}_j^{-1}$

PETSc now has the capability of doing this using SuperLU_Dist concurrently on each plane

Transport Timescale simulations in which stability is important: with $\Delta t = 40\ \tau_A$

Resistivity: $\eta = n^{3/2} p^{-3/2}$

Thermal Conductivity: $\kappa_\perp = n^{3/2} p^{-1/2}$

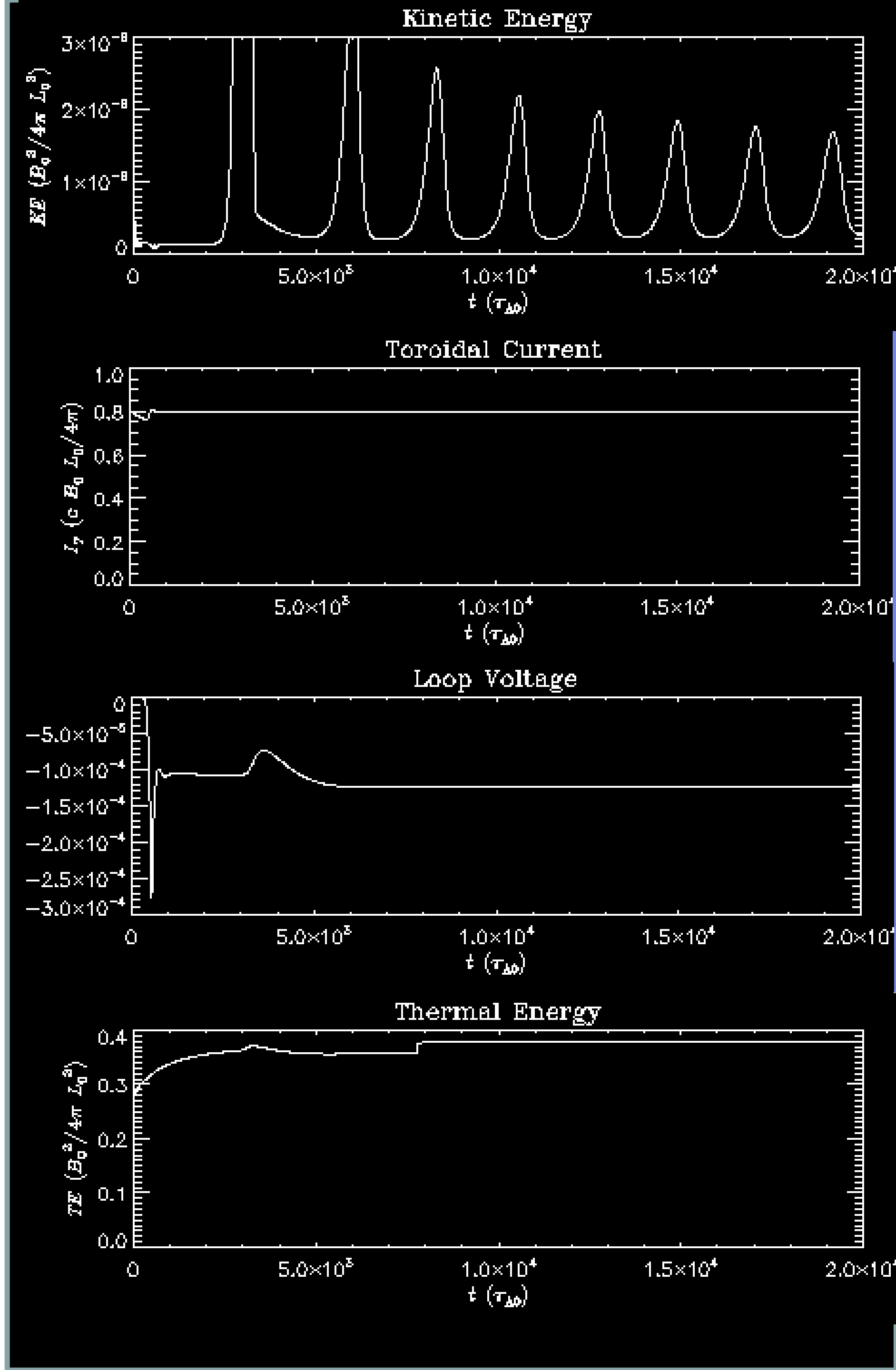
$$\kappa_\parallel = 10^6 \kappa_\perp$$

Viscosity: uniform (~η)

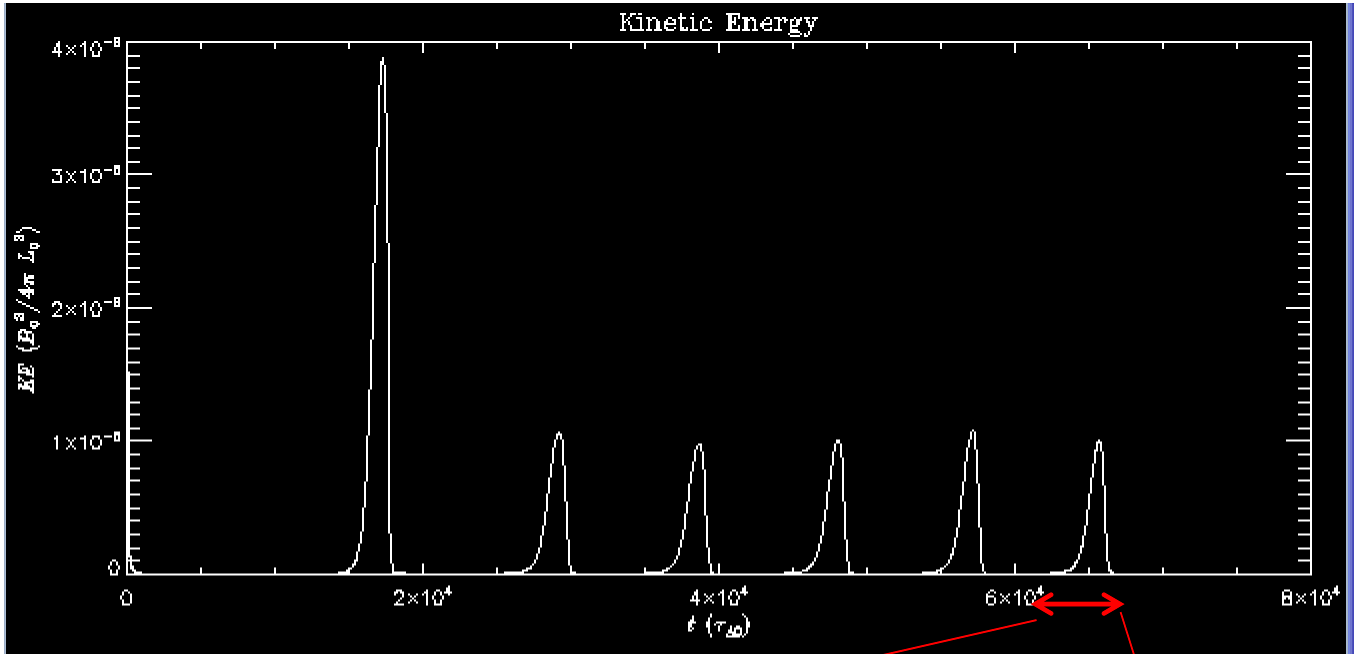
Current controller provides loop voltage to maintain plasma current at initial value.

Loop voltage provides thermal energy through Ohmic heating

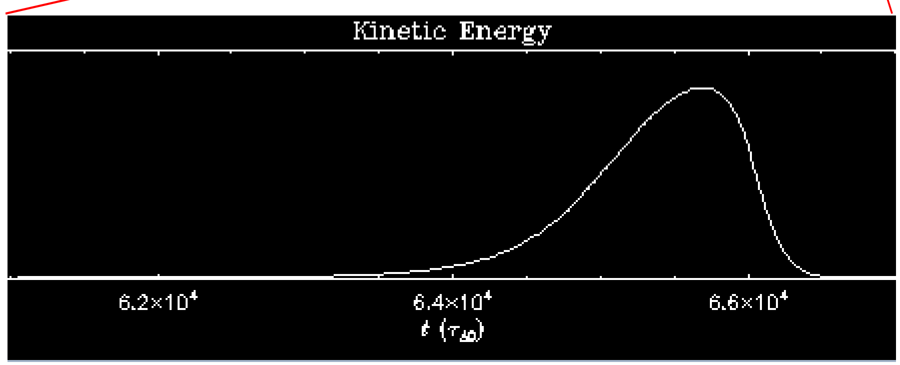
Current density periodically peaks, becomes unstable, reconnects, and broadens…periodic cycle



Typical result: 1st sawtooth event depends on initial conditions.
After many events, system reaches steady-state or periodic behavior

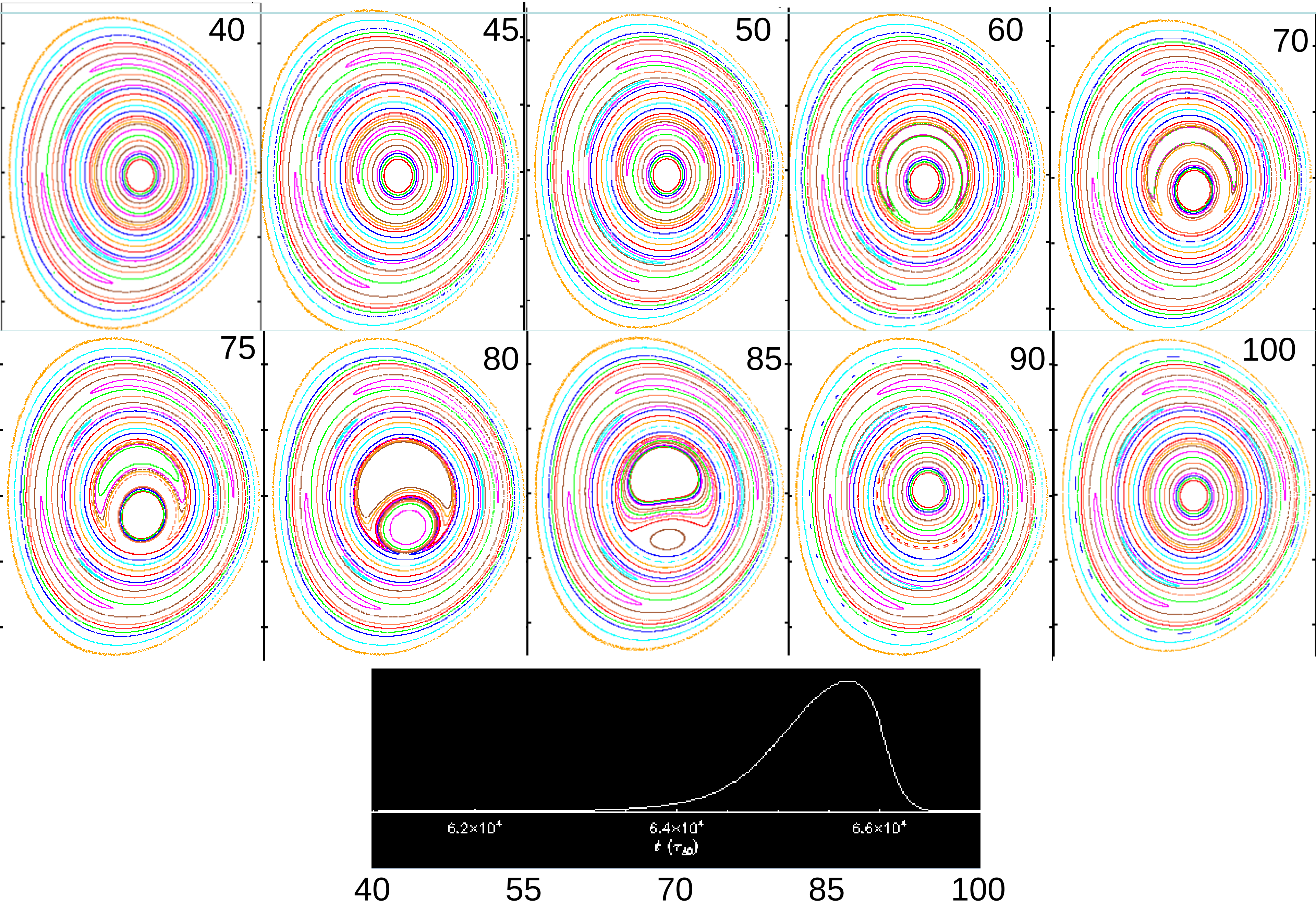


Repeating
sawtooth cycle

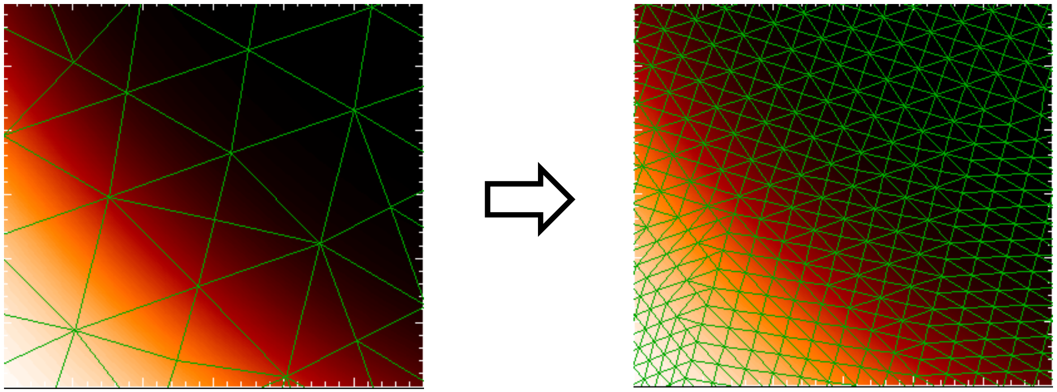


Precursor phase Crash phase

Poincare plots during a single sawtooth cycle

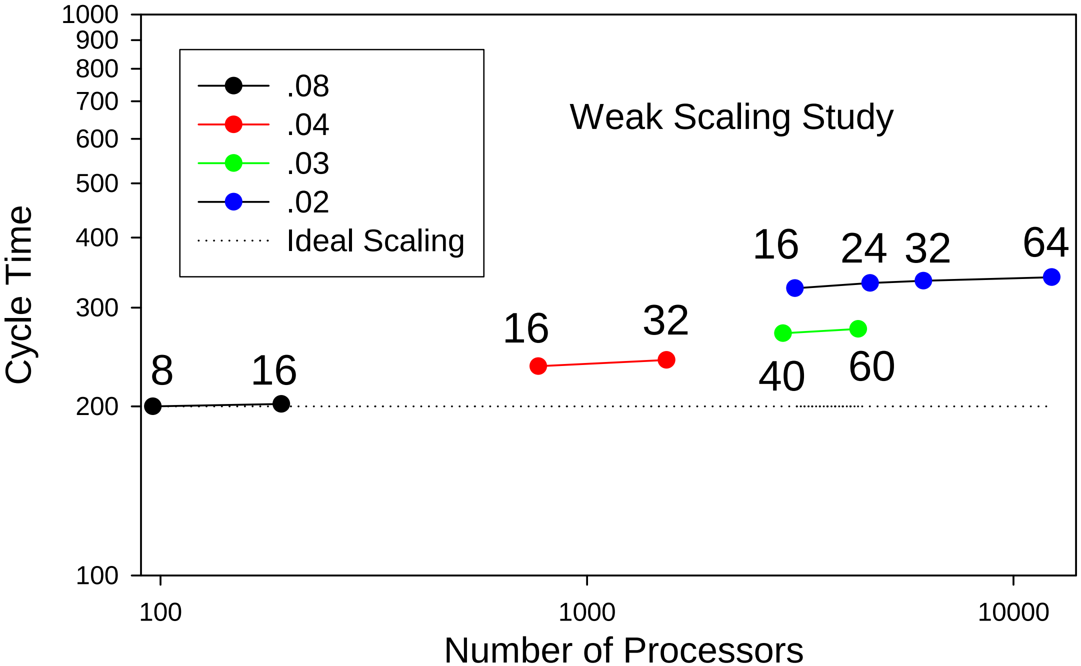


Parallel Scaling Studies have been performed from 96 to 12288 p



.08 .02

Triangle linear dimension varied by factor of 4



Number of toroidal planes varied from 8 to 64

Summary

- 3D MHD in a highly magnetized high temperature plasma
  - Multiple timescales (ideal, reconnection, transport) demand implicit time advance
  - Implicit matrix contains large range of eigenvalues associated with the 3 different MHD wave types
- 3-step physics based preconditioner employed
  - Split implicit method reduces matrix size by 2 and makes matrix near symmetric and diagonally dominant
  - Annihilation operators approximately split matrix into 3 diagonal blocks, each with a greatly reduced condition number
  - Block Jacobi preconditioner dramatically reduces the condition number of each of the diagonal blocks
  - Final preconditioned matrix given to GMRES converges in 10s of iterations for fine zoned problem
- Recent Results
  - Repeating sawtooth demonstrate multiple timescale calculations