Recent Algorithmic and Computational Efficiency Improvements in the NIMROD Code

S. J. Plimpton, *Sandia National Laboratory*
C. R. Sovinec and T. A. Gianakon, *Los Alamos National Laboratory*
S. E. Parker, *University of Colorado-Boulder*
and The NIMROD Team

presented at the

41st Annual Meeting of the American Physical Society, Division of Plasma Physics, November 15-19, 1999

OUTLINE

1) Introduction
   a) Code and Team descriptions
   b) Algorithm features
   c) Equations
   d) Challenges

2) Matrix Solution
   a) Global preconditioning for CG
   b) Linking AZTEC

3) Anisotropic thermal conduction

4) Generalized basis elements

5) Strong flow
   a) Time advance
   b) Divergence condition

6) Modeling kinetic effects

NIMROD

Non-Ideal MHD with Rotation
Open Discussion Project

The NIMROD Code:

- Developed to simulate nonlinear electromagnetic phenomena that fundamentally alter the magnetic confinement of fusion-grade plasmas.
- Designed to handle the stiff mathematical equations that describe hot magnetized plasmas, where time and spatial scales vary by many orders of magnitude.
- Implemented in a parallel algorithm to run on the fastest computers available today and in the future.
- NIMROD is publicly available from **http://nimrodteam.org**

The NIMROD Team:

- NIMROD is a multi-institutional project commissioned by the Department of Energy's Office of Fusion Energy Science. The product is publicly available to provide the greatest possible benefit to the fusion community.
- The NIMROD Team presently includes members from SAIC-San Diego, SNL, UT-Austin, General Atomics, UCLA, CU-Boulder, MSU, and LANL.

PRESENT TEAM MEMBERS AND ADVISORS

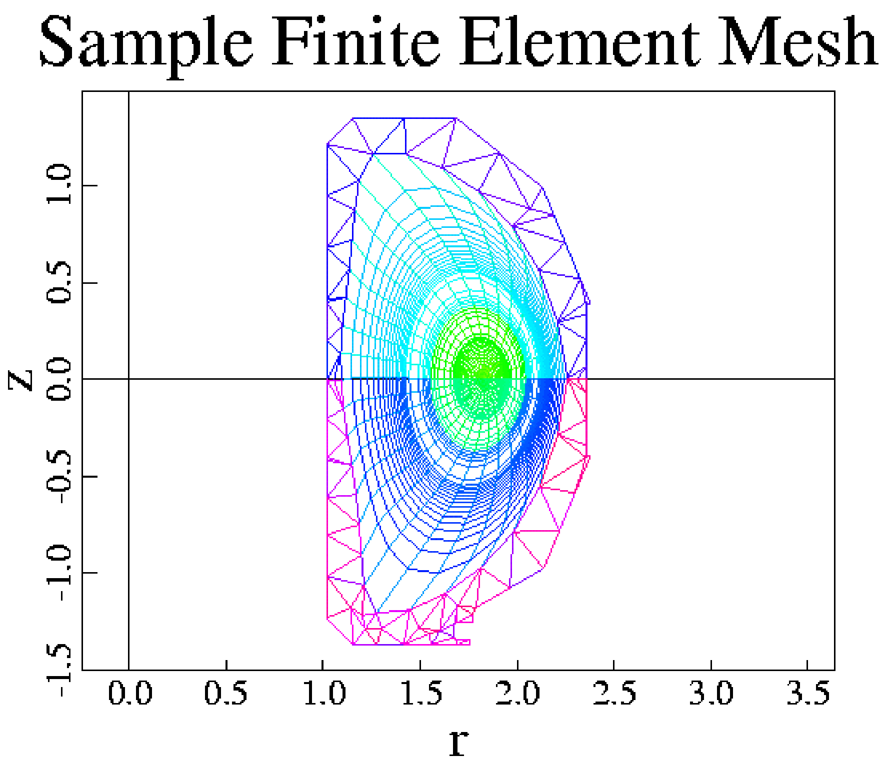
Ahmet Aydemir	IFS
James Callen	U-WI
Ming Chu	GA
John Finn	LANL
Tom Gianakon	LANL
Charlson Kim	CU-Boulder
Scott Kruger	SAIC
Jean-Noel Leboeuf	UCLA
Richard Nebel	LANL
Scott Parker	CU-Boulder
Steve Plimpton	SNL
Nina Popova	MSU
Dalton Schnack	SAIC
Carl Sovinec	LANL
Alfonso Tarditi	SAIC

NIMROD FEATURES

The NIMROD spatial representation is finite element for one plane and Fourier series for the third (periodic) direction.

- Permits simulating detailed configurations for comparison with experiment and for simulating non-fusion applications.

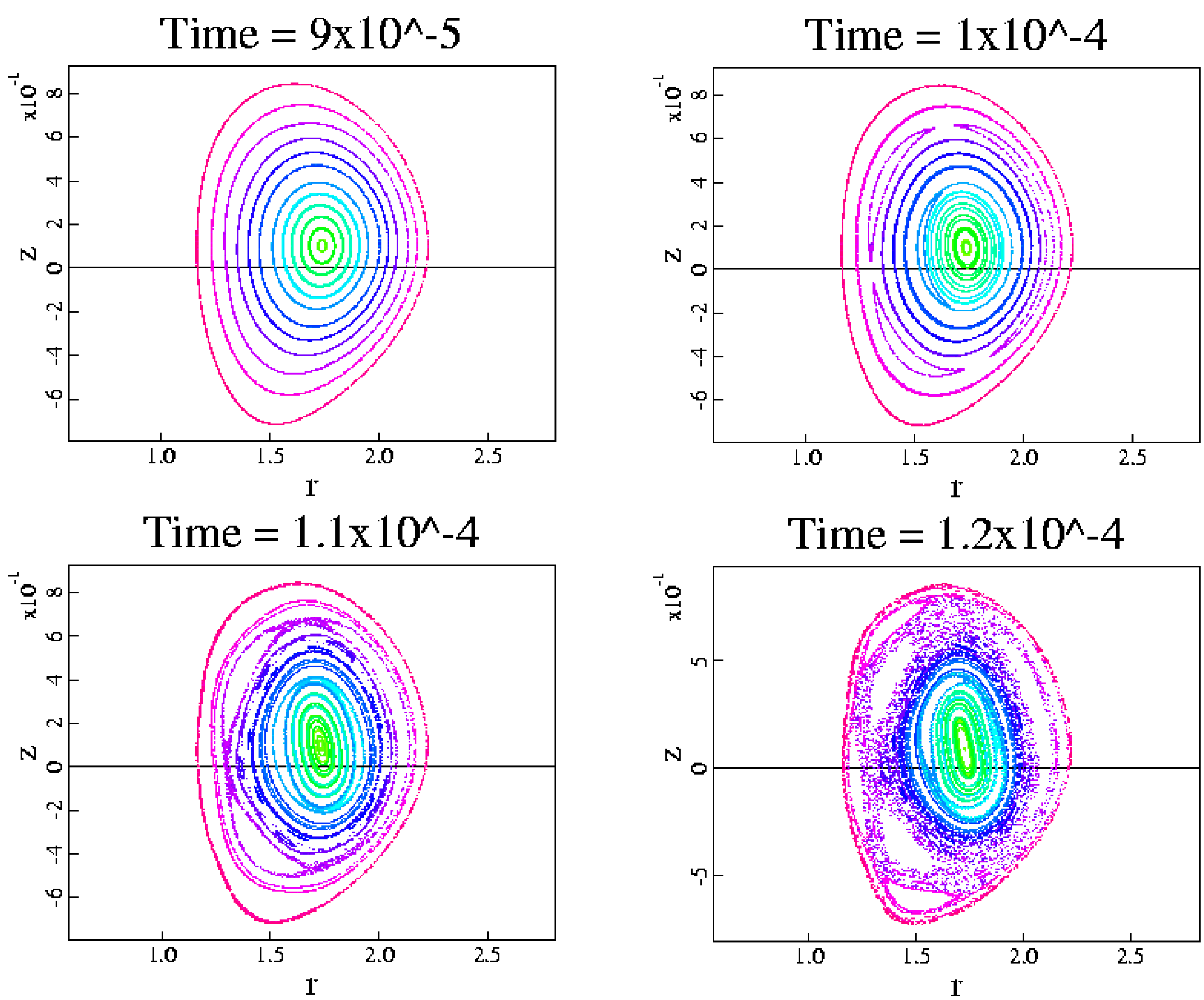
- Allows refinement in regions of interest.



- Present and future developments will take advantage of the flexibility of finite elements to change basis functions and to dynamically adapt the mesh to solutions.

Our time-split, semi-implicit advance of the coupled Maxwell/velocity-moment (electrons and ions) set of equations permits efficient parallel computation of slow, nonlinear electromagnetic phenomena.

- The following magnetic field Poincare plots show magnetic topology changing gradually over ~300 wave transit times around the toroidal configuration of the DIII-D tokamak at General Atomics in San Diego. *Initial conditions are taken from experimental measurements.*



EQUATIONS

NIMROD presently solves Maxwell's equations plus the following set of fluid moment relations:

$$\rho\left(\frac{\partial}{\partial t}+\mathbf{V}\cdot\nabla\right)\mathbf{V}=\mathbf{J}\times\mathbf{B}-\nabla p'-\nabla\cdot\Pi'$$

$$\mathbf{E}=-\mathbf{V}\times\mathbf{B}+\frac{1}{en}\frac{\left(1-Z_e m_e/m_i\right)}{\left(1+Z_e m_e/m_i\right)}\mathbf{J}\times\mathbf{B}+\eta\mathbf{J}$$
$$+\frac{1}{\varepsilon_0\omega_p^2}\left[\frac{\partial\mathbf{J}}{\partial t}+\nabla\cdot\left(\mathbf{JV}+\mathbf{VJ}\right)+\sum_\alpha\frac{q_\alpha}{m_\alpha}\left(\nabla p'_\alpha+\nabla\cdot\Pi'_\alpha\right)\right]$$

$$\frac{3}{2}\left(\frac{\partial}{\partial t}+\mathbf{V}_\alpha\cdot\nabla\right)p_\alpha=-\frac{5}{2}p_\alpha\nabla\cdot\mathbf{V}_\alpha-\nabla\cdot\mathbf{q}_\alpha+Q_\alpha$$

where quasineutrality, $n_e \cong Zn_i \equiv n$, $Z \equiv -q_i/q_e$, is assumed, ω_p is the plasma frequency, and ρ is the mass density.

CHALLENGES

- Our semi-implicit time advance solves the full ideal MHD operator for accuracy at large time steps. The matrix is symmetric positive definite but is often very ill conditioned. Subdomain-based preconditioners lose efficiency in large grids. Below we describe a global parallel preconditioning scheme that scales well for regions of structured grid. We are investigating the AZTEC package for help with regions of unstructured grid.

- Equilibration of temperature along field lines in realistic conditions stresses spatial truncation errors with extreme anisotropy. A semi-implicit form of anisotropic thermal conduction has been implemented and is now being used in simulations of neoclassical tearing modes.

- Linear and bilinear basis elements are convenient, but convergence properties have been disappointing, especially with respect to the required poloidal resolution. We are generalizing the NIMROD formulation to use polynomials of larger degree.

- Our time split semi-implicit scheme seems susceptible to numerical instabilities when fluid Reynolds numbers are significant. Efforts to improve the time and spatial discretization are expected to help.

- We have begun code development to incorporate kinetic effects via gyrokinetic particles.

SOLVING ILL CONDITIONED MATRICES

When applied to high temperature magnetized plasma, the NIMROD time advance requires solution of an ill conditioned matrix.

- The condition number of our semi-implicit matrix may be estimated as $v_A^2 \pi^2 \Delta t^2 / \Delta x^2$, from the fastest normal mode represented in the numerical system on a given grid. [The smallest eigenvalue is approximately unity.] When simulating slow resistive behavior, we may need to take time steps that are much larger than the Alfven time for propagation across the entire domain. Thus, the condition number may be very large.

- A further complication stems from parallel decomposition of the poloidal plane. Conjugate gradient iterations require only local communication of decomposed data. However, preconditioning needs some form of global communication to relax errors in the lowest wavenumbers.

- A set of block-based preconditioners have been developed for NIMROD, but all become ineffective as the problem sizes are pushed to solve realistic conditions.

We have recently found that global line Jacobi preconditioning in the structured grid region scales well with problem size and time step. [See below.]

As a refresher, a Jacobi iteration scheme is the simplest matrix splitting technique for solving a matrix equation. If we split the matrix **A** into **L**+**D**+**U**, where **D** is the diagonal and **L** and **U** are the strictly lower and upper triangles, respectively, Jacobi iterations find the solution of

$$\mathbf{A} \cdot \mathbf{x} = \mathbf{b}$$

by iterating

$$\mathbf{D} \cdot \mathbf{x}_{i+1} = \mathbf{b} - (\mathbf{L} + \mathbf{U}) \cdot \mathbf{x}_i$$

or

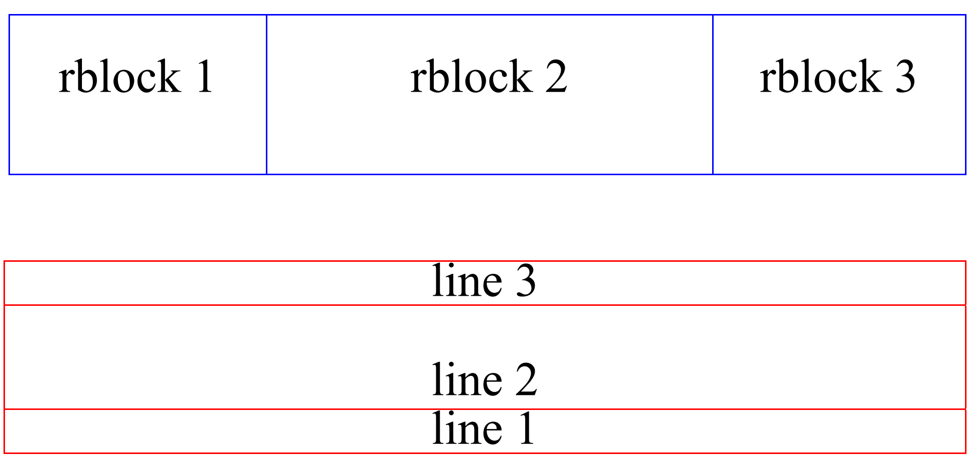
$$\mathbf{D} \cdot \Delta\mathbf{x}_{i+1} = \mathbf{r}_i \equiv \mathbf{b} - \mathbf{A} \cdot \mathbf{x}_i \ .$$

The scheme converges provided the magnitude of the largest eigenvalue of $\mathbf{D}^{-1}(\mathbf{L}+\mathbf{U})$ is less than unity. Furthermore, a Jacobi iteration step may be used to precondition the conjugate gradient scheme.

The definition of **D** may be generalized to improve the effectiveness of each iteration step.

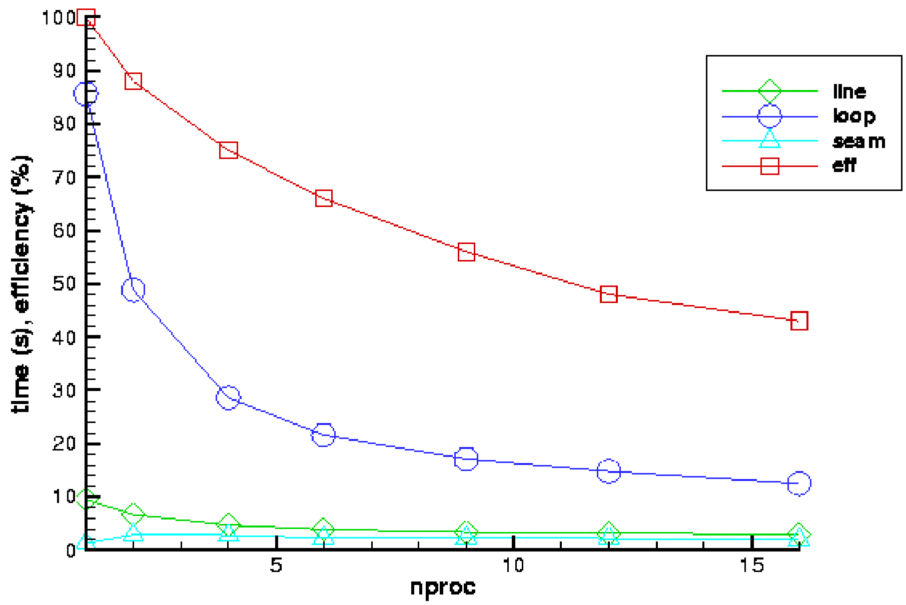
We have found that using the 1D matrices formed by matrix elements along the radial grid lines and along the azimuthal grid lines in the structured grid region produces an effective preconditioner when the semi-implicit matrix is ill conditioned. [This forms two sets of 1D matrices, and the solutions from the two sets are averaged.]

This line-Jacobi preconditioning is most effective when the 1D solves are extended as far as possible across the grid. Data for each line may be distributed on different processors. We use non-blocking MPI point-to-point communication to make the exchanges illustrated in the following schematic:

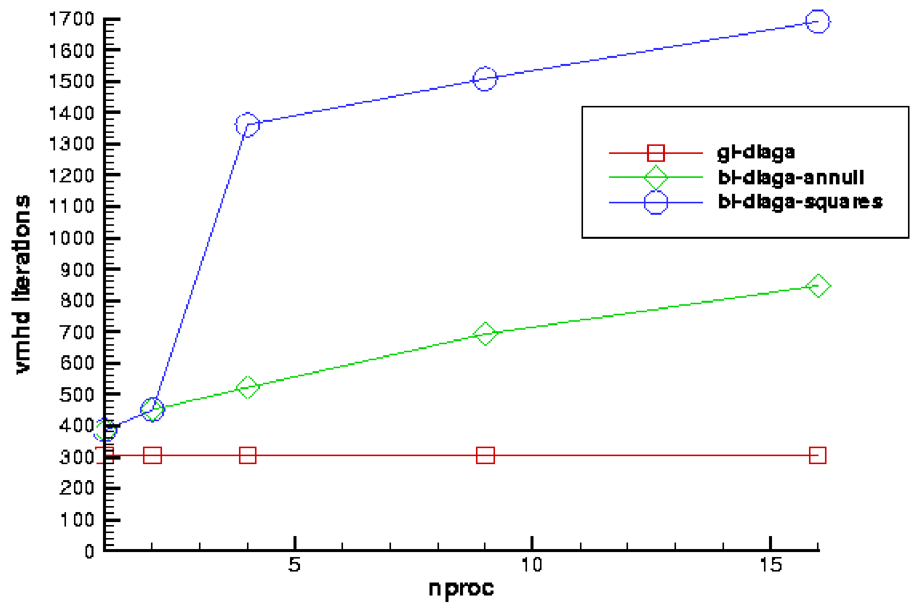


Decomposition swaps reorganize from the normal block pattern (above) to the line pattern (below) for global preconditioning, then swap back to the block pattern.

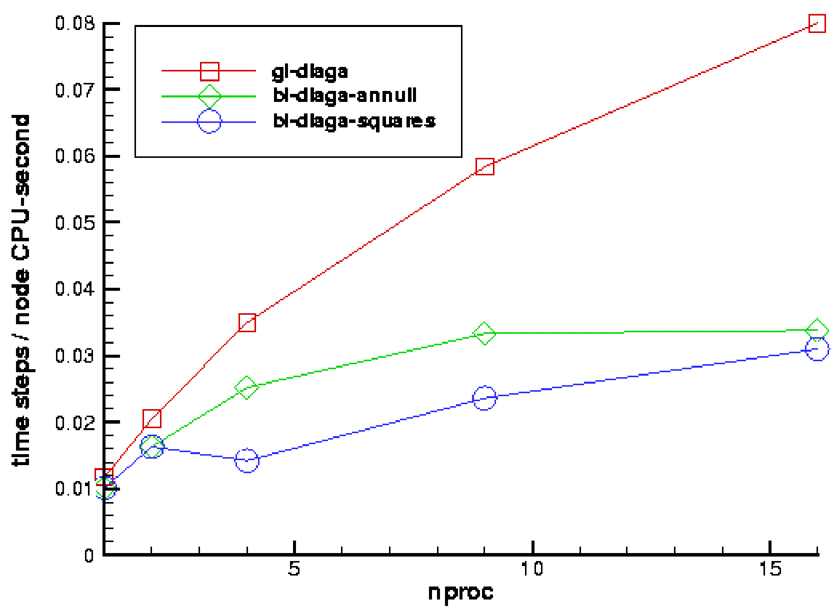
SCALING RESULTS



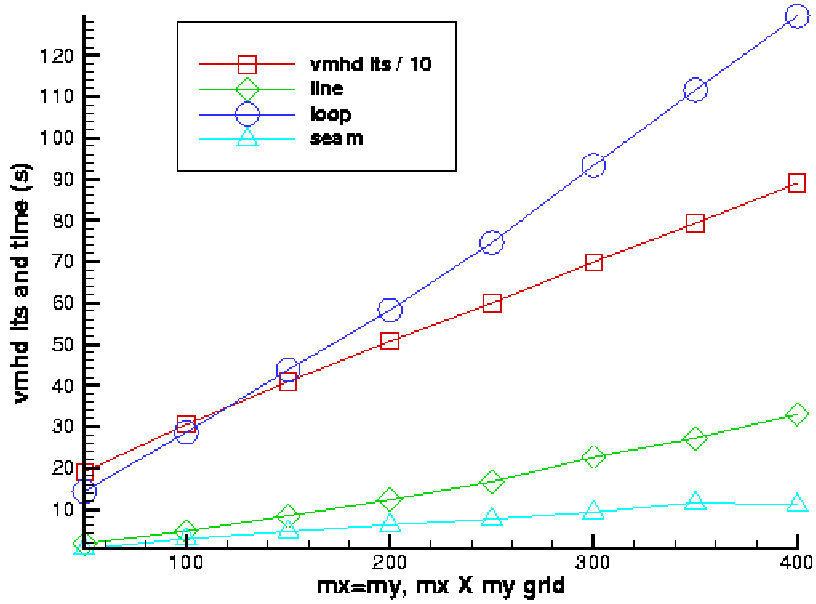
Fixed problem size scaling for global line Jacobi on the T3E for a 100x100 polar grid case. The reported times are the total time step loop ("loop"), the line swapping time ("line") including data copies and parallel communication, and the seam communication time ("seam"). Times are per time step, averaged over 5 steps. The efficiency is (1-block loop time)/(# processors * loop time). The number of blocks equals the number of processors in each case, and the nxbl / nybl decomposition is balanced.



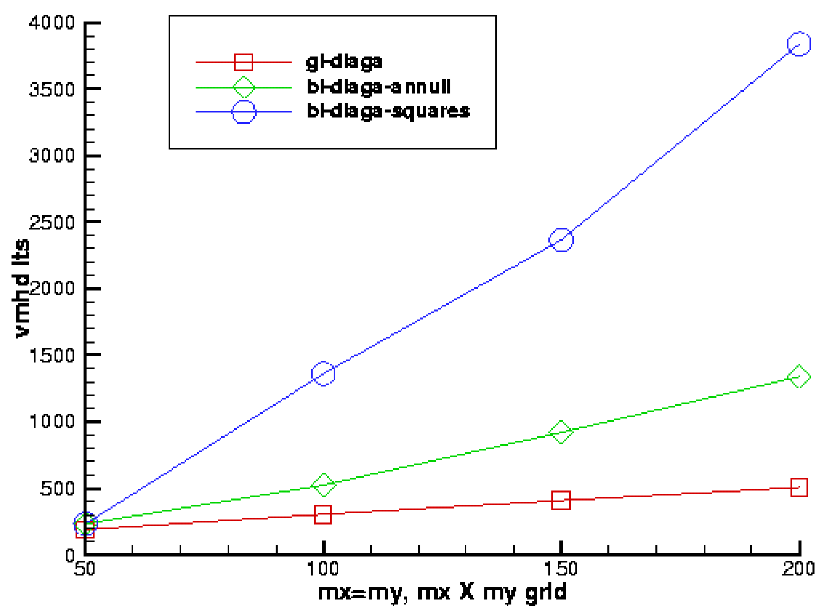
Comparison of preconditioner effectiveness on the 100x100 case. Block-based line Jacobi with periodic blocks (nybl=1, nxbl=nprocs), and balanced decomposition are compared with the global scheme.



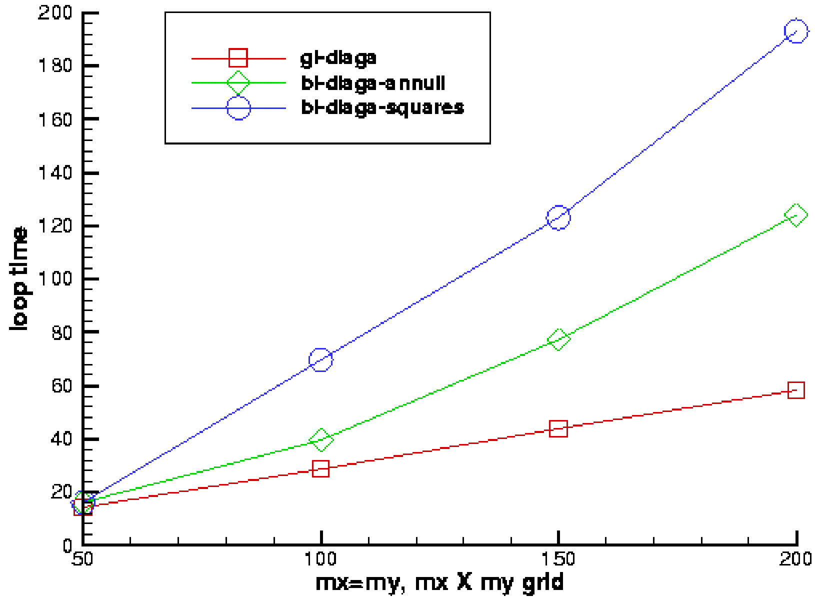
Comparison of time step rates (per CPU time per processor). The rate is the inverse of the loop time per node per time step.



Increasing problem size scaling for global line Jacobi on the T3E. The nxbl / nybl decomposition is balanced.



Comparison of preconditioner effectiveness with increasing problem size.



Comparison of time per time step on the T3E with increasing problem size.

Linking the AZTEC Parallel Linear Solver Package

What is AZTEC?

- solves linear systems of equations
- variety of pre-conditioners (Jacobi, ILU, ICC, ILUT, ...)
- variety of solvers (CG, GMRES, ...)
- point or vector unknowns (MSR, VBR matrix formats)
- real and complex solves
- designed for unstructured grids, asymmetric matrices
- highly parallel
- developed at Sandia National Labs (http://www.cs.sandia.gov/CRF/aztec1.html)
- similar in scope to Argonne's PeTSc

Why?

- rapid trial of different solvers and preconditioners on NIMROD physics
- NIMROD preconditioners work well on regular-blocks, but they only use point-block Jacobi for triangulated regions
- AZTEC has multilevel solvers for general unstructured grids (still in beta test)

Code development was required to link NIMROD and AZTEC, since they have different representations for matrix equations.

NIMROD:
- duplicates unknowns at processor boundaries
- matrix rows/elements not fully formed at boundaries
- F90 data structures with local indexing

AZTEC:
- unknowns must be assigned uniquely to a processor
- processor must own all matrix elements (entire row of matrix)
- C data structures with global indexing

So far, three tests have been run to compare AZTEC solvers with the preconditioned conjugate gradient solver contained within NIMROD.

All timings in are in CPU seconds on 1 processor of the T3E. The convergence tolerance is 10^{-6}.

Test #1: 64x100 circular grid w/ center point

	Complex Equation			Real Equation		
	CPU	its	err	CPU	its	err
NIM Jacobi	60	1000	$2x10^{-3}$			
NIM line-Jacobi	48	453	$1x10^{-6}$	3.3	48	$1x10^{-6}$
AZ CG/Jacobi	84	1000	$2x10^{-3}$			
AZ CG/ICC	267	1000	$7x10^{-4}$	7.1	56	$1x10^{-6}$
AZ GMRES/Jacobi	351	1000	$5x10^{-4}$			

Test #2: 2 external tblocks + 48x32 circular grid w/ center point

	Complex Equation			Real Equation		
	CPU	its	err	CPU	its	err
NIM Jacobi	51	1000	$5x10^{-4}$			
NIM line-Jacobi	21	241	$1x10^{-6}$	1.5	25	$1x10^{-6}$
AZ CG/Jacobi	58	1000	$5x10^{-4}$			
AZ CG/ICC	153	789	$1x10^{-6}$	3.9	29	$1x10^{-6}$
AZ GMRES/Jacobi	231	1000	$9x10^{-5}$	4.0	72	$1x10^{-6}$
AZ GMRES/ICC	219	583	$1x10^{-6}$	4.1	26	$1x10^{-6}$

Test #3: same as #2 w/ 10x smaller timestep

	Complex Equation			Real Equation		
	CPU	its	err	CPU	its	err
NIM Jacobi	29	571	$1x10^{-6}$	1.6	40	$1x10^{-6}$
NIM line-Jacobi	5.6	65	$1x10^{-6}$	0.73	12	$1x10^{-6}$
AZ CG/Jacobi	34	573	$1x10^{-6}$	1.2	40	$1x10^{-6}$
AZ CG/ICC	48	183	$1x10^{-6}$	3.1	15	$1x10^{-6}$
AZ GMRES/Jacobi	149	657	$1x10^{-6}$	1.6	35	$1x10^{-6}$
AZ GMRES/ICC	71	169	$1x10^{-6}$	3.2	14	$1x10^{-6}$

Comments:

- NIMROD line-Jacobi preconditioning works best in these test problems even when tblocks exist.
- The only AZTEC method faster than its NIMROD equivalent is real Jacobi.
- GMRES is about 4x slower per iteration than CG.
- AZTEC scales well in parallel, but so does the NIMROD line-Jacobi.
- This is (primarily) a structured grid and a symmetric (Hermitian) matrix. AZTEC development has focussed on unstructured grids and nonsymmetric matrices.

What to try with AZTEC?

- *multilevel* should work well on problems where Jacobi works but is slow.

- *dual preconditioning*
  - NIMROD preconditions rblocks
  - AZTEC preconditions tblocks
  - AZTEC performs the solve

GENERALIZED BASIS ELEMENTS

The truncation error of the finite element method is intimately tied to interpolation errors of the chosen discrete space.

For piecewise polynomials of order $r \geq 1$, the interpolation of a sufficiently smooth function u satisfies:

$$\|u - \Pi_h u\|_{0,\Omega} \leq Ch^{r+1}|u|_{r+1,\Omega}$$

and

$$\|u - \Pi_h u\|_{1,\Omega} \leq Ch^{r}|u|_{r+1,\Omega}$$

where h is a measure of the grid spacing, and $\| \ \|$ and $| \ |$ are the norm and semi-norm, respectively, of the Sobolev space over the domain Ω with the first subscript indicating order of differentiation. [Jiang, *The Least-squares Finite Element Method,* Springer-Verlag 1998, p. 64.]

Hence, order of convergence depends on the choice of polynomials used.

Until now, the NIMROD algorithm has been restricted to using only bilinear basis functions in regions of quadrilateral elements and linear basis functions in regions of triangular elements.

This restriction has led to at least two difficulties:

1) Spatial convergence in the azimuthal direction takes a large number of elements, often comparable to the resolution required in the radial direction. Besides the obvious costs associated with using more elements, this also increases matrix condition numbers and tightens flow CFL restrictions.

2) Our magnetic field representation has nonzero divergence, and controlling that error without being too invasive is often tricky.

We expect that using quadratic basis elements will improve both issues significantly.

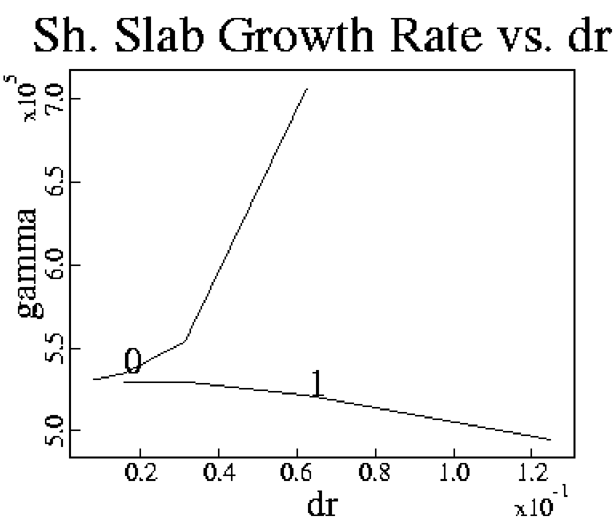
- Quadratic elements will better represent poloidal curvature in magnetic fusion configurations.

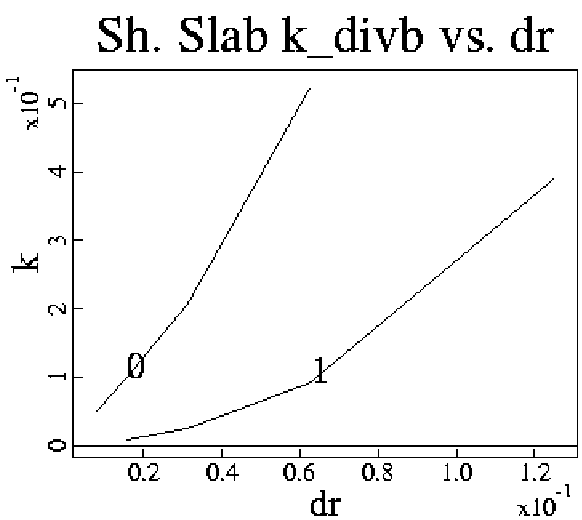
- Quadratic elements will allow us to satisfy the Ladyzhenskaya-Babuska-Brezzi condition for the magnetic field, which is critical for the velocity in finite element computations of incompressible flows. [Gunzburger, "Mathematical Aspects of Finite Element Methods for Incompressible Flows," in *Finite Elements Theory and Application,* eds. Dwoyer, Hussaini, and Voigt, ICASE/NASA LaRC Series, Springer-Verlag, 1986.]

NIMROD is presently being modified to use any Lagrange type element, where only first-order derivatives are square-integrable.

The generalized version of NIMROD is partially functional, and we have been able to compare biquadratic and bilinear elements on two test problems.

1) Sheared slab with a linear *n*=0 tearing instability, where then number of cells in the "radial" direction is twice the number in the "azimuthal" direction.

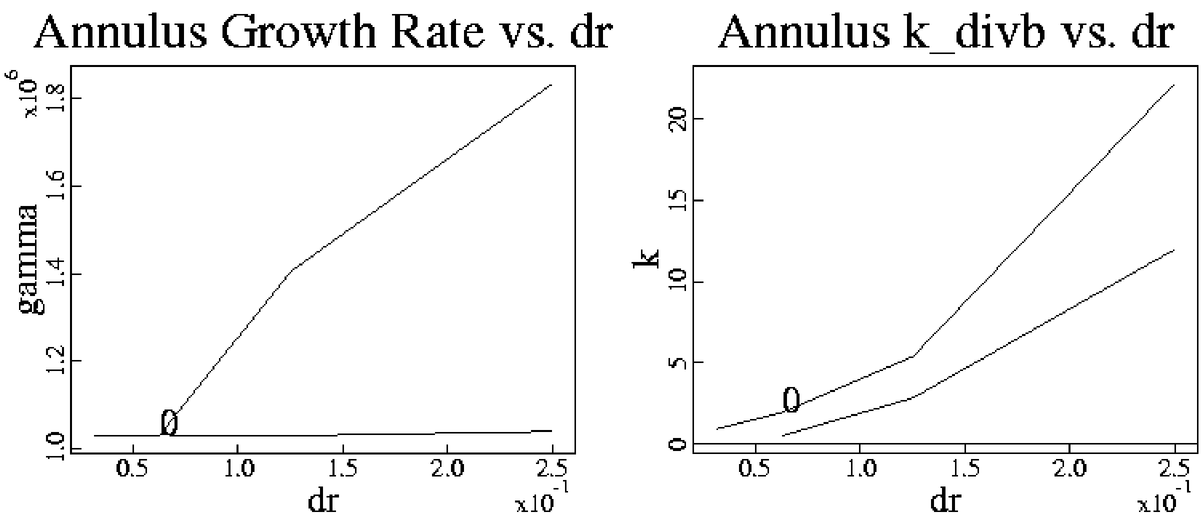




- The curves labeled "0" result with bilinear elements and the curves labeled "1" result with biquadratic elements.
- The quantity k_divb is a wavenumber associated with error in magnetic field divergence. It is defined as

$$\sqrt{\frac{\int dVol(\nabla \cdot \mathbf{b})^2}{\int dVol\mathbf{b}^2}}$$

2) Resonant n=2 instability in an annulus. Azimuthal resolution is the same as radial resolution in these cases.



<u>Comments on these test cases:</u>

- Biquadratic elements use twice as much data per direction as bilinear elements, so the CPU times for a biquadratic simulation scales like CPU time for a bilinear simulation with twice the number of mesh points per direction.
- Only the divergence error for the sheared slab case shows clear asymptotic scaling. For bilinear elements the error converges linearly and for biquadratic elements it converges quadratically. This is expected from the convergence estimate for first-order derivatives.
- Neither test is difficult with bilinear elements. Harder tests will provide more conclusive evidence regarding the efficacy of higher-order elements.

PROBLEMS WITH SIGNIFICANT FLOWS

- NIMROD simulations with significant poloidal flow, even linear simulations, frequently result in numerical instability.

- We have tested two schemes that are proven stable in 1D analyses.

  - The semi-implicit scheme described by Lionello, Mikic, and Linker [J. Comput. Phys. **152**, 346 (1999)] has a leap-frog character. They recommend: 1) including the wave generating terms in the advection predictor step but with a coefficient of 1/2, 2) using the semi-implicit operator in both predictor and corrector steps, and 3) increasing the semi-implicit coefficient from the minimum value required without flow.

  - A time-split scheme related to an Arbitrary Lagrangian-Eulerian approach, where all variables are first advanced without advection, then advection is applied to each.

  - Neither approach solved the problem.

- Present suspicions are that the Ladyzhenskaya-Babuska-Brezzi condition needs to be applied to the magnetic field representation in finite element MHD computations. The development of quadratic basis functions will make this possible.

SUMMARY

- For cases without significant regions of unstructured grid, NIMROD now has an effective, scalable linear solver for conditions where the semi-implicit operator is very ill conditioned.

- We have linked NIMROD with the AZTEC package to find better performance for regions of unstructured grid and to open the possibility of solving nonsymmetric matrices.

- Anisotropic thermal conduction has been implemented with a semi-implicit approach, which seems to be effective even with widely disparate parallel and perpendicular diffusivities.

- The finite element basis functions in NIMROD are being generalized to reduce resolution requirements and to alleviate problems associated with nonzero divergence of magnetic field.

- Simulating strong flows remains a concern. We are expecting to make progress through the generalized basis functions.

- Code development for modeling kinetic effects is underway.

Please visit us at http://nimrodteam.org or http://nimrod.saic.com, where this poster will be made available.