Implementation and evaluation of multigrid linear solvers into extended MHD codes for petascale computing.
Phase 1 Final Report.

Tech-X Corporation
5621 Arapahoe Ave., Suite A
Boulder, CO 80303
Srinath Vadlamani, PI
Scott Kruger and Travis Austin

FY 2007 SBIR/STTR Phase 1
Topic: 36.b
Grant Award Number: DE-FG02-07ER84730

Contents

1 Significance, Background Information, and Technical Approach

1.1 Identification and Significance of the Problem or Opportunity, and Technical Approach

The effort by the fusion community to model reactor plasmas is complicated by multi-scale physics. The modeling of long wavelength, low frequency instabilities from the core to the edge of a tokamak fusion device has traditionally been modeled using the magnetohydrodynamics (MHD) equations. This model exhibits magnetic Alfven waves on a sub-microsecond timescale and global instability phenomena on a 10 milliseconds timescale, giving a timescale ratio that can exceed 10^5 . The typical instability in tokamak devices have narrow radial perturbations compared to the direction parallel to the background magnetic field, giving a spatial scale ratio that can exceed 10^4 . These disparate scale lengths require accurate temporal and spatial discretization. The severity of these computational constraints only increase when the model is more accurately expanded to incorporate additional physics, known as the extended MHD model [1]. High-order finite element representations have demonstrated to be effective for the spatial discretization. Implicit time stepping is a necessity to avoid small time-steps. As a result of these spatial and temporal discretizations, the matrices for the implicit time advance are ill-conditioned (order of 10^9 or greater). The use of efficient parallelized preconditioners and iterative linear solvers are necessary as greater than 70% of the computation is frequently spent in solving the linear matrix that results.

The Office of Science is investing considerable resources in its National Leadership Facilities and plan to have a petascale system online by 2008. The new systems offer many opportunities, but also many challenges, especially for initial-value codes like those that solve the extended MHD system. To enable the codes in the Office of Fusion Energy Sciences (OFES) Center for Extended MHD Modeling (CEMM) SciDAC project to fully exploit the new systems, solvers that can scale to the petascale are needed.

Multigrid linear solvers have demonstrated scaling to the petascale on elliptic operators. Application of multigrid to the operators used in extended MHD, is still an open research topic. The Center for Applied Scientific Computing (CASC) library of high performance preconditioners (HYPRE) is a project that has optimized the parallel performance of these types of methods, thus is a natural candidate for the exploration of implementation and validation for petascale performance. This project will also require development of the multigrid approach on dispersive operators.

This proposed project will improve existing multigrid linear solver libraries applied to the extended MHD system to work efficiently on petascale computers. This improved library will enable the extended MHD simulation codes to efficiently use petascale computers. By providing this capability to the CEMM project, we will enable the fusion community to make greater use of existing investments in computational development and achieve greater physics fidelity in the modeling of instabilities in tokamak plasmas. The success of the CEMM project will improve the scientific community's confidence in the International Thermonuclear Experimental Reactor (ITER) project. The goal of this project will be achieved by closely interacting with the domain scientists of the CEMM SciDACs project, such as Dr. Scott

Kruger of Tech-X Corporation , and the computational mathematics group at Front Range Scientific Computations, Inc. led by Drs. Marian Brezina, Steve McCormick and Tom Manteuffel.

1.1.1 Background Information: Extended MHD Modeling of Fusion Plasmas

Magnetohydrodynamic equations are widely used to model plasmas, including laboratory and space plasmas. By integrating the particle distribution function; which is a function of space, velocity, and time; over all velocity space, a reduction in dimensionality results in a significant simplification of the description, and a concomitant savings in computational cost. Formal derivations of fluid equations result in equations that are more accurate than the more familiar MHD equations. These fluid equations are known as the extended MHD equations, and are summarized in Appendix A, along with a discussion of how they differ from the normal resistive MHD equations.

The standard resistive MHD equations have three fundamental waves that are anisotropic in nature. For magnetized plasmas, these waves are much faster than the instabilities that are of interest. To model the slow growing instabilities, implicit methods have long been used in the fusion community [2, 3]. The extended MHD equations add terms that make the implicit methods even more challenging by adding dispersive waves to the normal modes, and highly anisotropic thermal diffusion. The implicit scheme used by the NIMROD code [4] for the full extended MHD equations is discussed in Appendix A. The NIMROD code is one of the codes under the Center for Extended MHD Modeling (CEMM) SciDAC. The other code in the CEMM SciDAC is the M3D code [5]. Unlike NIMROD which uses a primitive variable representation, the M3D uses a potential representation for the vector fields which offers certain advantages in representing the longitudinal and solenoidal components. The matrix operators will have many similarities however. Another similarity between the two codes is that they use different discretization schemes for the toroidal and poloidal directions as seen in Fig. 1. The reason for the different discretization schemes in the toroidal direction is that tokamak simulations typically have the symmetric part of the magnetic field

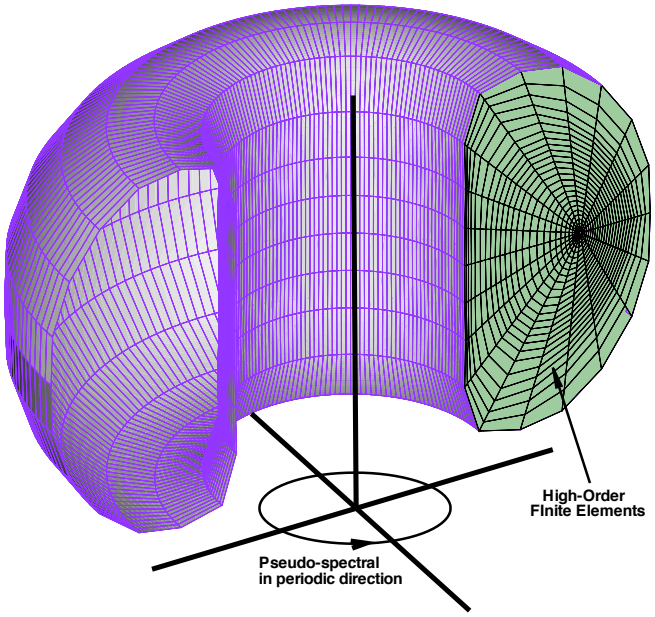

Figure 1: NIMROD uses high-order finite elements in the poloidal plane, and a pseudo-spectral discretization for the toroidal direction. The M3D code differs by using unstructured elements in the poloidal plane and high-order finite differences in the toroidal direction.

energies at least an order of magnitude larger than the non-symmetric components of the evolution. The difference between the two directions has many implications for the matrix inversion routines used by the codes.

For example, earlier versions of the extended MHD codes always assumed that just symmetric parts of the density and resistivity were needed, and were not evolved. In the case of NIMROD, this meant that no Fourier transforms were needed when solving the matrix equations, resulting in a significant computational savings. As the capability of the code has been extended to include the full three-dimensional variations of these quantities, the approximate two-dimensional nature of the quantities has been used to construct preconditioners for the full three-dimensional solves. As we discuss the operators below, we will discuss both the two-dimensional (or perhaps more aptly, linear in the toroidal direction) operators, as well as the full three-dimensional operators. This will help explain the workplan in this Phase II.

As detailed in Appendix A, the time advance has several matrix operators for advancing the time discretization scheme. The first operator we wish to consider is the resistive diffusion operator which is used in the magnetic field advance:

$$\mathcal{D}_{res}(\Delta\vec{B}) = \vec{\nabla} \times \left(\frac{\eta}{\mu_0} \vec{\nabla} \times \Delta\vec{B} \right), \tag{1}$$

where η is the resistivity and can be a function of the temperature ($\eta \sim T^{-3/2}$). When the resistivity is a function of the three-dimensional temperature, then the toroidally-averaged resistivity is used in constructing the two-dimensional preconditioner. The equivalent Courant-Friedrich-Lewy (CFL) number for this operator $(\eta/\mu_0 \Delta t/(\Delta x)^2$ is typically of order unity. Another operator that is always used in the magnetic field advance is the divergence cleaning operator

$$\mathcal{D}_{divB}(\Delta\vec{B}) = \vec{\nabla}\Big(\kappa_{divB} \vec{\nabla} \cdot \Delta\vec{B} \Big). \tag{2}$$

This operator can be done as an operator split from the resistive operator as discussed in the Phase I accomplishments (Sect. 1.3.3). Because κ_{divB} is a constant, this operator is always two dimensional.

The ideal MHD wave operator is always used as it is required to handle the stiffness of the anisotropic MHD waves. This semi-implicit operator is in the velocity advance and has the form:

$$\begin{aligned} \mathcal{L}_{ideal}(\Delta\vec{V}) \;=\; & \frac{1}{\mu_0} \left[\vec{\nabla} \times \vec{B} \times \vec{\nabla} \times \left(\Delta\vec{V} \times \vec{B} \right) - \vec{B} \times \vec{\nabla} \times \left[\vec{\nabla} \times \left(\Delta\vec{V} \times \vec{B} \right) \right] \right] \\ & - \vec{\nabla} \left[\Delta\vec{V} \cdot \vec{\nabla} p + \frac{5}{3} p \vec{\nabla} \cdot \Delta\vec{V} \right] \end{aligned} \tag{3}$$

The waves that this operator acts on are non-dispersive ($\omega \sim k$). With this operator, the NIMROD code can take time steps with CFL numbers on the order of 10^5 [4]. The large CFL numbers puts large off-diagonal terms into the matrix making them ill-conditioned; however, the resultant matrix operator has the advantage of being Hermitian positive definite.

The first operator that we wish to consider that is beyond the normal resistive MHD model is the operator required for anisotropic heat conduction:

$$\mathcal{D}_{thermal}(\Delta T_\alpha) = \vec{\nabla} \cdot \Big((\kappa_\parallel - \kappa_\perp) \hat{\mathbf{b}}\hat{\mathbf{b}} \cdot \vec{\nabla} \Delta T_\alpha + \kappa_\wedge \hat{\mathbf{b}} \times \vec{\nabla} \Delta T_\alpha + \kappa_\perp \vec{\nabla} \Delta T_\alpha \Big) \tag{4}$$

Operator	**Physics**	**Eqn.**	**Properties**	**3 Dimensional?**
$\mathcal{D}_{res}$	Resistive Diffusion	$\vec{B}$	HPD	If $\eta \sim T^{3/2}$
$\mathcal{D}_{divB}$	Divergence Cleaner	$\vec{B}$	HPD	No
$\mathcal{L}_{ideal}$	MHD Waves	$\vec{V}$	HPD	If n is modeled as 3D
$\mathcal{D}_{thermal}$	Anisotropic Thermal Diffusion	T_α	HPD and Non-Symmetric	Always
$\mathcal{L}_{whistler}$	Whistler Waves	$\vec{B}$	Non-Symmetric	Always

Table 1: A summary of the five main operators in the extended MHD time advance.

The first and last terms are symmetric, but the middle term is non-symmetric, and as a result is not always included (it has a minor physical effect compared to the other terms). Also, the temperature dependence of the first diffusion coefficient ($\kappa_\parallel \sim T^{5/2}$) can be important in some simulations, and this also increases the difficulty of the numerical solve (and accuracy of the time advance). It is more common to have constant coefficients with the correct ratios of parallel to perpendicular diffusion. For tokamak simulations, this ratio is large: $\kappa_\parallel/\kappa_\perp \geq 10^8$. Because of the high anisotropy, the three-dimensional nature of the $\hat{\mathbf{b}}\hat{\mathbf{b}}$ dyad must always be used in this operator for accuracy, and the two-dimensional preconditioner is just the toroidally-averaged version of the full operator.

The next wave operator is in the magnetic field advance and handles the Whistler waves:

$$\mathcal{L}_{whistler}(\Delta\vec{B}) = \vec{\nabla} \times \frac{1}{ne}\left[\left(\vec{\nabla} \times \vec{B}^{j+1/2}\right) \times \Delta\vec{B} + \left(\vec{\nabla} \times \Delta\vec{B}\right) \times \vec{B}^{j+1/2}\right] \tag{5}$$

Whistler waves are dispersive ($\omega \sim k^2$) and are the fastest waves in the system when this term is included in the equations. The resultant matrices are non-symmetric.

1.1.2 Background Information: Properties of Operators

In resistive MHD, the diffusive operator, $\mathcal{D}_{res}$, and the ideal MHD operator, $\mathcal{L}_{ideal}$, present the greatest challenge to the development of efficient resistive MHD simulations. The specific features of these operators are their isotropic and anisotropic curl-curl components. The diffusive operator, $\mathcal{D}_{res}$, is in effect a variation on the 3D eddy current formulation of Maxwell's equations [6]. An enormous amount of research has gone into developing solution methods (both discretization approaches and iterative solution approaches) for such systems in order to properly treat the curl-curl components [6, 7, 8]. Unfortunately, in resistive MHD, the historically challenging curl-curl component is present in three of the operators under consideration, with the additional complication that two of them have anisotropic forms of the curl-curl operator.

The difficulty of the curl-curl operator is based on the richness of its nullspace, which presents tremendous complexity when designing optimal multilevel solvers. For systems with a dominant curl-curl operator, its eigenspace consists of a number of eigenvectors in the near nullspace of the curl-curl operator. Furthermore, these components can be highly oscillatory even though their eigenvalue is relatively small in the eigenspectrum.

Multilevel solution methods like HYPRE's BoomerAMG and Trilinos' ML depend on the assumption that highly oscillatory components have relatively large eigenvalues, such as is found with diffusion operators. When combined in a discretization with a mass matrix from an implicit time step, both $\mathcal{D}_{res}$ and $\mathcal{L}_{ideal}$ lead to a subspace of oscillatory eigenvectors with eigenvalues near one. Furthermore, $\mathcal{L}_{ideal}$ can be highly anisotropic due to the anisotropic nature of magnetic field, $\vec{B}$. All of these features are problematic in the context of multilevel solvers that rely on oscillatory eigenvectors having large relative eigenvalue. For this reason, a significant effort from the applied mathematics community has gone into properly discretizing these equations and especially focusing on the components in the nullspace of the curl-curl operator [9, 10].

Even with the problems presented by $\mathcal{D}_{res}$ and $\mathcal{L}_{ideal}$, the problems induced by moving to extended MHD present another set of related challenges. Two operators are required when moving to extended MHD, $\mathcal{D}_{thermal}$ and $\mathcal{L}_{whistler}$. The anisotropic thermal diffusion operator is well-known to be highly anisotropic achieving anisotropies on the order of 10^{10}. While the anisotropies are troublesome, the applied mathematics community has extensive experience in providing optimal multilevel solution methods for strongly anisotropic diffusion problems. However, $\mathcal{D}_{thermal}$ has the additional complexity of being non-symmetric. Non-symmetric operators have notoriously been troublesome to solve efficiently because there is not a clear delineation between smooth components (with low relative eigenvalue) and oscillatory components (with high relative eigenvalue). An understanding of the eigenspectrum is helpful to properly solve these systems efficiently. The last operator, $\mathcal{L}_{whistler}$, again presents problems as a result of its rich nullspace, and in addition, the operator is non-symmetric resulting in the difficulty of developing optimal solution methods.

Lastly, we note the divergence cleaner that is applied to enforce the divergence-free condition on the magnetic field. Although $\mathcal{D}_{divB}$ is not required by the physics, it is required by the discretization approach presented in Appendix A. Besides enforcing the divergence-free condition on $\vec{B}$, this operator that is added to both $\mathcal{D}_{res}$ and $\mathcal{L}_{whistler}$ improves the condition number of the entire system for $\Delta\vec{B}$ by raising the eigenvalue of those oscillatory nullspace components of the curl-curl operator. Ideally, if κ_{divB} is of the same order as the coefficient on the curl-curl operator, then the entire operator resembles a vector mass matrix plus a vector Lapalacian operator. Realistically, κ_{divB} is not the same order as the coefficient, so in certain case we cannot expect to receive much as much help from $\mathcal{D}_{divB}$. Then, in that case, the oscillatory nullspace components of the curl-curl operator still have an eigenvalue near one, leading to difficulties in developing efficient multilevel solvers.

1.1.3 Background Information: Current linear solver usage in NIMROD

From the beginning of the NIMROD project, it has always used and maintained it's own "home-grown" linear solvers. Initially, this was a conjugate gradient method for the matrices arising from the finite-element plane only (the "two-dimensional" matrices discussed earlier). Comparisons with other iterative solvers at that time showed that NIMROD's solver was more efficient because the global line averaging preconditioner was more effective [11]. A substantial increase in performance ($\sim 4-5$) was obtained when, in collaboration with the TOPS SciDAC, SuperLU [12] was used for these two-dimensional matrices. Currently, SuperLU remains the workhorse for the production level code, even though it tends to scale

poorly for our largest problems.

The development of three-dimensional solves greatly extended NIMROD's capability for including additional physics. The matrix free three-dimensional solves uses conjugate gradient for the symmetric matrices, and GMRES for non-symmetric matrices, because constructing the full three-dimensional matrix for SuperLU would be prohibitive. Currently, all of the preconditioning uses the two-dimensional versions of the three-dimensional operators. Because of the lack in toroidal variation of the solutions of many of our problems, this works well for many problems. For the non-symmetric matrices, this approach has been less effective and other methods are being explored.

One of the disadvantages of using home-grown solvers from the beginning of the project is that the data structures were not designed for interoperability with external packages, but rather were designed for making the finite element operations easier. For example, the two-dimensional matrices which are formed are not in a single contiguous array, but rather are in four separate arrays corresponding to different basis functions of the high-order finite elements. This has been a significant barrier to easily interfacing to the external packages.

1.1.4 Technical Approach

This proposed project will improve existing multigrid linear solver libraries for the extended MHD system to work efficiently on petascale computers. In this Phase I, we explored existing HYPRE solver packages applied to the $\mathcal{D}_{res}$ and $\mathcal{D}_{divB}$ operators in NIMROD, one of the large, parallel extended MHD codes of the CEMM project. The Phase I also consisted of establishing the protocol for validation of the petascale scaling of different HYPRE solvers within NIMROD. In Phase II, we will develop these multi-level solvers for the other extended MHD matrix operators discussed earlier. This work will be beneficial for the other CEMM extended MHD code, M3D. M3D currently uses HYPRE via the petsc interface and uses quite similar operators as NIMROD. Thus, any insight into increasing multigrid applicability for extended MHD will benifit both codes. At the successful completion of the SBIR process, we will deliver the capability for longer extended MHD simulations with less computational cost on petascale computers. This will provide the CEMM project the ability to model the ITER device more accurately. This will help efforts within ASCR for implementation of their valuable mathematical libraries to an active OFES plasma simulation community.

1.2 Anticipated Public Benefits

At the successful completion of the entire SBIR proposal, we will enable the HYPRE multigrid mathematical library solver for extended magnetohydrodynamic equations that works efficiently in the proposed petascale computers. This capability will allow the codes of the CEMM project to simulate the complicated components of fusion plasmas in a reasonable amount of time. We will contribute to the efficacy of multigrid methods for high performance computing. For example, the climate modeling community uses semi-implicit operators on high-order finite elements [13], and their operators share many similarities. For further benefits please refer to the separate Commercialization pdf.

1.3 Degree to which Phase I has Demonstrated Technical Feasibility

Objective 1: Implemented the HYPRE solver and preconditioner package into NIMROD via PETSc as the interface.

Objective 2: Provided initial metrics to establish the efficiency of currently available and future solvers.

Objective 3: Explored initial analysis necessary for development of dedicated multigrid course-grid correction for the extended MHD equations.

We have achieved the Phase I objectives by carrying out the tasks proposed in the Phase I. The research results for the Phase I project have well-demonstrated the technical feasibility of successfully carrying out the overall project objectives. These results are discussed in detail below.

1.3.1 Task 1: Implemented HYPRE Solvers In NIMROD Via PETSc

PETSc offers a suite of preconditioners that can be applied to facilitate the convergence of linear solves for extended MHD. Efficient linear solves of poloidal plane problems in NIMROD are the key to success of large simulations that will allow to understand the physics at very fine spatial resolutions. Multilevel methods, in which coarse grids are used to accelerate fine-grid convergence to solution, offer the best promise in attaining both fast linear convergence and good parallel performance. The downside of these methods is the implementation complexity, which slows down their widespread acceptance. The first task in the Phase I was to enable the NIMROD code to use the PETSc interface to allow access the capability to use multilevel methods.

We completed this Phase I task successfully. The PETSc framework requires users to determine use of real or complex primitive data types upon build of the package. To further restrict users, external multigrid packages such as HYPRE only possesses real data type interfaces in PETSc. It is possible to write the $\mathcal{D}_{res}$ and $\mathcal{D}_{divB}$ operators in terms of only real matrices and solve for the real and complex parts of the solution vectors individually. This results in a saving in memory storage for these matrices, and NIMROD takes advantage of this. Because these operators share many similarities of the other operators, we used just these two operators in our prototype implementation of the PETSc interface. Many of the difficulties in dealing with the NIMROD data structures was learned in doing this step.

Until the success of this task, NIMROD used only the SuperLU (parallel or sequential) for preconditioning within its own home-grown conjugate-gradient and GMRES solvers for the matrix-free portion of the three-dimensional matrices. The reason that the homegrown solvers are used instead of standard packages is due to the complicated data structures that are used within the NIMROD code to handle the finite-element quadratures and the pseudo-spectral discretization. The data structures were created to make the discretization calculations simple, and by writing their own solvers, the difficulty of using an external package was avoided.

As mentioned earlier, the only external package NIMROD used was SuperLU. Prior experiences indicated that significant performance advantages can be gained strictly by creating an efficient interface to external packages, but that ultimately overall performance requires an intimate understanding of the discretization and data structures. This was true for the conjugate gradient solvers, and is especially true for implementing multigrid methods as presented in the background section. To test the implementation we compared the PETSc interface versus the original use of the distributed SuperLU_Dist interface, which stores groups of rows local to a processor for memory efficiency. Processor scaling tests were performed on the NERSC leadership machine Bassi. All cases are *weak scaling* where by the problem size (ie. number of unknowns) increases proportionally to the increase of number of processors. The simulation was run for 100 timesteps for each case in order to suppress start up computational costs from dominating any timing results. The scaling studies ceased at 256 processors because Bassi, as a leadership class machine, is heavily used for scientific applications with a limited 888 processors available to all users. Requests for simulations using greater than 256 processors were delayed in the queuing system indefinitely. There are current efforts to continue these scaling studies and the Phase II tasks on Franklin, a NERSC Cray XT4 system with 19,320 processor cores available for scientific applications. The NIMROD project has been awarded the Scaling Reimbursement Program on Franklin.

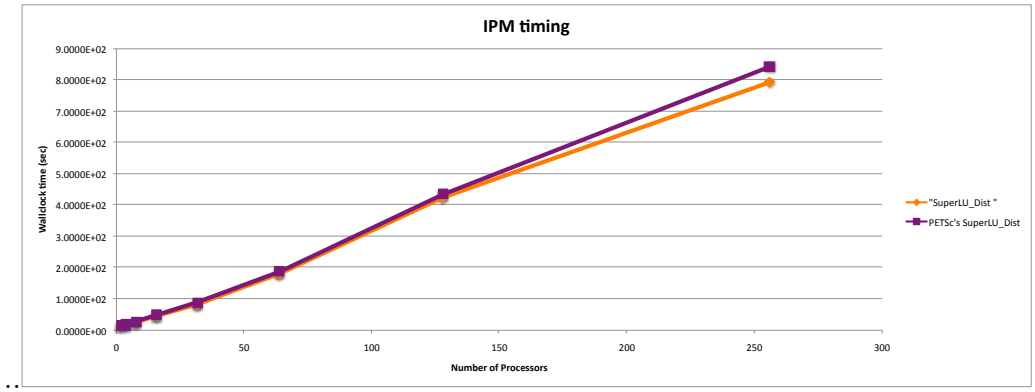

Figure 2: A comparison between direct access to the SuperLU library and via the PETSc interface shows no efficiency loss due to interface

Figure 2 displays the comparison of timings for running NIMROD with direct access to the SuperLU_Dist library and a custom build of PETSc that is linked to the same SuperLU_Dist library. This graph conveys the minimal overhead for using the PETSc interface. The discrepancy for the 256 processor timing is well within timing discrepancies that can be observed by running the same simulation at different daily start times. The lack of weak scaling demonstrated by this graph is because the number of unknowns for the scaling study was too small. This results in simulations with large communication compared to computation. The reason for using the insufficient unknowns is because of initial difficulties with an `mpi_allreduce` operation that was present in the original implementation of the SuperLU interface. This limitation has been fixed by Dr. C. Sovinec. We will improve this scaling study when we progress with scaling studies and proposed Phase II tasks on Franklin. The

graph reinforces the conclusion that users should expect almost no computational overhead for using the PETSc interface. This task was accomplished as a result of collaborations with Dr. Sovinec, Dr. Kruger, and the PETSc team.

1.3.2 Task 2: Established Metrics For the Efficiency of Solvers

We completed this Phase I task successfully. The first goal of this task was to establish an appropriate collection of benchmark problems that isolate specific difficulties associated different complicated operators within extended MHD modeled in NIMROD, such as the anisotropic thermal diffusion operator (Eqn 1). The most intensive of benchmark cases should posses sufficiently stiff and anisotropic features which resemble current laboratory plasmas.

The first case is a slow growing $(2, 1)$ "tearing mode" problem, used to benchmark both temporal and spatial discretization of NIMROD [4]. This instability grows much more slowly than the fundamental wave speed. Using NIMROD's $\mathcal{L}_{ideal}$ operator, wave CFL numbers of 5×10^5 were obtained. This problem also used anisotropic heat conduction, and the accuracy of high-order elements was proven for this experimentally-relevant problem. In the published work, the Whistler terms were not used. By varying model parameters and thermal conduction coefficients, we can isolate the individual operators.

The second case is the toroidal Edge Localized Mode (ELM) case being used as a benchmark problem testing the extended MHD model within NIMROD [14]. This problem is a fast growing instability, and the time step is much more restricted due to the large, generated flows. The wave CFL number in this is approximately 500. The problem also has a large toroidal variation in the linear instabilities, which results in broad nonlinear spectrum. This case stresses the three-dimensional solves; i.e., the two-dimensional preconditioners are less effective for this problem. To get the linear instability spectrum correct, two-fluid terms are important, and improving the ability to invert $\mathcal{L}_{whistler}$ is important. This extremely challenging case will provide insight to the multigrid community for further development of coarsening schemes. These two problems will allow a systematic approach to developing existing mathematical packages to help NIMROD approach petscale performance.

The second goal was to provide a common base of *metrics* that convey application behavior with regards to achieving petscale performance. Weak scaling for performance analysis is the chosen method for testing parallel behavior. A necessary requirement for scalability for an iterative solver is as the set of unknowns grows proportionally to the set of processors, the number of iterations should remain the same. This demonstrates that the solver's preconditioner is insensitive to the discretization size, ie. *iteration count does not depend on element size-h.*

The purpose of an iterative solver is to achieve some convergence to a solution. This convergence criteria can be stated as the

$$\frac{||Ae^n||}{||Ax||} \leq tol., \tag{6}$$

where *tol.* is a specified tolerance that usually set by experience of the domain computational scientist to observe the desired phenomena. So efficiency of the preconditioning technique can be quantified by measuring the amount of *computational effort* for a desired *accuracy.*

Usually, computational effort is accounted for by number of relative residuals calculated, which is the number of Krylov iterations needed for the desired convergence.

Of course, simulation *wall-clock-time* to completion is another metric for comparing efficacy of different linear solver methods. In our case, this will reflect efficiency for different preconditioner methods. The last metric we are considering to help us judge the effectiveness of the preconditioning method is to understand its parallel performance with regards to communication. So analyzing Message Passage Interface (MPI) activity with portable profiling tools such as as Integrated Performance Monitoring (IPM) or Cray PAT will be necessary.

The collection of the above three metrics will allow for a sound comparison between preconditioning techniques. This will aid in the effort to scale solvers that will allow NIMROD to achieve petascale performance.

1.3.3 Task 3: Provided Initial Analysis Of Multigrid Capability For Extended MHD

We completed this Phase I task successfully. We used the first benchmark case of the nonlinear toroidal tearing mode with no anisotropic heat conduction and other model restrictions such there was no variation and dependence in the toroidal direction with no 2-fluid effects. This results in all implicit solves being restricted to $2D$ poloidal planes for symmetric positive-definite operators. We used only 2 Fourier modes to limit influence due to calculations over these modes. Furthermore, the restriction of real data type by PETSc and HYPRE forced this Phase 1 activity to only address 3 matrix equations associated with (1) the finite element stiffness matrix (2) the resistive MHD operator $\mathcal{D}_{res}$ and (3) use of the $\mathcal{D}_{divB}$ operator of divergence cleaning. Task 1 of the proposed Phase II is to alleviate this restriction. We performed comparisons on NERSC's Bassi machine between using BoomerAMG and SuperLU_Dist as the preconditioner to NIMROD's internal conjugate-gradient solver. All results are preformed as weak scaling measurements. Figure 3 demonstrates the new

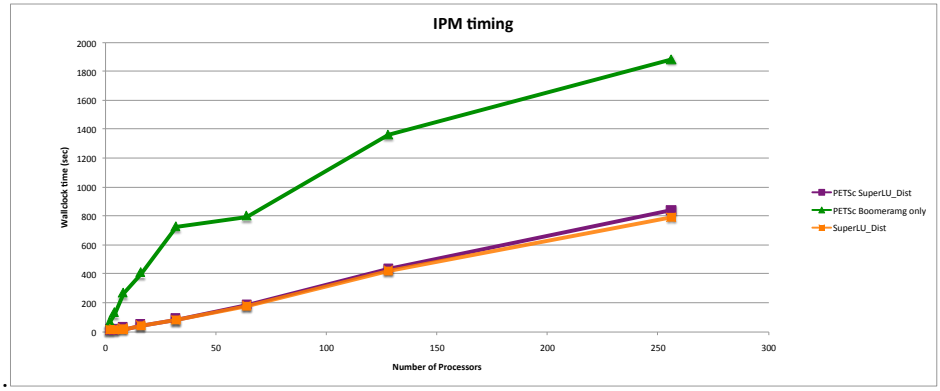

Figure 3: Default BoomerAMG requires 2 times the wall clock time till end of simulation as compared to SuperLU_Dist.

capability of NIMROD to use *HYPRE's BoomerAMG* as a preconditioner for the NIMROD

internal conjugate-gradient linear solver via the PETSc interface. This initial implementation of BoomerAMG used defaults for all available PETSc tuning parameters. Suggestions via conversations with R. Falgout may increase performance of BoomerAMG for this overall wall clock time scaling [15].

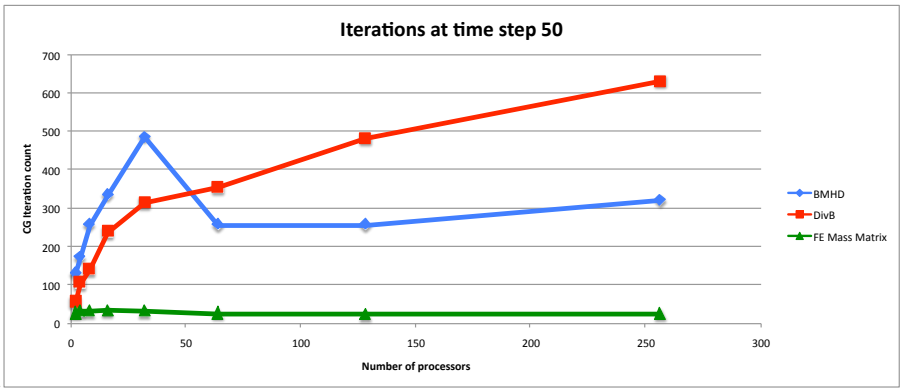

Figure 4: Grid size insensitivity is demonstrated by elliptic problem with mass matrix operator.

Computational work versus accuracy is a desired metric to aid in fair comparisons between preconditioning methods. This may also be used to judge effectiveness of one preconditioning method for different matrix equations. Figure 4 demonstrates "out of the box" BoomerAMG does not fit within the requirement of insensitivity to grid size for the equations with $B_{MHD} \approx \mathcal{D}_{res}$ and $\mathcal{D}_{divB}$ operators. The scaling study was done using the $\mathcal{D}_{divB}$ operator splitting to allow for specific analysis of the $\mathcal{D}_{res}$ operator. The lack of grid size insensitivity for the $\mathcal{D}_{res}$ operator is due to the difficulties associated with AMG applied to the curl-curl operator as discussed in Sect. 1.1.2. Default BoomerAMG does show grid-size insensitivity when preconditioning the iterative solver for the finite element stiffness (mass matrix) equation, which is expected. The results are with a common tolerance of 10^{-12}, which is suggested by domain scientists as necessary convergence to observe needed pertabative phenomena. These results suggest improvements of preconditioning implementation can and should be made.

The three tasks performed in the Phase 1 project have proven the feasibility of implementing the HYPRE solvers via the PETSc framework withing the cutting edge extended magnetohydrodynamic simulation code NIMROD. Metrics and test cases for proper comparison of preconditioning methods have been established. The Phase II project tasks will provide further understanding of multigrid's current treatment of the curl-curl operator. This knowledge will be implemented to further develop NIMROD's preconditioners to achieve linear solvers that work efficiently on petscale computers.

References

[1] D. D. Schnack, D. C. Barnes, D. P. Brennan, C. Hegna, C., E. Held, C. C. Kim, S. E. Kruger, Y. Pankin, A., and C. R. Sovinec, "Computational modeling of fully ionized magnetized plamsas using the fluid approximation," *Phys. Plasmas*, vol. 13, 2006.

[2] D. Harned and W. Kerner *J. Comp. Phys.*, vol. 60, p. 62, 1985.

[3] D. Schnack, D. Barnes, Z. Mikic, D. Harned, and E. Caramana, "Semi-implicit method magnetohydrodynamic calculations," *J. Comp. Phys.*, vol. 70, p. 330, 1987.

[4] C. R. Sovinec, A. H. Glasser, T. A. Gianakon, D. C. Barnes, R. A. Nevel, S. E. Kruger, S. J. Plimpton, A. Tarditi, M. S. Chu, and the NIMROD Team, "Nonlinear magnetohydrodynamics with high-order finite elements," *J. Comp. Phys.*, vol. 195, p. 355, 2004.

[5] W. Park, E. V. Belova, G. Y. Fu, X. Z. Tang, H. R. Strauss, and L. E. Sugiyama, "Plasma simulation studies using multilevel physics model," *Phys. Plasmas*, vol. 6, p. 1796, 1999.

[6] J. Hu, R. Tuminaro, P. Bochev, C. Garasi, and A. Robinson, "Toward an h-Independent Algebraic Multigrid Method for Maxwell's Equations," *SIAM Journal on Scientific Computing*, vol. 27, p. 1669, 2006.

[7] S. Reitzinger and J. Schoberl, "An algebraic multigrid method for ÿnite element discretizations with edge elements," *Numer. Linear Algebra Appl*, vol. 9, pp. 223–238, 2002.

[8] P. Arbenz, M. Bečka, R. Geus, U. Hetmaniuk, and T. Mengotti, "On a parallel multilevel preconditioned Maxwell eigensolver," *Parallel Computing*, vol. 32, no. 2, pp. 157–165, 2006.

[9] T. Kolev and P. Vassilevski, "Auxiliary Space AMG for H (curl) Problems," *Lecture Notes in Computational Science and Engineering*, vol. 60, p. 147, 2008.

[10] M. Clemens, S. Feigh, and T. Weiland, "Construction principles of multigrid smoothers for Curl-Curl equations," *Magnetics, IEEE Transactions on*, vol. 41, no. 5, pp. 1680–1683, 2005.

[11] S. Plimpton, C. Sovinec, and the NIMROD Team, "Recent algorithmic and computational efficiency improvements in the nimrod code," *Bulletin of the American Physical Society*, 1999.

[12] J. Demmel, J. Gilbert, and X. Li, "SuperLU user's guide," *LBNL-44289*, 2003.

[13] S. Thomas, R. Loft, and J. Dennis, "Parallel implementation issues: Global vs. local methods," *Computers in Science and Engineering*, vol. 5, p. 26, 2002.

[14] C. R. Sovinec, D. D. Schnack, A. Y. Pankin, D. P. Brennan, H. Tian, D. C. Barnes, S. E. Kruger, E. D. Held, C. C. Kim, X. S. Li, D. K. Kaushik, S. C. Jardin, and the NIMROD Team, "Nonlinear extended magnetohydrodynamics simulation using high-order finite elements," *Journal of Physics: Conference Series*, vol. 16, pp. 25–34, 2005.

[15] R. Falgout *Private communication.*

[16] S. I. Braginskii, *Transport Processes in a Plasma.* New York: Consultants Bureau, 1966.

[17] S. P. Hirshman and D. J. Sigmar, "Neoclassical transport of impurities in tokamak plasmas," *Nucl. Fusion*, vol. 21, p. 1079, 1981.

[18] F. L. Hinton and R. D. Hazeltine *Reviews of Modern Physics*, vol. 48, p. 239, 1976.

[19] C. R. Sovinec, J. M. Finn, and D. del Castillo-Negrete, "Formation and sustainment of electrostatically driven spheromaks in the resistive magnetohydrodynamic model," *Phys. Plasmas*, vol. 8, p. 475, 2001.

[20] S. Kruger, D. Schnack, and C. Sovinec, "Dynamics of a major disruption of a diii-d plasma," *Phys. Plasmas*, vol. 12, p. 056113, 2005.

[21] S. C. Jardin, "A triangular finite element with first-derivative continuity applied to fusion mhd applications," *J. Comp. Phys.*, vol. 200, no. 1, pp. 133–152, 2004.

[22] A. Glasser and X. Tang, "The sel macroscopic modeling code," *Comp. Phys. Comm.*, vol. 164, pp. 237–243, 2005.

A Summary of the extended MHD equations and discretization schemes

Formal derivations of fluid equations are derived by taking velocity moments of the kinetic equation [16]. For magnetized plasmas, the minimal derivation of fluid equations requires moments to be taken for the kinetic equations for two-species, an ion and electron species, to allow for the plasma to be approximately neutral. By transforming to the center-of-mass coordinate system, the single fluid form of the two-fluid equations can be written in terms of evolution equations for density, n, flow, $\vec{V}$, current, $\vec{J}$, and species temperature, T_α:

$$\frac{dn}{dt} + n\vec{\nabla}\cdot\vec{V} = 0, \tag{7}$$

$$m_i n\frac{d\vec{V}}{dt} = -\vec{\nabla}p + \vec{J}\times\vec{B} - \underline{\vec{\nabla}\cdot\vec{\Pi}}, \tag{8}$$

$$\vec{E} + \vec{V}\times\vec{B} = \eta\vec{J} + \underline{\frac{1}{ne}\left[-\vec{\nabla}p_e + \vec{J}\times\vec{B} - \vec{\nabla}\cdot\vec{\Pi}_e\right] + \frac{1}{\epsilon_0\omega_{pe}^2}\left[\frac{\partial\vec{J}}{\partial t} + \vec{\nabla}\cdot\left(\vec{V}\vec{J} - \vec{J}\vec{V}\right)\right]}, \tag{9}$$

$$n\frac{dT_\alpha}{dt} = -(\gamma-1)\left[nT_\alpha\vec{\nabla}\cdot\vec{V}_\alpha + \underline{\vec{\nabla}\cdot\vec{q}_\alpha + \Pi_\alpha : \vec{\nabla}\vec{V}_\alpha - \eta J^2 - Q_\alpha}\right], \tag{10}$$

where the total time derivative, $d/dt = \partial/\partial t + \vec{V}\cdot\vec{\nabla}$. Here $\vec{E}$ and $\vec{B}$ are the electric and magnetic fields $\mu_0\vec{J} = \vec{\nabla}\times\vec{B}$. Q_α represents collisional energy exchange. These equations, in conjunction with Maxwell's equations, are not closed because the viscous stress tensors, Π_α, and the conductive heat flows, $\vec{q}_\alpha$, are not specified. Specifically, truncating the infinite hierarchy of fluid moment equations with the above set requires specification of Π_α and $\vec{q}_\alpha$ [17, 18].

Neglecting the underlined terms gives the (resistive) *magnetohydrodynamic (MHD) equations*. Including the underlined terms, gives the *extended MHD* equations, and includes many important physical effects beyond the usual MHD equations. The focus of the Center for Extended Magnetohydrodynamic Modeling (CEMM) modeling is to develop numerical algorithms for this set of equations, and apply them to problems of relevance to the Office of Fusion Energy Sciences. The additional terms in the extended MHD equations have required significant development of algorithms and experience beyond the more familar MHD equations, because they fundamentally change the nature of the solutions. Here, we briefly discuss the important physical effects that are given by the extended MHD terms, but will not discuss the the stress tensor terms, as those operators are not considered in this proposal.

In the generalized Ohm's law, Eq. (9), the first two underlined terms are known as the two-fluid terms, and are usually the main difference between the MHD equations and the extended MHD equations. The $\vec{J}\times\vec{B}$ term in particular gives rise to the Whister wave, which becomes the fastest wave in the system. This wave is also dispersive, and is typically the most difficult term to treat implicitly. Often in resistive MHD, the resistivity is treated as a constant, but the resistivity actually varies strongly with temperature: $\eta \sim T^{3/2}$ [16]. Including this temperature dependence has been important for many simulations [19, 20], and is a key part of the capabilities of the extended MHD codes.

The most commonly used closure for the heat flux, is the anisotropic diffusion closure of Braginskii [16]. This closure may be written as:

$$\vec{q}_\alpha = -\kappa_\parallel \hat{\mathbf{b}}\hat{\mathbf{b}} \cdot T_\alpha - \kappa_\wedge \hat{\mathbf{b}} \times \vec{\nabla} T_\alpha - \kappa_\perp \vec{\nabla}_\perp T_\alpha \tag{11}$$

where $\hat{\mathbf{b}} = \vec{B}/|\vec{B}|$. The anisotropic diffusion coefficients are temperature dependent. The first coefficient is the largest and the last is the smallest. The ratio of these two coeffients is $\kappa_\parallel/\kappa_\perp \geq 10^{10}$ or greater. Thus, the temperature wants to rapidly equilibrate along the dynamically evolving magnetic field. The difficulty of resolving this anisotropy has lead to the adoption of high-order finite elements by the extended MHD community [4, 21, 22], where the spectral convergence allows one to converge at these ratios.

Given this background, we now present the discretization scheme used by the NIMROD code. We note that the M3D-C1 code uses a potential representation for the magnetic and velocity fields, but the temporal discretization scheme has many similarities to the NIMROD's [21]. NIMROD uses a semi-implicit/fully implicit scheme, with the velocity field staggered a half-step with respect to the other fields. The full scheme is [14]:

$$\begin{aligned} m_i n^{j+1/2} \left[\frac{\Delta \vec{V}}{\Delta t} \right. &+ \left. \frac{1}{2}\vec{V}^j \cdot \vec{\nabla}\Delta\vec{V} + \frac{1}{2}\Delta\vec{V} \cdot \vec{\nabla}\vec{V}^j \right] - \mathcal{L}^{j+1/2}_{ideal}(\Delta\vec{V}) + \vec{\nabla}\cdot\vec{\Pi}_i(\Delta\vec{V}) \\ &= \vec{J}^{j+1/2} \times \vec{B}^{j+1/2} - \vec{\nabla} p^{j+1/2} + -\vec{\nabla}\cdot\vec{\Pi}_i(\vec{V}^j) + m_i n^{j+1/2}\vec{V}^j \cdot \vec{\nabla}\vec{V}^j \end{aligned} \tag{12}$$

$$\frac{\Delta n}{\Delta t} + \frac{1}{2}\vec{V}^{j+1} \cdot \vec{\nabla}\Delta n = -\vec{\nabla}\cdot\left(n^{j+1/2}\vec{V}^{j+1}\right) \tag{13}$$

$$\begin{aligned} \frac{3n}{2}\left[\frac{\Delta T_\alpha}{\Delta t} \right. &+ \left. \frac{1}{2}\vec{V}^{j+1}\cdot\vec{\nabla}\Delta T_\alpha \right] + \frac{1}{2}\mathcal{D}_{thermal}(\Delta T_\alpha) \\ &= \frac{3n}{2}\vec{V}^{j+1}\cdot\vec{\nabla}T_\alpha^{j+1/2} - nT_\alpha\vec{\nabla}\cdot\vec{V}^{j+1} \\ &\quad + \vec{\nabla}\cdot\left((\kappa_\parallel - \kappa_\perp)\hat{\mathbf{b}}\hat{\mathbf{b}}\cdot\vec{\nabla}T_\alpha^{j+1/2} + \kappa_\wedge\hat{\mathbf{b}}\times\vec{\nabla}T_\alpha^{j+1/2} + \kappa_\perp\vec{\nabla}T_\alpha^{j+1/2} \right) \end{aligned} \tag{14}$$

$$\begin{aligned} \frac{\Delta\vec{B}}{\Delta t} &+ \frac{1}{2}\vec{V}^{j+1}\cdot\vec{\nabla}\Delta\vec{B} + \frac{1}{2}\mathcal{L}_{whistler}(\Delta\vec{B}) + \frac{1}{2}\mathcal{D}_{res}(\Delta\vec{B}) + \frac{1}{2}\mathcal{D}_{divB}(\Delta\vec{B}) \\ &= -\vec{\nabla}\times -\vec{V}^{j+1}\times\vec{B}^{j+1/2} + \eta\vec{J}^{j+1/2} + \frac{1}{ne}\left[\vec{J}^{j+1/2}\times\vec{B}^{j+1/2} - \vec{\nabla}p_e^{j+1/2}\right] \\ &+ \left[\frac{\Delta\vec{J}}{\Delta t} + \vec{\nabla}\cdot\left(\vec{V}^{æ1}\vec{J}^{j+1/2} - \vec{J}^{j+1/2}\vec{V}^{æ1}\right)\right] - \kappa_{divB}\vec{\nabla}\vec{\nabla}\cdot\vec{B}^{j+1/2}. \end{aligned} \tag{15}$$

The last term in this equation is used to diffuse divergence errors out of the system because NIMROD does not have a manifestly divergence-free magnetic field. This scheme has been shown to produce acceptable levels of error [4]. Here we use the notation of $\mathcal{L}$ to

denote the implicit wave operators, and $\mathcal{D}$ to denote the implicit diffusion operators. The definitions of these operators are given in the main text, along with a discussion of their implication for matrix solves.

The code has flexibility in solving these equations, and can turn off the extended MHD terms. Also, this discretization scheme shows implicit these contributions to the matrices in the proposal because while they are non-symmetric, they do not present the numerical challenges of the other operators. Also, when the code is run with only symmetric operators, a predictor-corrector method can be used for the advection terms to avoid any non-symmetric linear solves. Finally, we note that there can be a strong coupling among the last three equations, especially when the diffusivities are temperature dependent. Currently, this handled by using a simple predictor-corrector method. Newton methods are being investigated for the numerical accuracy.