SIAM J. Sci. Comput.
Vol. 35, No. 5, pp. S171–S191

ISLAND COALESCENCE USING PARALLEL FIRST-ORDER SYSTEM LEAST SQUARES ON INCOMPRESSIBLE RESISTIVE MAGNETOHYDRODYNAMICS[*]

J. H. ADLER[†], M. BREZINA[‡], T. A. MANTEUFFEL[‡], S. F. MCCORMICK[‡], J. W. RUGE[‡], AND L. TANG[§]

Abstract. This paper investigates the performance of a parallel Newton, first-order system least-squares (FOSLS) finite-element method with local adaptive refinement and algebraic multigrid (AMG) applied to incompressible, resistive magnetohydrodynamics. In particular, an island coalescence test problem is studied that models magnetic reconnection using a reduced two-dimensional (2D) model of a tokamak fusion reactor. The results show that, using an appropriate temporal and spatial resolution, these methods are capable of resolving the physical instabilities accurately at small computational cost. The time-dependent, nonlinear system of PDEs is solved using work equivalent to about 50–60 simple relaxation sweeps (Gauss–Seidel iterations) per time step. Experiments show that, unless the time step is sufficiently small, nonphysical numerical instabilities may occur. Further, decreasing the time step size does not proportionally increase the cost of the computation, because AMG convergence is improved. In addition, an effective implementation of the methods in parallel keeps load balancing issues to a minimum. Various quantities, such as the reconnection rate and the "sloshing" effect of the plasma instability, are measured to confirm that the correct physics is reproduced.

Key words. magnetohydrodynamics, adaptive local refinement, AMG, FOSLS, nested iteration

AMS subject classifications. 65F10, 65N55, 65N50, 76W05

DOI. 10.1137/120880227

1. Introduction.

The island coalescence problem for studying fast magnetic reconnection in a plasma has been studied extensively (e.g., in [9, 11, 34, 41, 42, 45, 52]). Many numerical algorithms have been implemented to simulate this problem using various types of physical and mathematical models [1, 15, 22, 24, 25, 35, 37, 38, 41, 43, 44, 46, 47, 49, 50]. The aim of this paper is to extend the results of [2, 3, 4] and show that by using the first-order system least-squares (FOSLS) finite-element method along with nested iteration (NI), algebraic multigrid (AMG), and an efficiency-based adaptive local refinement scheme (ACE) in parallel, the relevant physics of the reconnection is modeled with a low amount of computational cost. Preliminary results are obtained in [2, 3, 4] using an incompressible resistive magnetohydrodynamics (MHD) model with the above methods. However, limitations with the computational resources used exposed several deficiencies in the method. Namely, at high Lundquist numbers (low resistivity), time integration was not resolving the magnetic instabilities, resulting in numerical oscillations in the current density peak. These, in turn, affected the performance of the solvers and the convergence of the discrete methods. With an efficient implementation of the method on computers with a distributed

[*] Received by the editors June 7, 2012; accepted for publication (in revised form) March 21, 2013; published electronically October 28, 2013. This work was supported by the Department of Energy under grants DE-FG02-03ER25574 and DE-FC02-06ER25784, the Lawrence Livermore National Laboratory under contracts B568677, and the National Science Foundation under grants DMS-0621199, DMS-0749317, and DMS-0811275.

http://www.siam.org/journals/sisc/35-5/88022.html

[†] Department of Mathematics, Tufts University, Medford, MA 02155 (james.adler@tufts.edu).

[‡] Department of Applied Mathematics, University of Colorado at Boulder, Boulder, CO 80309-0526 (marian.brezina@colorado.edu, tmanteuf@colorado.edu, stevem@colorado.edu, jruge@colorado.edu).

[§] Amazon — Wainwright (SEA23), Seattle, WA 98109 (tanglei@amazon.com).

memory architecture, including a parallel version of the ACE adaptive refinement algorithm [16], higher spatial and temporal resolutions are obtained. This higher resolution removes the numerical instability, produces an accurate solution, and verifies the asymptotic efficiency of the NI-Newton-FOSLS-ACE-AMG methodology. This paper shows that decreasing the time step size of the simulations not only increases the accuracy of the numerical solutions, but does so without significantly increasing the amount of computational work. The aim here is to show these results and demonstrate the powerful performance of the NI-Newton-FOSLS-ACE-AMG methodology on a parallel machine. In addition, load balancing issues are easily managed. After each refinement, a new partitioning is created based on a parallel quadtree structure and a space filling curve (SFC). This preserves the locality of the mesh, so that most communication happens among nearest neighboring processors. Also, the ACE algorithm is designed to equally distribute local errors, which leads to nearly uniform refinement on successively finer grids. This, in turn, eliminates load balancing on finer levels. Numerical results show that the methods developed here are highly efficient for solving complex physical problems, such as the MHD system.

This paper starts in section 2 with a description of the NI, FOSLS, AMG, and ACE algorithms, as well as a description of the parallel considerations that are needed to obtain an efficient algorithm. In section 3, a brief background of the MHD system and the island coalescence problem are presented. Then, in section 4, the numerical results for various resistivities including qualitative measures of the accuracy and quantitative measures of the efficiency of the simulation are shown. Finally, section 5 contains concluding remarks and a discussion of future work.

2. NI-Newton-FOSLS-ACE-AMG.

This section briefly describes the basic concepts behind the NI-Newton-FOSLS-AMG approach and introduces the notation used in the rest of the paper.

2.1. FOSLS methodology.

First-order system least squares (FOSLS) is a finite-element method that is based on reformulating a set of PDEs as a system of first-order equations. The problem is posed as the minimization of a functional in which the first-order differential terms appear quadratically, so that the functional norm is equivalent to a norm that is meaningful for the given problem. In equations of elliptic type, this is usually a product H^1-norm. In other contexts, it may be product spaces of H^1, $H(\text{div})$, and $H(\text{curl})$. Some of the compelling features of the FOSLS methodology include self-adjoint discrete equations stemming from the minimization principle; good operator conditioning stemming from the use of first-order formulations of the PDE; and finite-element and multigrid performance that is optimal and uniform in certain parameters (e.g., the Reynolds number for the Navier–Stokes equations), stemming from uniform product-norm equivalence. Many large-scale physical problems, including those described by self-adjoint elliptic PDEs, can be solved by minimizing a known "energy" functional over an infinite-dimensional space of admissible functions. When properly posed, such optimization problems have the advantage that they can be discretized by the Rayleigh–Ritz process of minimizing the functional over a finite-dimensional subspace of the admissible function space. If done with a correct formulation and an appropriate function space, this leads to a continuous and coercive weak form of the problem.

To illustrate the basic concept of FOSLS, consider a PDE written abstractly as $\mathcal{L}\mathbf{u} = \mathbf{f}$. It should be noted that, for a time-dependent, nonlinear problem, it is assumed that the time discretization is done first, followed by several linearization steps. At each step, the following minimization is performed on the semidiscrete linearized system. This is illustrated further in section 2.1.2. Introducing new variables, a first-order system is given:

$$\mathcal{L}_i \mathbf{u} = f_i, \qquad i = 1, 2, \ldots, M, \tag{2.1}$$

where M is the number of equations in the system and $\mathcal{L}_i$ is a first-order, linear, differential operator. Assuming $f_i \in L^2(\Omega)$, consider the associated FOSLS functional given by

$$\mathcal{G}(\mathbf{u}, \mathbf{f}) = \sum_{i=1}^{M} ||\mathcal{L}_i \mathbf{u} - f_i||_{0,\Omega}^2, \tag{2.2}$$

where $||u||_{0,\Omega} = \sqrt{\int_\Omega |u|^2}$ is the L^2-norm. This functional is then minimized over an appropriate Hilbert space, $\mathcal{V}$, such that

$$\mathbf{u} = \arg \min_{\mathbf{v} \in \mathcal{V}} \mathcal{G}(\mathbf{v}; \mathbf{f}). \tag{2.3}$$

Usually, $\mathcal{V}$ is equivalent to a product of H^1, $H(\text{div})$, and $H(\text{curl})$ spaces. Under general regularity assumptions, the homogeneous part, $\mathcal{G}(\mathbf{v}; \mathbf{0})$, is equivalent to the squared $\mathcal{V}$-norm:

$$c_1 \leq \frac{\mathcal{G}(\mathbf{v}; \mathbf{0})}{||\mathbf{v}||_{\mathcal{V}}^2} \leq c_2 \tag{2.4}$$

for some positive constants c_1 and c_2 and for every $\mathbf{v} \in \mathcal{V}$. In this case, the functional is said to be continuous and coercive (i.e., "elliptic") with respect to the $\mathcal{V}$-norm; see, e.g., [20, 21]. This ellipticity guarantees the existence and uniqueness of the solution $\mathbf{u}$. Next, let $\mathcal{V}^h \subset \mathcal{V}$ be a finite-dimensional subspace of $\mathcal{V}$, which often consists of continuous piecewise polynomials. Then, the discretization can be written as a minimization problem:

$$\mathbf{u}^h = \arg \min_{\mathbf{v}^h \in \mathcal{V}^h} \mathcal{G}(\mathbf{v}^h; \mathbf{f}). \tag{2.5}$$

Well-posedness of (2.5) follows directly from ellipticity, since the weak form obtained comes from a minimization principle. Therefore, the FOSLS formulation is not restricted by any Ladyžsenskaja–Babuška–Brezzi condition (i.e., inf-sup condition) [14, 17]. This is a condition on the finite-element spaces that would restrict them to those that ensure stability and weak coercivity of the bilinear form. Also, while not a necessary condition, if $\mathcal{V}$ is a product of H^1 spaces, then ellipticity also ensures that there is an optimal multigrid solver of the discrete system [21, 55]; that is, standard multigrid solvers converge with factors bounded uniformly in the mesh size, h.

The introduction of new dependent variables increases the number of degrees of freedom, much like in the mixed finite-element methods. However, unlike mixed methods, FOSLS yields a symmetric positive definite algebraic system that is, in general, amenable to multilevel solution techniques. As a result, it is often possible to obtain a specified accuracy with much smaller computational cost.

2.1.1. Newton-FOSLS.

In the context of this paper, a nonlinear PDE is considered. Therefore, the functional of the nonlinear operator, referred to as the "nonlinear functional," needs to be minimized. One way to accomplish this is by directly minimizing the nonlinear functional, yielding a nonlinear discrete system to solve. This can be solved using nonlinear multigrid methods such as the full approximation scheme [18]. A simpler approach, Newton-FOSLS, consists of linearizing the PDE itself and then performing the minimization on this "linearized functional." In this way, standard Newton steps can be used in conjunction with classical AMG iterations on the resulting discrete linear systems. This approach is described in more detail in [4, 27]. In addition, other approaches to solving sets of nonlinear transient equations can lead to unconditionally stable methods that can reduce the cost of the linearizations and increase the accuracy of the time discretization (e.g., in [40, 51]).

2.1.2. Time-dependent Newton-FOSLS. To illustrate the approach used, consider a time-dependent nonlinear first-order system of PDEs,

$$\frac{\partial \mathbf{u}}{\partial t} + \mathcal{L}(\mathbf{u}) = \mathbf{f}.$$

First, discretize the time derivative using some time-stepping scheme. For simplicity, consider Backward Euler (BDF-1). Let $\mathbf{u}^k$ denote the solution at time step k with time step Δt. This yields

$$\frac{\mathbf{u}^k - \mathbf{u}^{k-1}}{\Delta t} + \mathcal{L}(\mathbf{u}^k) = \mathbf{f}.$$

Next, linearize the nonlinear system in $\mathcal{L}$ by a simple Newton step about the solution, $\mathbf{u}^k_\ell$, and solve for the update:

$$\begin{gathered} \frac{\delta \mathbf{u}}{\Delta t} + \mathcal{L}'(\mathbf{u}^k_\ell)[\delta \mathbf{u}] = \mathbf{f} + \frac{\mathbf{u}^{k-1} - \mathbf{u}^k_\ell}{\Delta t} - \mathcal{L}(\mathbf{u}^k_\ell), \\ \mathbf{u}^k_{\ell+1} = \mathbf{u}^k_\ell + \delta \mathbf{u}. \end{gathered} \tag{2.6}$$

Here, $\mathcal{L}'(\mathbf{u}^k_\ell)[\delta \mathbf{u}]$ is the Fréchet derivative of $\mathcal{L}$ centered at $\mathbf{u}^k_\ell$ in the direction of $\delta \mathbf{u}$. The FOSLS method is then applied to the linear system obtained in (2.6) by minimizing the functional,

$$\left\| \frac{\delta \mathbf{u}}{\Delta t} + \mathcal{L}'(\mathbf{u}^k_\ell)[\delta \mathbf{u}] - \mathbf{f} - \frac{\mathbf{u}^{k-1} - \mathbf{u}^k_\ell}{\Delta t} + \mathcal{L}(\mathbf{u}^k_{ell}) \right\|_0^2,$$

and solving for a discrete approximation to $\delta \mathbf{u}$ in a subspace of H^1. When referring to the ellipticity of the FOSLS functional, we are usually referring to this linearized system at each time step.

2.2. Efficiency-based adaptive refinement (ACE). Another direct result of FOSLS is that its functional is equivalent to the error in the solution-space norm. In general, this induces a seminorm on a subdomain, which is then available for use as a local a posteriori error estimator. The FOSLS functional provides a unique capability for adaptive refinement: a sharp error indicator at no additional computational cost [10]. Since the functional value is zero at the solution, the FOSLS functional itself is a measure of the total error in a given approximation. It provides both absolute and relative error measures, as well as global and local error estimates that are much simpler and potentially sharper than conventional error estimators. To illustrate this, for any element, $\tau \in \mathcal{T}$, define the local FOSLS functional as

$$\mathcal{G}_\tau(\mathbf{u}^h; \mathbf{f}) = \sum_{i=1}^{M} ||\mathcal{L}_i \mathbf{u}^h - f_i||^2_{0,\tau}. \tag{2.7}$$

Writing $\epsilon_\tau = \sqrt{\mathcal{G}_\tau(\mathbf{u}^h; \mathbf{f})}$, the ellipticity expressed in (2.4) implies that

$$\frac{1}{c_2}\epsilon_\tau^2 = \frac{1}{c_2}\mathcal{G}_\tau(\mathbf{u}^h - \mathbf{u}; 0) \le ||\mathbf{u}^h - \mathbf{u}||^2_{\mathcal{V},\tau} \tag{2.8}$$

and

$$||\mathbf{u}^h - \mathbf{u}||_{\mathcal{V}}^2 \leq \frac{1}{c_1}\mathcal{G}(\mathbf{u}^h - \mathbf{u}; 0) = \frac{1}{c_1}\sum_{\tau \in \mathcal{T}} \epsilon_\tau^2. \tag{2.9}$$

An error estimate, ϵ_τ, that satisfies an inequality of type (2.8) is called locally sharp. It implies that if ϵ_τ is large, then the error is large within that element. In the literature, an inequality of type (2.9) is called a reliability bound; see [54]. Note that a small sum of local estimates, ϵ_τ, implies a small global error.

Specifically, this property of FOSLS helps make it possible to efficiently solve complex systems. At each step in the solution algorithm, a local measure of the functional is available. This allows judgements in the grid-refinement process to be made based on estimates of the increase of accuracy that results from an increase in computational cost. As a result, the elements are ordered (symbolically) in terms of estimated error and a refinement pattern is then chosen that optimizes the *effective error reduction*. In this manner, a near optimal grid is obtained, where optimal means a grid that provides a specified accuracy with the smallest number of degrees of freedom. In other words, the refinement that obtains the highest *accuracy per computational cost* is found [2, 30].

2.3. NI and AMG. Along with the FOSLS discretization and local adaptive grid refinement, multigrid is applied to solve the discrete systems. NI, or full multigrid (FMG) [18] in the multigrid context, involves starting the solution process on a relatively coarse grid, where the computational cost is relatively low. The solution on the coarse grid is used as an initial guess for the problem on the next finer grid. Since the objective on each grid is to minimize the FOSLS functional, the coarse-grid solution should provide a good starting guess. On each refinement level, solving the discrete minimization problem, (2.5), involves fast iterative solvers applied to the matrix equations. If the FOSLS functional is equivalent to a product H^1-norm, then there exists an optimal multilevel solution algorithm [55]. Experience shows that, in this context, AMG also yields an approximate solution to the discrete equations associated with quasi-uniform grids in optimal time with the convergence factor, ρ, bounded uniformly below 1, independent of mesh size h. AMG methods, together with the NI strategy and local refinement, provide a powerful approach for approximating solutions of PDEs. Numerical and theoretical results confirm that the overall cost of such a scheme resides predominantly in the cost of the finest-level processing. Due to the good initial approximation obtained inexpensively from the coarse grids, the total cost is usually much cheaper than solving the problem directly on the finest, near optimal grid, which generally is not even known in advance.

Finally, the NI approach complements the Newton-FOSLS method. Since, on each successively finer grid, the initial guess is a better approximation to the discrete solution, the convergence of Newton's method will take fewer and fewer iterations. As a result, when the desired grid resolution is reached, only one Newton iteration is usually needed to solve the nonlinear problem. This has been theoretically established under mild hypotheses in [27, 28]. This greatly reduces the cost of the algorithm, because the setup cost for construction of the Jacobian in the relinearization can be expensive on fine grids.

2.4. Parallel FOSLS. In this section, application of the method in parallel is discussed, with special attention to the adaptive refinement scheme, ACE. An efficient parallel extension of the ACE algorithm to massively parallel distributed memory

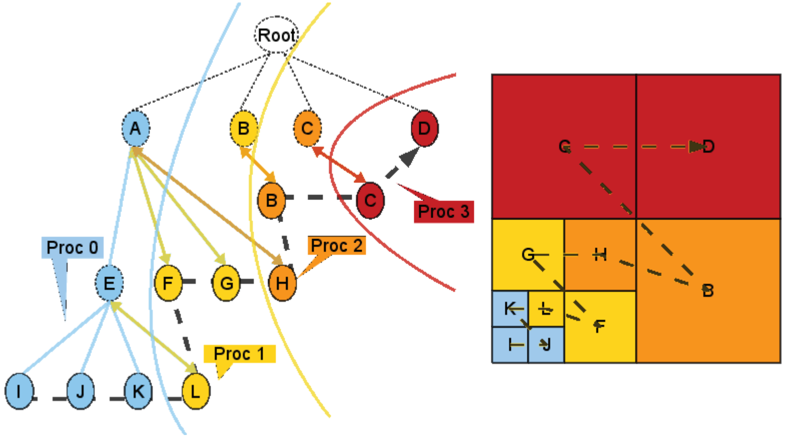


FIG. 2.1. *Parallel quadtree–based adaptive mesh refinement and load balancing. The left diagram is the associated parallel quadtree structure and the right is the actual mesh. Dashed circles represent inactive parent elements and solid circles represent currently active elements on the finest grid. Double-arrow lines represent off-processor parent-child connections. Postorder traversal of the active leafs gives the Lebesgue SFC that connects active elements in the finest mesh. Equal partition of the curve yields the new partition of elements.*

machines relies on binning strategies that group elements on the basis of local error (cf. [16]). At each refinement level, the global maximum local functional found in element τ, $\epsilon^2_{\max,\tau}$, is obtained through a local search in each processor followed by a simple global "all-reduce" communication. Elements are then grouped into bins such that the ith bin contains elements with the local functional in the range

$$(q^{i+1}\epsilon^2_{\max,\tau}, q^i\epsilon^2_{\max,\tau}], \qquad i = 0, 1, 2, \ldots.$$

Here, $0 < q < 1$ is chosen based on the polynomial degree of the finite-element subspace, $\mathcal{V}^h$, such that if an element in bin i is refined and bin $i+1$ is not refined, then child elements from refining bin i will land in the bin $i+1$. In addition, this puts the element with the largest error, $\epsilon_{\max,\tau}$, in the first bin, $i = 0$. Refinement decisions are then made on the basis of treating each bin as an abstract element in order to minimize the *effective error reduction*. Various numerical results show that this parallel ACE (pACE) algorithm produces results similar to the original serial algorithm with low communication cost [16]. On finer levels, almost all error falls in one or two bins, which implies that the error is almost equidistributed. Once the error has become equidistributed, subsequent refinement becomes uniform global refinement, which mitigates load balancing requirements as described below.

To address load balancing issues, the parallel quadtree– (octree– in three dimensions) based mesh [19, 53] is used. Starting on coarser grids, where computation and communication are relatively cheap, and after performing local refinement, a preorder traversal of the associated quadtree generates a Lebesgue SFC (or Morton ordering of the elements). Equal partition of the curve yields the new partition of elements. For example, in Figure 2.1, leafs are ordered as

$$I \to J \to K \to L \to F \to G \to H \to B \to C \to D,$$

which leads to the new partition as illustrated by the picture on the right.

Finally, the parallel quadtree structures are used to help in tracking solutions of time-dependent problems, where each time step may use a different locally refined spatial mesh. Assuming that every time step starts with the same coarsest grid, the tree structures associated with the previous time step's mesh is used to evaluate the previous time step's solution at any point. By using the parent-child tree structure, the finest region in the previous mesh that includes any given set of quadrature nodes is located with low cost. For more details, see [16].

3. Incompressible resistive MHD.

The incompressible resistive MHD equations are a time-dependent, nonlinear set of PDEs that model the movement of charged particles as a plasma [34]. While there are many physical and mathematical models to describe a plasma, the single fluid approach is taken here. As a result, the system is a coupling of the incompressible Navier–Stokes and Maxwell's equations. The primitive variables are defined to be the fluid velocity, $\mathbf{u}$, the fluid pressure, p, the magnetic field, $\mathbf{B}$, the current density, $\mathbf{j}$, and the electric field, $\mathbf{E}$. In addition, a resistive form of Ohm's law,

$$\mathbf{j} = \sigma(\mathbf{E} + \mathbf{u} \times \mathbf{B}), \tag{3.1}$$

is used to eliminate the electric field, $\mathbf{E}$, from the equations. After a nondimensionalization using Alfvén units, the following equations for incompressible resistive MHD are obtained (i.e., Navier–Stokes coupled with Maxwell's equations) [45]:

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} - \mathbf{j} \times \mathbf{B} + \nabla p - \frac{1}{R_e} \nabla^{\mathbf{2}} \mathbf{u} = \mathbf{f}, \tag{3.2}$$

$$\frac{\partial \mathbf{B}}{\partial t} - \mathbf{B} \cdot \nabla \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{B} + \frac{1}{S_L}(\nabla \times \mathbf{j}) = \mathbf{g}, \tag{3.3}$$

$$\nabla \times \mathbf{B} = \mathbf{j}, \tag{3.4}$$

$$\nabla \cdot \mathbf{B} = 0, \tag{3.5}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{3.6}$$

$$\nabla \cdot \mathbf{j} = 0. \tag{3.7}$$

Here, R_e is the fluid Reynolds number and S_L is the Lundquist number, both of which are assumed to be constants and adjusted for different types of physical behavior. The Lundquist number is inversely proportional to the resistivity of the system, and, therefore, large values of this parameter coincide with small resistivities. The lower the resistivity, the more "ideal" the plasma behaves and the more the PDE becomes advection dominated. For the numerical experiments in this paper, the Reynolds number is assumed to be equal to the Lundquist number.

3.1. Island coalescence.

In this section, a test problem that investigates magnetic reconnection in a tokamak fusion model is considered [22, 41, 46, 50, 52]. A reduced set of MHD equations is obtained that models a "large-aspect-ratio" tokamak, with noncircular cross-sections. The magnetic B-field along the z-direction, or the toroidal direction, is very large and mostly constant. In this context, the plasma behavior of interest occurs in the poloidal cross-section. The reduced model allows the tokamak geometry to be simulated on a Cartesian grid by a small annular cross-section in the poloidal direction. See Figure 3.1. Using the FOSLS methodology, the system is first put into a differential first-order system of equations. This is done based on a vorticity-velocity-pressure-current formulation, referring to the dependent variables used to make it first order. Since explicit vorticity boundary conditions are

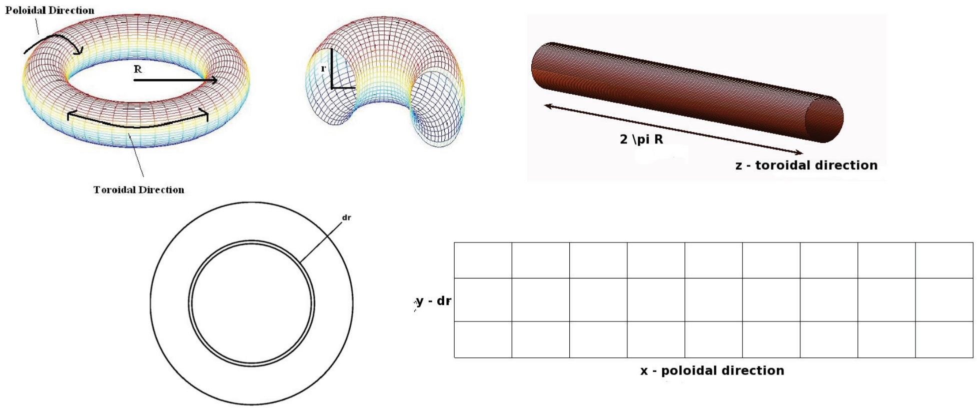


FIG. 3.1. *Torus-shaped tokamak reduced to a two-dimensional (2D) Cartesian grid. The domain is periodic in x, the poloidal direction, and y represents the annular region in the poloidal cross-section.*

provided in all the test problems (shown below), this is an appropriate formulation. A scaling analysis is performed in [3], which yields a nice block structure for the MHD system.

Vorticity, $\boldsymbol{\omega} = \nabla \times \mathbf{u}$, is introduced and the final formulation is

$$\mathcal{L}_1(\mathcal{U}) = \frac{1}{\sqrt{R_e}}\nabla \times \mathbf{u} - \sqrt{R_e}\omega = 0, \tag{3.8}$$

$$\mathcal{L}_2(\mathcal{U}) = \frac{1}{\sqrt{R_e}}\nabla \cdot \mathbf{u} = 0, \tag{3.9}$$

$$\frac{1}{\sqrt{R_e}}\frac{\partial \mathbf{u}}{\partial t} + \mathcal{L}_3(\mathcal{U}) = \frac{1}{\sqrt{R_e}}\frac{\partial \mathbf{u}}{\partial t} - \mathbf{u} \times \boldsymbol{\omega} - \mathbf{j} \times \mathbf{B} - \sqrt{R_e}\nabla p + \frac{1}{\sqrt{R_e}}\nabla^{\perp}\omega = \mathbf{f}, \tag{3.10}$$

$$\mathcal{L}_4(\mathcal{U}) = \frac{1}{\sqrt{S_L}}\nabla \times \mathbf{B} - \sqrt{S_L}j = 0, \tag{3.11}$$

$$\mathcal{L}_5(\mathcal{U}) = \frac{1}{\sqrt{S_L}}\nabla \cdot \mathbf{B} = 0, \tag{3.12}$$

$$\frac{1}{\sqrt{S_L}}\frac{\partial \mathbf{B}}{\partial t} + \mathcal{L}_6(\mathcal{U}) = \frac{1}{\sqrt{S_L}}\frac{\partial \mathbf{B}}{\partial t} + \frac{1}{\sqrt{R_e S_L}}(\mathbf{u} \cdot \nabla \mathbf{B} - \mathbf{B} \cdot \nabla \mathbf{u}) + \frac{1}{\sqrt{S_L}}\nabla^{\perp}j = \mathbf{g}, \tag{3.13}$$

where $\mathcal{U} = (\mathbf{u}, \omega, p, \mathbf{B}, j)^T$ represents all the unknowns in the system. The x-direction denotes the periodic poloidal direction in the tokamak, while the y-direction represents a thin annulus in the poloidal cross-section. In this 2D setting, the vorticity, ω, and the current density, j, are both scalar variables. The vector notation for these variables makes the cross-product well defined: $\boldsymbol{\omega} = (0, 0, \omega)^T$ and $\mathbf{j} = (0, 0, j)^T$. The equations have been scaled using the Reynolds number, R_e, and the Lundquist number, S_L, so that, in the context of a FOSLS discretization, the resultant discrete linear system is more amenable to solution by AMG.

One physical instability that can arise in a tokamak fusion reactor is an island coalescence in the current density arising from perturbations in an initial current density sheet. This instability causes a reconnection in the magnetic field lines and the merging of two islands in the current density field, producing a sharp peak in current density where the magnetic field lines reconnect. This region is known as the reconnection zone, and the point at which the magnetic field lines break is known as the $\mathcal{X}$-point. See [9, 41] for more details. For the simulations shown in this paper, the domain $\Omega = [-1,1] \times [-1,1]$ is used. See Figure 4.1. The initial conditions at equilibrium are

$$\mathbf{B}_0(x,y) = \frac{1}{\cosh(2\pi y) + k\cos(2\pi x)} \begin{pmatrix} \sinh(2\pi y) \\ k\sin(2\pi x) \end{pmatrix}, \tag{3.14}$$

$$\mathbf{u}_0(x,y) = \mathbf{0}, \tag{3.15}$$

$$\omega_0(x,y) = 0, \tag{3.16}$$

$$j_0(x,y) = \nabla \times \mathbf{B}_0 = \frac{2\pi(k^2-1)}{(\cosh(2\pi y) + 0.2\cos(2\pi x))^2}, \tag{3.17}$$

$$p_0(x,y) = \frac{(1-k^2)}{2}\left(1 + \frac{1}{(\cosh(2\pi y) + 0.2\cos(2\pi x))^2}\right), \tag{3.18}$$

where $k = 0.2$. These initial conditions are perturbed away from equilibrium as follows:

$$\delta\mathbf{B}_0(x,y) = \begin{pmatrix} -\epsilon\frac{1}{\pi}\cos(\pi x)\sin(\pi\frac{y}{2}) \\ \frac{1}{2}\epsilon\frac{1}{\pi}\cos(\pi\frac{y}{2})\sin(\pi x) \end{pmatrix}, \tag{3.19}$$

$$\delta j_0(x,y) = \epsilon\cos\left(\pi\frac{y}{2}\right)\cos(\pi x), \tag{3.20}$$

where $\epsilon = -0.01$. The boundary conditions are periodic in x and Dirichlet for the current density and vorticity on the top and bottom of the domain. Also, $\mathbf{n}\cdot\mathbf{u}$ and $\mathbf{n}\cdot\mathbf{B}$ are known on the top and bottom. With these boundary conditions and the simple domain, $H(\text{curl}) \cap H(\text{div}) = H^1$ [32, 33], and, thus, the linearized version of the FOSLS formulation, (3.8)–(3.13), is elliptic in a subspace of a product H^1-norm [5]. No higher regularity for the variables is needed to achieve this, since the weak form comes from a minimization of a quadratic functional. This assumes that the time derivative has been discretized using an implicit time-stepping scheme. For the results that follow, we use a second-order backward differencing formula (BDF-2). Thus, the functional that is minimized for each time step k and Newton step ℓ is (using the notation in (3.8)–(3.13))

$$\begin{aligned}
&||\mathcal{L}_1(\delta\mathcal{U}) + \mathcal{L}_1(\mathcal{U}^k_{\ell-1})||_0^2 + ||\mathcal{L}_2(\delta\mathcal{U}) + \mathcal{L}_2(\mathcal{U}^k_{\ell-1})||_0^2 \\
&+ \left|\left|\frac{3}{2\sqrt{R_e}\Delta t}\delta\mathbf{u} + \mathcal{L}_3(\delta\mathcal{U}) + \mathcal{L}_3(\mathcal{U}^k_{\ell-1}) + \frac{3}{2\sqrt{R_e}\Delta t}(\mathbf{u}^k_{\ell-1} - \frac{4}{3}\mathbf{u}^{k-1} + \frac{1}{3}\mathbf{u}^{k-2}) - \mathbf{f}\right|\right|_0^2 \\
&+ \left|\left|\mathcal{L}_4(\delta\mathcal{U}) + \mathcal{L}_4(\mathcal{U}^k_{\ell-1})\right|\right|_0^2 + \left|\left|\mathcal{L}_5(\delta\mathcal{U}) + \mathcal{L}_5(\mathcal{U}^k_{\ell-1})\right|\right|_0^2 \\
&+ \left|\left|\frac{3}{2\sqrt{S_L}\Delta t}\delta\mathbf{B} + \mathcal{L}_6(\delta\mathcal{U}) + \mathcal{L}_6(\mathcal{U}^k_{\ell-1}) + \frac{3}{2\sqrt{S_L}\Delta t}(\mathbf{B}^k_{\ell-1} - \frac{4}{3}\mathbf{B}^{k-1} + \frac{1}{3}\mathbf{B}^{k-2}) - \mathbf{g}\right|\right|_0^2,
\end{aligned}$$

where $\mathcal{U}^k_\ell = \mathcal{U}^k_{\ell-1} + \delta\mathcal{U}$.

4. Numerical results.

With these boundary conditions, it is shown here that the appropriate physical behavior is efficiently captured by the NI-Newton-FOSLS-ACE-AMG algorithm. A range of Lundquist values from high resistivity to low resistivity is modeled. In this range, variations in the reconnection rate of the system are seen. Here, the reconnection rate is defined as the time rate of change of the poloidal flux function, Ψ, where $\mathbf{B} = -\nabla^\perp\Psi$, evaluated at the $\mathcal{X}$-point [9, 34, 41, 52]. Using the first-order formulation, this is rewritten in terms of the Lundquist number and

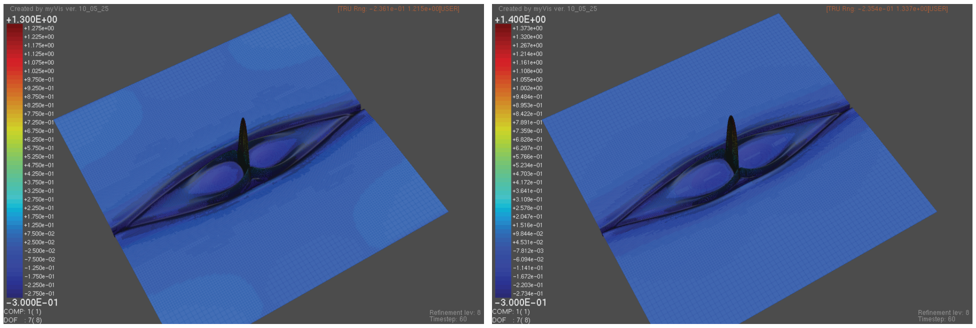


FIG. 4.1. *Current density plot at $t = 6\tau_a$ for $S_L = 5{,}000$ (left) and $S_L = 10{,}000$ (right) using a time step of $\Delta t = 0.1$.*

the current density,

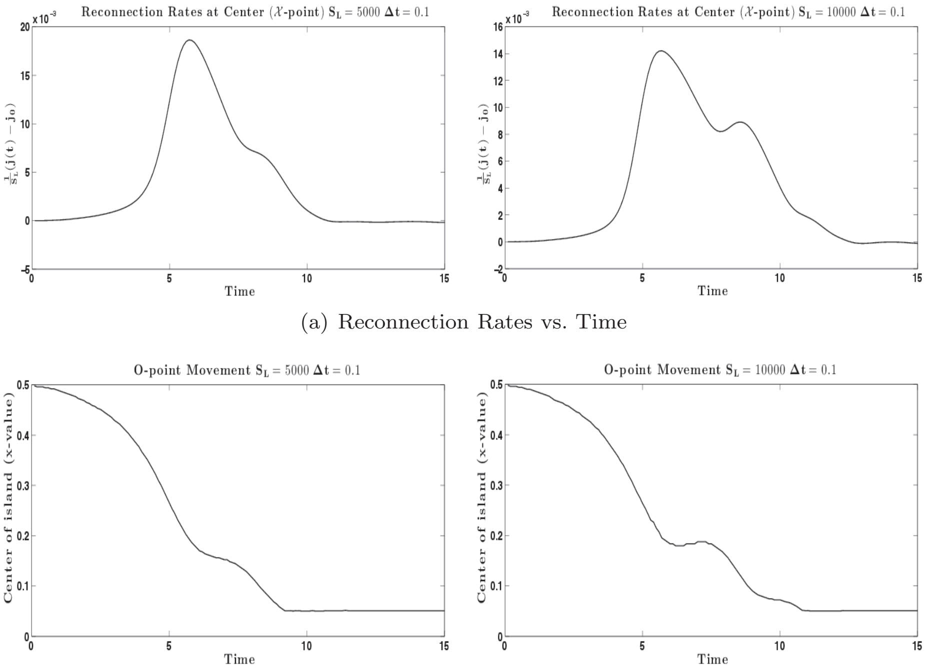
$$\mathcal{R} = \left.\frac{\partial \Psi}{\partial t}\right|_{\mathcal{X}} = \frac{1}{S_L}\left(j(\mathcal{X}) - j_0(\mathcal{X})\right), \tag{4.1}$$

where j_0 is the equilibrium state of the current density. At low Lundquist numbers, the reconnection zone is wider with a less steep gradient in the current density when the peak occurs. As the Lundquist number increases, this reconnection zone narrows, resulting in a sharper, yet shorter, peak. In addition, a "sloshing" effect occurs, where the islands bounce a little before fully merging into one. This yields a peak in the reconnection rate, whose height oscillates as the islands come together. See Figure 4.2.

In all test cases, the problem was run to time $15\tau_A$ with varying time step sizes, using a BDF-2 implicit time-stepping scheme. Here, τ_A is the time in Alfvén units. It is the time needed for an Alfvén wave to travel across the domain [9, 41]. By this time, the islands have begun to coalesce and a sharp peak in current density has formed at the reconnection point.

For the spatial discretization, biquadratic H^1 conforming finite elements are used on a quadrilateral mesh. All simulations were performed on an IBM Blue Gene/L machine using up to 1024 cores in the co-processor mode so that communication and computation can overlap in a way that improves parallel efficiency. The linear system on each refinement level is solved by the conjugate gradient method with a single V(1, 1) AMG cycle used as a preconditioner. BoomerAMG from the HYPRE package developed by Lawrence Livermore National Laboratory was used, with symmetric hybrid Gauss–Seidel (Gauss–Seidel on nodes within the processor and block Jacobi across processors) as the smoother. In the previous results shown in [2, 4], a serial machine was used and, thus, the spatial resolution was limited. With the use of a parallel machine, the results were greatly improved, in both accuracy and efficiency of the methods, by using finer spatial resolutions as well as smaller time step sizes. A more detailed analysis of the parallel FOSLS performance, including scalability and parallel efficiency, is given in [16].

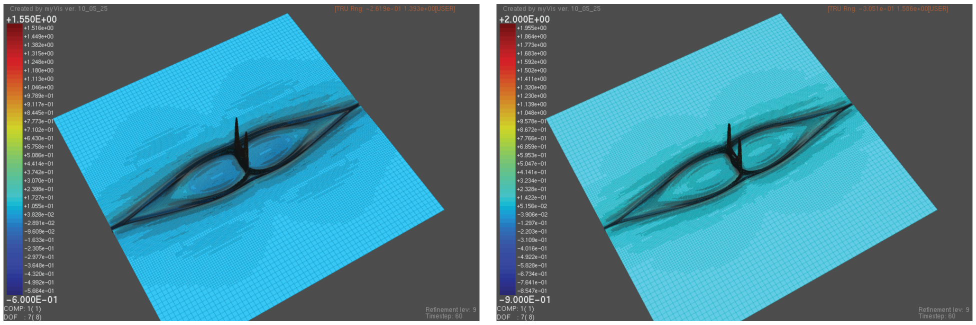
For high resistivities, the NI-Newton-FOSLS-ACE-AMG method is able to capture the reconnection fairly easily. Figure 4.1 shows the current density at time $t = 6\tau_A$, when the peak reconnection occurs. A time step of $0.1\tau_A$ is used. Figure 4.2 shows the reconnection rate versus time for these simulations, as well as the position of the "o-point" or center of the island over time. Notice that, as the Lundquist number is increased, more "sloshing" of the islands occurs.



(a) Reconnection Rates vs. Time

(b) O-point position vs. Time

Fig. 4.2. *Simulations for $S_L = 5{,}000$ (left) and $S_L = 10{,}000$ (right) using a time step of $\Delta t = 0.1$.*

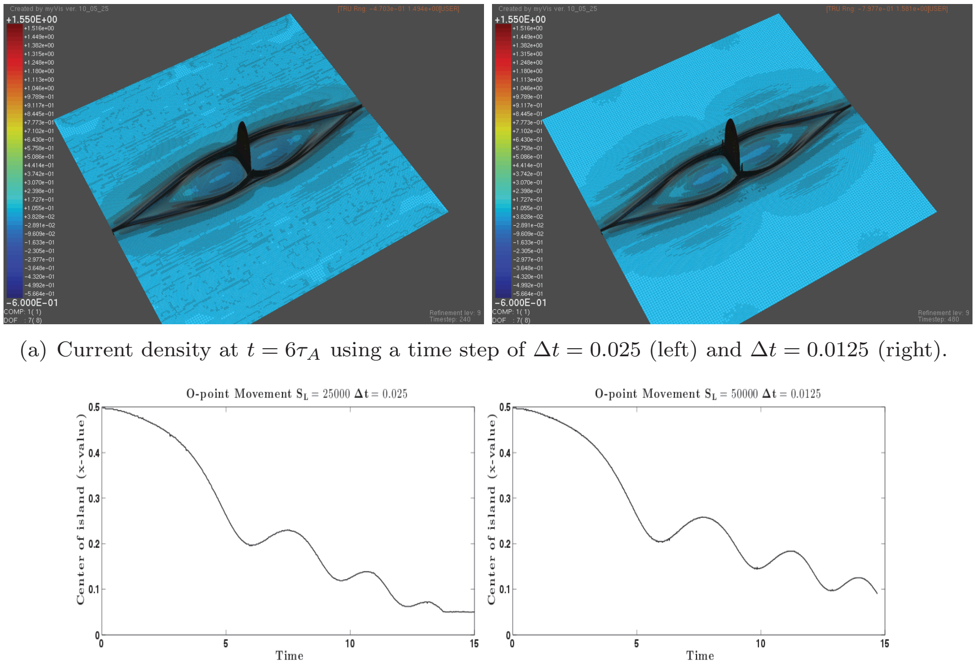


(a) $S_L = 25{,}000$

(b) $S_L = 50{,}000$

Fig. 4.3. *Current density plot at $t = 6\tau_A$ using a time step of $\Delta t = 0.1$.*

When lower resistivities or higher Lundquist numbers are simulated, still using a time step of 0.1, numerical instabilities are introduced. Figure 4.3 shows that, for $S_L = 25{,}000$ and $S_L = 50{,}000$, a double peak occurs. This instability comes from not capturing the solenoidal constraints as accurately as possible, due to a lack of temporal resolution in the simulation [8, 12]. In the FOSLS setting, the system is written as a set of differential algebraic equations (DAEs) [13], where not all variables are time evolved and auxiliary non–time-dependent equations need to be satisfied, such as the solenoidal constraint. Since the FOSLS discretization minimizes the residual of all the equations in the system equally, these auxiliary equations are not solved any more accurately than the rest of the system. As a result, errors in these equations or



(a) Current density at $t = 6\tau_A$ using a time step of $\Delta t = 0.025$ (left) and $\Delta t = 0.0125$ (right).

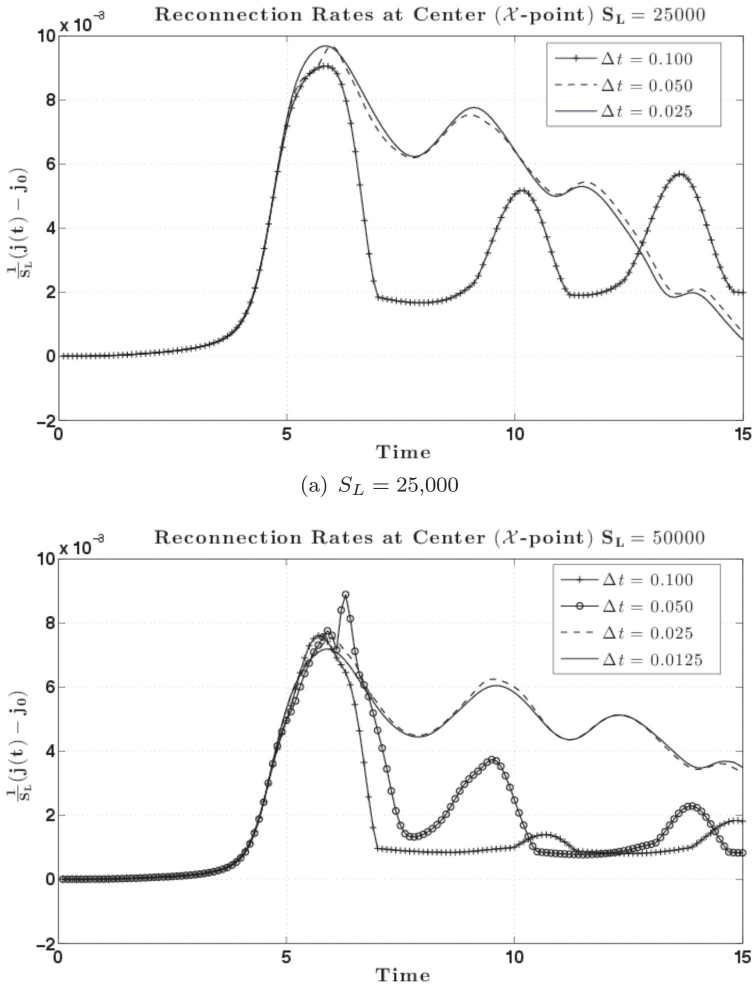
(b) O-point position vs. Time

FIG. 4.4. *Left plots show* $S_L = 25{,}000$ *with* $\Delta t = 0.025$. *Right plots show* $S_L = 50{,}000$ *with* $\Delta t = 0.0125$.

variables may feed back into the time-evolved equations, resulting in a lower accuracy than expected when using a standard implicit time-stepping scheme. However, as the time step size is cut down, the solutions better approximate the reconnection peak. Figure 4.4 shows plots similar to those in Figure 4.3, but with smaller time step sizes. Plots of the "O-point" positions in Figure 4.4 also indicate that the appropriate "sloshing" effect of the islands remains stable.

In addition, the reconnection rates are captured accurately when using the appropriate time step size. Figure 4.5 gives the reconnection rates for $S_L = 25{,}000$ and $S_L = 50{,}000$ using various time step sizes. As Δt gets smaller, the solution more accurately approximates the magnetic reconnection. In Figure 4.6, using the highest resolution simulations, the peak reconnection rate versus Lundquist number is compared and the expected square root decay in maximum peak height of the current density at the reconnection point is seen [41]. This suggests that adaptive time stepping should be explored. This would allow the time step size to be small enough when needed to resolve the physics, or to be larger when that is sufficient to solve the problem, without adding extra computational cost.

Next, the performance of the algorithm is discussed. To understand the efficiency of the method, computational cost is given in terms of a *work unit* (WU), defined to be the amount of work required to perform a single iteration of a simple relaxation method, such as Gauss–Seidel or Jacobi, on the linear system obtained from the Newton-FOSLS method on a nearly optimal grid. A grid is said to be optimal if it yields a discrete solution with the total functional less than a given tolerance, while using a minimal number of degrees of freedom. One objective of the ACE algorithm is to produce a nearly optimal grid. The minimal cost to solve a problem is, then, di-



(a) $S_L = 25{,}000$

(b) $S_L = 50{,}000$

FIG. 4.5. *Reconnection rates using various time step sizes for the low viscosity (high Lundquist number) cases.*

rectly tied to the cost of forming a residual, or performing a simple relaxation method on the linear system corresponding to this optimal grid. This cost is the equivalent of performing one matrix-vector multiplication on this optimal grid. However, at each time step, this measure changes as a different optimal grid is found, depending on the evolving solution. Therefore, in order to compare the results evenly across time steps and across simulations, a "standard" WU is considered. For the simulations presented in this paper, this standard WU is defined on a uniformly refined grid using 1024 by 1024 biquadratic rectangular elements. Thus, since the 2D MHD system has eight unknowns (a stream function is introduced to system (3.8)–(3.13) in order to better enforce incompressibility), this corresponds to a system with over 33 million degrees of freedom. In order to calculate the WUs appropriately, we have to consider the costs of the linear solver. For all of the parallel runs, we use CLJP-coarsening (a parallel coarsening algorithm using independent sets) and extended classical modified interpolation [36]. The strength of the strong connection used was 0.25. Grid com-

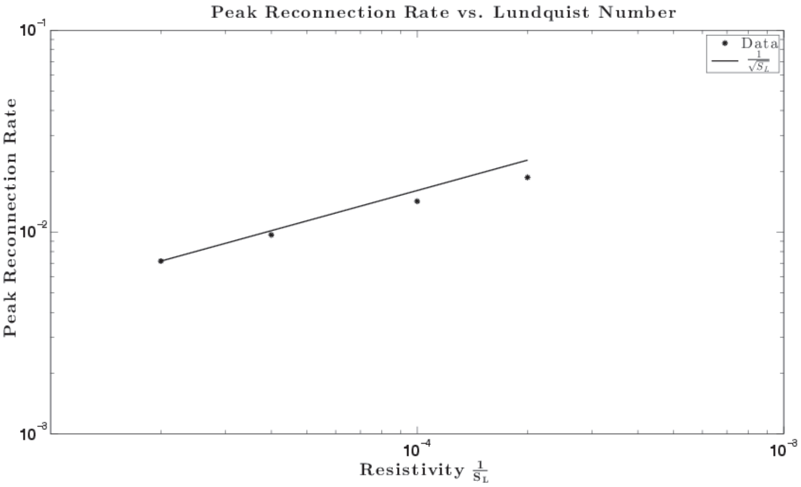
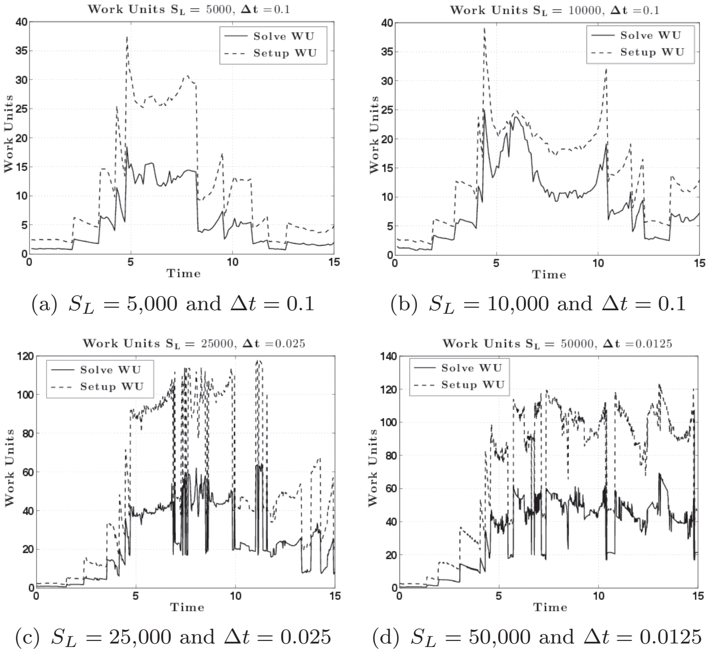


FIG. 4.6. *Peak reconnection rate versus resistivity, $\frac{1}{S_L}$, is proportional to $\frac{1}{\sqrt{S_L}}$.*

plexities ranged from 1.40 to 1.65, which gave operator complexities for a V(1,1) cycle ranging from 2.80 to 3.3. This is lower than the operator complexities using the serial AMG that was implemented using Ruge–Steuben coarsening given that the parallel coarsening algorithm is more aggressive. The total WUs, and also total standardized WUs, are then based on these operator complexities and are equal to (the number of V-cycles)*(the operator complexity). Thus, based on these average operator complexities, each iteration of a V-cycle reported in the following tables is equivalent to about 3 WUs or 3 matrix-vector multiplications on the fine grid. In Figure 4.7, the number of standard WUs needed per time step for the various Lundquist numbers are shown. In each figure, the work due to the setup of the method is separated from the work due to the linear solves, noting that the setup phase is more costly than the solve phase. Also, it is interesting to note that the WUs increase during the time when the reconnection occurs, or when steep gradients are introduced into the solutions.

At higher Lundquist numbers, specifically $S_L = 50{,}000$, AMG performance improves when using a smaller time step. As the temporal resolution is improved, the reconnection is captured more accurately as shown above. In addition, the efficiency of the method improves. When no numerical instabilities are introduced, the approximate solution is smoother and, therefore, the AMG solver performs better. Thus, the number of WUs required to solve the system decreases. Figure 4.8 shows the average number of iterations required to get one digit of accuracy in the solution using AMG at each time step. This is related to the convergence factor of the AMG solves, ρ, by solving for the number of iterations, q, in $\rho^q = 0.1$. Thus, $q = -\frac{1}{\log_{10}(\rho)}$. For the smaller time step size, the average convergence factor and, therefore, the number of iterations are reduced. However, this simulation requires twice as many time steps. Comparing with twice the number of iterations per digit of accuracy at each time step shows that this does not correspond to twice the number of iterations. To make this more concrete, Table 4.1 compares the average standard WUs per time step for the solve over the whole simulation, as well as the maximum amount of standard WUs needed at some time step (usually when the reconnection just starts). Less work is being done with the smaller time step size during the solve phase per time step. During the reconnection, when the maximum amount of work is being performed, halving



(a) $S_L = 5{,}000$ and $\Delta t = 0.1$ (b) $S_L = 10{,}000$ and $\Delta t = 0.1$

(c) $S_L = 25{,}000$ and $\Delta t = 0.025$ (d) $S_L = 50{,}000$ and $\Delta t = 0.0125$

FIG. 4.7. *WUs per time step. Setup cost (dashed curve) and solve cost (solid line) are separated.*

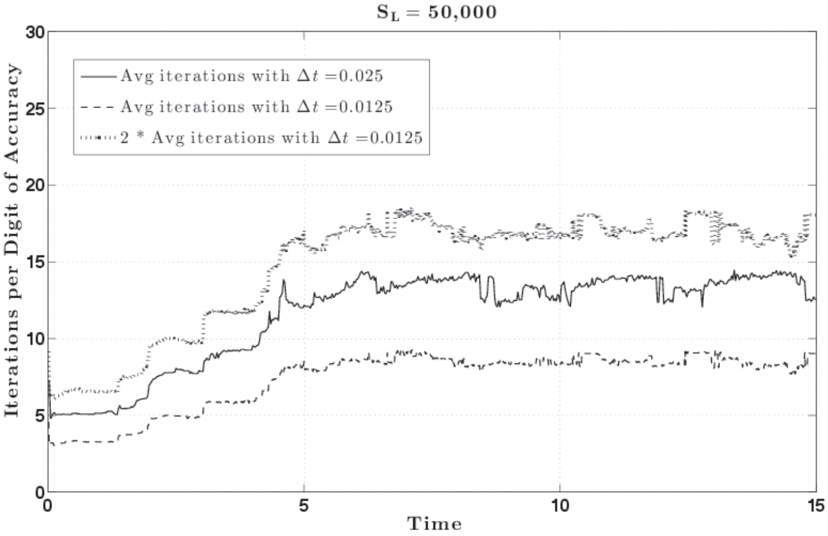


FIG. 4.8. *Average iterations per digit of accuracy for AMG solves at each time step for* $S_L =$ 50,000*. Results for* $\Delta t = 0.025$ *and* $\Delta t = 0.0125$ *are shown as well as twice the number of iterations for* $\Delta t = 0.0125$ *for comparison.*

the time step almost halves the work per time step. For the entire simulation, the ratio of WUs for the smaller time step size to the bigger one is $\frac{2*32.68}{44.15} = 1.48$. Thus, the smaller time step size simulation requires only 48% more work, while performing twice as many time steps. This is because fewer V-cycles are needed per iteration due to better convergence. Since accuracy is better in this case and work is about equal,

Table 4.1

Standard WUs are calculated based on a 1024 by 1024 biquadratic grid for each time step. Average and maximum are computed over all time steps. The average number of V-cycles on the finest grid are also shown to give an idea of what the actual work units are.

S_L = 50,000 Standard WUs			
Δt	Average WU	Max WU	Average V-cycles
0.025	44.15	129.71	16.5
0.0125	32.68	69.32	12.9

Table 4.2

Average number of degree of freedoms (eight unknowns for each node on the grid) and average number of Newton steps at each NI grid level per time step.

	$S_L =$ 5,000 $\Delta t = 0.1000$			$S_L = 10{,}000$ $\Delta t = 0.1000$		
Level	DOFs	Newton	V-cycles	DOFs	Newton	V-cycles
1	8,712	1.993	17.06	8,712	1.993	18.89
2	33,800	1.540	11.86	33,800	1.613	15.37
3	50,935	1.286	10.67	50,700	1.387	11.93
4	103,850	1.213	13.45	98,261	1.360	16.07
5	312,940	1.140	13.87	256,903	1.353	19.56
6	747,591	1.000	13.37	643,268	1.295	22.71
7	1,378,802	1.000	13.55	1,398,085	1.130	22.57
8	2,321,583	1.000	13.47	1,914,774	1.000	21.89
9	7,050,189	1.000	14.31	4,908,584	1.000	21.88

	$S_L = 25{,}000$ $\Delta t = 0.0250$			$S_L = 50{,}000$ $\Delta t = 0.0125$		
Level	DOFs	Newton	V-cycles	DOFs	Newton	V-cycles
1	33,800	1.998	12.60	33,800	1.999	10.31
2	133,128	1.543	7.92	133,128	1.083	4.84
3	184,355	1.000	7.45	189,181	1.000	4.56
4	320,833	1.614	13.25	303,223	1.722	10.20
5	618,366	1.236	12.54	484,576	1.574	10.95
6	877,089	1.495	18.44	799,240	1.606	16.52
7	1,574,753	1.000	15.25	1,592,414	1.000	15.59
8	3,480,815	1.000	14.51	2,965,895	1.000	13.90
9	9,869,081	1.000	14.80	8,837,814	1.000	14.01
10	29,946,167	1.000	10.64	27,971,981	1.000	12.88

the overall accuracy per computational cost is much better and the system is solved more efficiently. As stated earlier, future research will address adaptive time stepping in order to further reduce the computational cost.

Furthermore, the effectiveness of the nested iteration algorithm is analyzed. Over all time steps, the average number of degrees of freedom needed at each grid level is calculated, as is the average number of Newton steps required to solve the nonlinear system at that level. Here, Level 1 corresponds to the coarsest grid. As can be seen in Table 4.2, the average number of Newton steps approaches one step before the finest grid is reached, where the Newton iterations are the most computationally expensive. Thus, the extra overhead of many linearizations is avoided by performing this work on coarser grids, which involve many fewer elements. Also, the average number of V-cycles performed at each level is given. At the finest level, only a handful of cycles are needed. Thus, in terms of actual WUs (i.e., the smallest amount of work needed to form the residual on the finest grid of the given time step), the problem is being solved quite effectively.

Finally, performance of the pACE algorithm and the load balancing method are discussed. Details of the strong scalability and weak scalability results for the pACE

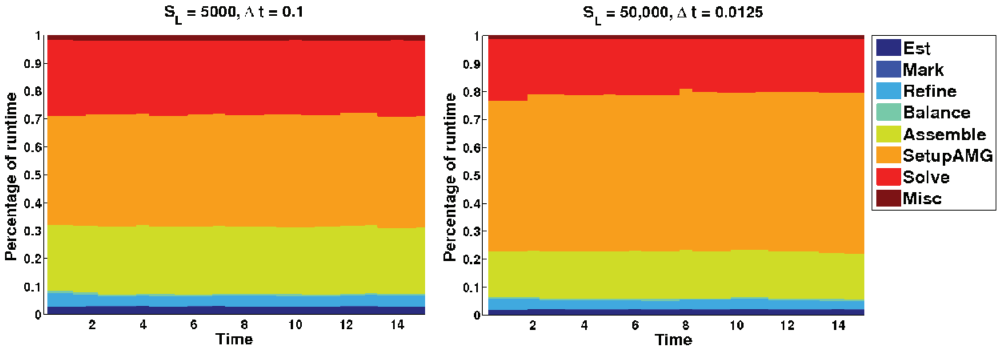


FIG. 4.9. *Breakdown of total runtime into different components related to ACE routines (light and dark blue) and numerical PDE routines (green, yellow, orange, and red) at each time step for* $S_L =$ 5,000 *(left) and* $S_L =$ 50,000 *(right). Color is available only in the online version.*

approach applied to various problems can be found in [16]. Due to the great complexity required to solve the MHD system and hardware limitations, scalability results for the MHD system are not listed here. Instead, for various Lundquist numbers and time step sizes, a breakdown of the overall runtime is plotted into two major categories: numerical PDE solves (assembly of matrices, AMG solver setup, and linear system solves) and adaptive refinement routines (estimating error, marking, refining, and load balancing); see Figure 4.9. The cost of all ACE routines is controlled within 10% of the overall runtime for all time steps and Lundquist numbers. In particular, the CPU time corresponding to the load balancing cost (the light blue strip) is barely recognized, which confirms the efficiency of the load balancing approach applied to complex systems such as the MHD equations. Even for a low Lundquist number, $S_L =$ 5,000, which requires fewer Newton iterations and AMG cycles to solve the problem, the overall cost of the pACE algorithm and the load balancing is a lot smaller than the numerical PDE solves, 10% versus 90%. It should also be noted that a switch was included in cases where a high number of elements were being marked to be refined (about 99%). In these instances, uniform refinement was performed instead, as it involved less overhead. Since memory was an issue on the machine used, a bit of performance was sacrificed in order to improve the accuracy. However, as seen in Figure 4.9, grid repartition took up a negligible amount of the computation time. Determining the most effective percentage at which to switch is the subject of future work. Thus, the parallel scalability of the NI-Newton-FOSLS-AMG-ACE approach is mainly determined by the three components of the numerical PDE solves: assembling matrices, AMG setup, and AMG solves. The first component is almost embarrassingly parallel. Matrices are assembled blockwise within each processor without communication. The third component usually has nice parallel scalability up to tens of thousand of processors [6, 7]. The issue, then, is the second component, AMG setup. Depending on the specific problem, it can be expensive. For example, for high Lundquist numbers (right side of Figure 4.9), AMG setup takes roughly 50% of the overall simulation time, since more Newton iterations are required, and the setup has to be redone for every Newton step. Therefore, improving AMG setup in the context of NI remains an open problem. One option is to reuse the coarsening hierarchy across Newton steps and possibly time steps as well. However, in the context of NI, the solutions tend to change quickly across linearization as more features are introduced and, therefore, reusing the setup is not so useful. In addition, most of the linearizations are done on coarse grids where this setup is cheap. On the finest grids, where this could be beneficial, only one Newton step is used on average anyway. For using the same coarsening across time steps, the solution could change rapidly with respect to time and, then, the hierarchy of adaptive grids across time steps would change drastically and, therefore, should be independent of each other. However, as shown above, reducing the time step size improves the efficiency of the NI algorithm by using fewer nonlinear iterations and, thus, less of these setup phases are needed.

5. Discussion. Using the NI-Newton-FOSLS-ACE-AMG scheme, reconnection rates are accurately captured. Using an appropriate time step size and spatial resolution, the discrete systems are solved using about 50–60 standard WUs per time step, or the equivalent of that many matrix-vector operations on a linear system with 33 million degrees of freedom. The key features are the efficiency-based local adaptive refinement and NI, which allow for most of the computationally expensive work to be performed on coarse grids, reducing the cost. Also, implementation of a scalable parallel scheme for these methods allows for the system to be solved much more efficiently. Care is taken in parallelization of the adaptive refinement routine so that load balancing issues are minimized and so that the MHD system is solved with much better resolution.

For higher values of the Lundquist number, numerical instabilities appear when the spatial and temporal scales are not resolved. With the aid of a parallel machine, however, appropriate accuracy is regained and the correct physical results are obtained. It is also demonstrated that the numerical methods are converging asymptotically at their expected rates. As stated above, the issue comes from the fact that the FOSLS formulation yields a system of DAEs, and more care has to be taken in the analysis of the time-stepping accuracy. The reduced, resistive, MHD system used above is classified as a nonlinear semiexplicit index-two DAE. It is semiexplicit in the sense that none of the nonlinearities involves the time derivatives. In other words, the time derivatives, $\frac{\partial \mathbf{u}}{\partial t}$ and $\frac{\partial \mathbf{B}}{\partial t}$, can be explicitly solved for in terms of the other variables. The index of a DAE is the minimal number of constraint equation differentiations needed to get explicit time-evolving equations for all the dependent variables. In [13], it is shown that backward differencing formulas of order $k < 7$ are convergent and accurate for DAE systems of this type assuming the constraint equations are solved to a sufficient accuracy. This affects any discretization method that introduces auxiliary equations into the system (for instance, mixed methods). However, with the numerical techniques used here, the accuracy of the approximation is easily measured. Future work will examine how the different first-order formulations affect the accuracy of the time-stepping schemes in this manner. Preliminary analysis on using FOSLS for the time-dependent Stokes equations shows that the stability is dependent on the time step size and Reynolds number. Since the MHD system shares many of the properties of the Stokes system, it is not surprising that, as the Lundquist number is increased, the time step must be decreased in order to get the expected stability of the numerical scheme. The plan is to investigate this further and find a better relationship between these parameters. In addition, the use of an adaptive time-stepping scheme would greatly improve the results. Only during the time interval when the reconnection occurs is the higher temporal accuracy needed. Thus, outside this interval, larger time steps could be taken, reducing the cost of solving the system. Also, similar to the AMR schemes employed, the error estimators could be used to optimize the appropriate time step size that is needed to resolve the solution. Time adaptivity has been studied in various complex fluid problems with success (e.g., in [31, 40]).

Future work will involve automating these processes to choose nearly optimal grids and time step sizes for every step of the algorithm based on the system and how well it is being solved. Nevertheless, as shown above, cutting the time step size in half, thus doubling the number of time steps, did not double the computational cost.

Another aspect to consider is that these physical systems are governed by an energy law that is satisfied exactly at the continuous level. Any discretization method can only approximate this energy law, and, therefore, it is important to analyze how this affects the numerical solution. For instance, understanding how accurately one needs to solve a solenoidal or incompressibility constraint can be studied by looking at the "discrete" energy laws that are produced. Future work will look at the FOSLS discretization and attempt to understand how well it approximates the energetics of the system. This includes considering symplectic time integrators such as Crank–Nicolson and the midpoint rule [29, 48].

Finally, a future goal is to apply these methods to MHD in more complex geometries and to introduce more physics into the model, such as Hall terms and electron inertia [23, 34, 45]. These more complex equations are more accurate for models of fusion reactors and in space physics [11]. The algorithms presented in this paper show the potential to efficiently resolve these more complicated models.

REFERENCES

[1] M. F. Adams, R. Samtaney, and A. Brandt, *Toward textbook multigrid efficiency for fully implicit resistive magnetohydrodynamics*, J. Comput. Phys., 229 (2010), pp. 6208–6219.

[2] J. H. Adler, T. A. Manteuffel, S. F. McCormick, J. W. Nolting, J. W. Ruge, and L. Tang, *Efficiency based adaptive local refinement for first-order system least-squares formulations*, SIAM J. Sci. Comput., 33 (2011), pp. 1–24.

[3] J. H. Adler, T. A. Manteuffel, S. F. McCormick, and J. W. Ruge, *First-order system least squares for incompressible resistive magnetohydrodynamics*, SIAM J. Sci. Comput., 32 (2010), pp. 229–248.

[4] J. H. Adler, T. A. Manteuffel, S. F. McCormick, J. W. Ruge, and G. D. Sanders, *Nested iteration and first-order system least squares for incompressible, resistive magnetohydrodynamics*, SIAM J. Sci. Comput., 32 (2010), pp. 1506–1526.

[5] J. H. Adler, *Nested Iteration and First-Order Systems Least Squares for Incompressible Resistive Magnetohydrodynamics*, Ph.D. thesis, University of Colorado at Boulder, Boulder, CO, 2009.

[6] A. H. Baker, R. D. Falgout, T. V. Kolev, and U. M. Yang, *Multigrid smoothers for ultraparallel computing*, SIAM J. Sci. Comput., 33 (2011), pp. 2864–2887.

[7] A. H. Baker, R. D. Falgout, T. V. Kolev, and U. M. Yang, *Scaling hypre's multigrid solvers to* 100,000 *cores*, in High-Performance Scientific Computing, M. W. Berry, K. A. Gallivan, E. Gallopoulos, A. Grama, B. Philippe, Y. Saad, and F. Saied, eds., Springer, London, 2012, pp. 261–279.

[8] T. Barth, *On the role of involutions in the discontinous Galerkin discretization of Maxwell and magnetohydrodynamic systems*, in Compatible Spatial Discretizations, IMA Vol. Math. Appl. 142, Springer, New York, 2006, pp. 69–88.

[9] G. Bateman, *MHD Instabilities*, The MIT Press, Cambridge, MA, 1978.

[10] M. Berndt, T. A. Manteuffel, and S. F. McCormick, *Local error estimates and adaptive refinement for first-order system least squares (FOSLS)*, Electron. Trans. Numer. Anal., 6 (1997), pp. 35–43.

[11] J. Birn, J. Drake, M. Shay, B. Rogers, R. Denton, M. Hesse, M. Kuznetsova, Z. Ma, A. Bhattacharjee, A. Otto, and P. Prichett, *Geospace environmental modeling (GEM) magnetic reconnection challenge*, J. Geophys. Res., 106 (2001), pp. 3715–3719.

[12] J. U. Brackbill and D. C. Barnes, *The effect of nonzero* $\nabla \cdot \mathbf{B}$ *on the numerical solution of the magnetohydrodynamic equations*, J. Comput. Phys., 35 (1980), pp. 426–430.

[13] K. E. Brenan, S. L. Campbell and L. R, Petzold, *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*, SIAM, Philadelphia, 1995.

[14] S. C. Brenner and L. R. Scott, *Mathematical Theory of Finite Element Methods*, 2nd ed., Springer, New York, 2002.

[15] J. Breslau, N. Ferraro, and S. Jardin, *Some properties of the m3d-c[sup 1] form of the three-dimensional magnetohydrodynamics equations*, Phys. Plasmas, 16 (2009), 092503.

[16] M. Brezina, J. Garcia, T. Manteuffel, S. McCormick, J. Ruge, and L. Tang, *Parallel adaptive mesh refinement for first-order system least squares*, Numer. Linear Algebra Appl., 19 (2012), pp. 343–366.

[17] F. Brezzi and M. Fortin, *Mixed and Hybrid Finite Elements Methods*, Springer Ser. Comput. Math. 15, Springer-Verlag, New York, 1991.

[18] W. L. Briggs, V. E. Henson, and S. F. McCormick, *A Multigrid Tutorial*, 2nd ed., SIAM, Philadephia, 2000.

[19] C. Burstedde, O. Ghattas, M. Gurnis, G. Stadler, E. Tan, T. Tu, L. C. Wilcox, and S. Zhong, *Scalable adaptive mantle convection simulation on petascale supercomputers*, in Proceedings of the ACM/IEEE Conference on Supercomputing (SC '08), 2008.

[20] Z. Cai, R. Lazarov, T. A. Manteuffel, and S. F. McCormick, *First-order system least squares for second-order partial differential equations: Part* I, SIAM J. Numer. Anal., 31 (1994), pp. 1785–1799.

[21] Z. Cai, T. A. Manteulffel, and S. F. McCormick, *First-order system least squares for second-order partial differential equations: Part* II, SIAM J. Numer. Anal., 34 (1997), pp. 425–454.

[22] L. Chacón, *An optimal, parallel, fully implicit Newton–Krylov solver for three-dimensional viscoresistive magnetohydrodynamics*, Phys. Plasmas, 15 (2008), 056103.

[23] L. Chacón and D. Knoll, *A 2D high-β Hall MHD implicit nonlinear solver*, J. Comput. Phys., 188 (2003), pp. 573–592.

[24] L. Chacón, D. A. Knoll, and J. M. Finn, *An implicit, nonlinear reduced resistive MHD solver*, J. Comput. Phys., 178 (2002), pp. 15–36.

[25] L. Chacón, D. A. Knoll, and J. M. Finn, *Nonlinear study of the curvature-driven parallel velocity shear-tearing instability*, Phys. Plasmas, 9 (2002), pp. 1164–1176.

[26] F. F. Chen, *Introduction to Plasma Physics*, Plenum, New York, 1974.

[27] A. L. Codd, T. A. Manteuffel, and S. F. McCormick, *Multilevel first-order system least squares for nonlinear elliptic partial differential equations*, SIAM J. Numer. Anal., 41 (2003), pp. 2197–2209.

[28] A. L. Codd, T. A. Manteuffel, S. F. McCormick, and J. W. Ruge, *Multilevel first-order system least squares for elliptic grid generation*, SIAM J. Numer. Anal., 41 (2003), pp. 2210–2232.

[29] M. d'Aquino, C. Serpico, and G. Miano, *Geometrical integration of Landau–Lifshitz–Gilbert equation based on the mid-point rule*, J. Comput. Phys., 209 (2005), pp. 730–753.

[30] H. De Sterck, T. A. Manteuffel, S. F. McCormick, J. W. Nolting, J. W. Ruge, and L. Tang, *Efficiency-based h- and hp-refinement strategies for finite element methods*, Numer. Linear Algebra Appl., 15 (2008), pp. 89–114.

[31] H. Elman, M. Mihajlović, and D. Silvester, *Fast iterative solvers for buoyancy driven flow problems*, J. Comput. Phys., 230 (2011), pp. 3900–3914.

[32] V. Girault and P.-A. Raviart, *Finite Element Qpproximation of the Navier-Stokes Equations*, Lecture Notes in Math. 749, Springer-Verlag, Berlin, 1979.

[33] V. Girault and P.-A. Raviart, *Finite Element Methods for Navier-Stokes Equations: Theory and Algorithms*, Springer Ser. Comput. Math. 5, Springer-Verlag, Berlin, 1986.

[34] R. J. Goldston and P. H. Rutherford, *Introduction to Plasma Theory*, Institute of Physics Publishing, London, 1995.

[35] D. S. Harned and D. D. Schnack, *Semi-implicit method for long time scale magnetohydrodynamic computations in three dimensions*, J. Comput. Phys., 65 (1986), pp. 57–70.

[36] V. E. Henson and U. M. Yang, *Boomeramg: A parallel algebraic multigrid solver and preconditioner*, Appl. Numer. Math., 41 (2000), pp. 155–177.

[37] P. Houston, D. Schötzau, and X. Wei, *A mixed DG method for linearized incompressible magnetohydrodynamics*, J. Sci. Comput., 40 (2009), pp. 281–314.

[38] O. Jones, U. Shumlak, and D. Eberhardt, *An implicit scheme for nonideal magnetohydrodynamics*, J. Comput. Phys., 130 (1997), pp. 231–242.

[39] A. N. Kaufman, *Plasma Physics in Theory and Application*, McGraw–Hill, New York, 1966.

[40] D. A. Kay, P. M. Gresho, D. F. Griffiths, and D. J. Silvester, *Adaptive time-stepping for incompressible flow part* II: *Navier–Stokes equations*, SIAM J. Sci. Comput., 32 (2010), pp. 111–128.

[41] D. A. Knoll and L. Chacón, *Coalescence of magnetic islands, sloshing, and the pressure problem*, Phys. Plasmas, 13 (2006), 032307.

[42] N. A. Krall and A. W. Trivelpiece, *Principles of Plasma Physics*, McGraw–Hill, New York, 1973.

[43] A. J. Meir and P. G. Schmidt, *Variational methods for stationary MHD flow under natural interface conditions*, Nonlinear Anal., 26 (1996), pp. 659–689.

[44] A. J. Meir and P. G. Schmidt, *Analysis and numerical approximation of a stationary MHD flow problem with nonideal boundary*, SIAM J. Numer. Anal., 36 (1999), pp. 1304–1332.

[45] D. R. Nicholson, *Introduction to Plasma Theory*, John Wiley and Sons, New York, 1983.

[46] B. Philip, L. Chacón, and M. Pernice, *Implicit adaptive mesh refinement for 2D reduced resistive magnetohydrodynamics*, J. Comput. Phys, 227 (2008), pp. 8855–8874.

[47] D. R. Reynolds, R. Samtaney, and C. S. Woodward, *A fully implicit numerical method for single-fluid resistive magnetohydrodynamics*, J. Comput. Phys., 219 (2006), pp. 144–162.

[48] R. Rieben, D. White, and G. Rodrigue, *High-order symplectic integration methods for finite element solutions to time dependent Maxwell equations*, IEEE Trans. Antennas and Propagation, 52 (2004), pp. 2190–2195.

[49] J. A. Rossmanith, *An unstaggered, high-resolution constrained transport method for magnetohydrodynamic flows*, SIAM J. Sci. Comput., 28 (2006), pp. 1766–1797.

[50] J. N. Shadid, R. P. Pawlowski, J. W. Banks, L. Chacón, P. T. Lin, and R. S. Tuminaro, *Towards a scalable fully-implicit fully-coupled resistive MHD formulation with stabilized FE methods*, J. Comput. Phys., 229 (2010), pp. 7649–7671.

[51] J. C. Simo and F. Armero, *Unconditional stability and long-term behavior of transient algorithms for the incompressible Navier-Stokes and Euler equations*, Comput. Methods Appl. Mech. Engrg., 111 (1994), pp. 111–154.

[52] H. R. Strauss, *Nonlinear, three-dimensional magnetohydrodynamics of noncircular tokamaks*, Phys. Fluids, 19 (1976), pp. 134–140.

[53] T. Tu, D. O'Hallaron, and O. Ghattas, *Scalable parallel octree meshing for terascale applications*, in Proceedings of the ACM/IEEE Conference on Supercomputing (SC '05), 2005.

[54] R. Verfürth, *A Review of A Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques*, Wiley-Teubner, Chichester, 1995.

[55] X. Zhang, *Multilevel Schwarz methods*, Numer. Math, 63 (1992), pp. 521–539.