An Efficiency-Based Adaptive Refinement Scheme Applied to Incompressible, Resistive Magnetohydrodynamics

J. Adler[1], T. Manteuffel[1], S. McCormick[1], J. Nolting[1], J. Ruge[1], and L. Tang

University of Colorado at Boulder
Department of Applied Mathematics
Boulder, CO

Abstract. This paper describes the use of an efficiency-based adaptive mesh refinement scheme, known as ACE, on a 2D reduced model of the incompressible, resistive magnetohydrodynamic (MHD) equations. A first-order system least squares (FOSLS) finite element formulation and algebraic multigrid (AMG) are used in the context of nested iteration. The FOSLS a posteriori error estimates allow the nested iteration and ACE algorithms to yield the best accuracy-per-computational-cost. The ACE scheme chooses which elements to add when interpolating to finer grids so that the best error reduction with the least amount of cost is obtained, when solving on the refined grid. We show that these methods, applied to the simulation of a tokamak fusion reactor instability, yield approximations to solutions within discretization accuracy using less than the equivalent amount of work needed to perform 10 residual calculations on the finest uniform grid.

Key words: magnetohydrodynamics, adaptive mesh refinement, algebraic multigrid, nested iteration

1 Introduction

Magnetohydrodynamics (MHD) is a model of plasma physics that treats the plasma as a charged fluid. As a result, the set of partial differential equations that describe this model are a time-dependent, nonlinear system of equations. Thus, the equations can be difficult to solve and efficient numerical algorithms are needed. This paper shows the use of such an efficient algorithm on the incompressible, resistive MHD equations. A first-order systems least-squares (FOSLS) [1, 2] finite element discretization is used along with nested iteration and algebraic multigrid (AMG) [3–8]. The main focus of this paper is to show that if an efficiency-based adaptive mesh refinement (AMR) scheme is used, within the nested iteration algorithm, then a nonlinear system of equations, such as the MHD equations, can be solved in only a handful of work units per time step. Here, a work unit is defined as the equivalent of one relaxation sweep on the finest grid. In other words, the accuracy-per-computational-cost for solving the MHD equations can be maximized by the use of nested iteration and AMR. As is shown in the results section, we were able to resolve an island coalescence instability in less than 10 work units per time step.

The MHD system and the FOSLS methodology applied to it are discussed in detail in the companion paper [9], so we include only a brief description here in section 2. The nested iteration algorithm has also been described in [10] and [11], so we only briefly discuss it here in section 3. This section also discusses the efficiency-based AMR method known as ACE, which was developed in [12–14]. Finally, in section 4, numerical results are shown for a 2D reduced model that simulates plasma instabilities in a tokamak reactor. These results confirm that the AMR algorithm greatly reduces the amount of work needed to solve the MHD systems.

2 The MHD equations and FOSLS formulation

The resistive MHD equations are time-dependent and nonlinear, and involve several dependent variables. The system is a coupling of the incompressible Navier-Stokes and Maxwell's systems. 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,

$$\mathbf{j} = \sigma(\mathbf{E} + \mathbf{u} \times \mathbf{B}), \tag{1}$$

is used to eliminate the electric field, $\mathbf{E}$, from the equations. After a non-dimensionalization using Alfvén units, the following equations for incompressible resistive MHD are obtained (i.e., Navier-Stokes coupled with Maxwell's equations) [15, 16]:

$$\begin{aligned}
\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} - \mathbf{j} \times \mathbf{B} + \nabla p - \frac{1}{R_e} \nabla^{\mathbf{2}} \mathbf{u} &= \mathbf{f}, &(2)\\
\frac{\partial \mathbf{B}}{\partial t} - \mathbf{B} \cdot \nabla \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{B} + \frac{1}{S_L}(\nabla \times \mathbf{j}) &= \mathbf{g}, &(3)\\
\nabla \times \mathbf{B} &= \mathbf{j}, &(4)\\
\nabla \cdot \mathbf{B} &= 0, &(5)\\
\nabla \cdot \mathbf{u} &= 0, &(6)\\
\nabla \cdot \mathbf{j} &= 0. &(7)
\end{aligned}$$

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.

Using the FOSLS method [1, 2], the system is first put into a differential first-order system of equations. This is done based on a vorticity-velocity-pressure-current formulation [17–19]. A scaling analysis is performed in [9] for the full MHD system. The resulting scaling yields a nice block structure of the MHD system, which results in good AMG convergence of the linear systems obtained, while still preserving the physics of the system.

Vorticity, $\boldsymbol{\omega} = \nabla \times \mathbf{u}$, is introduced and the final formulation in 3D used is

$$\begin{aligned}
\frac{1}{\sqrt{R_e}} \nabla \times \mathbf{u} - \sqrt{R_e} \boldsymbol{\omega} &= 0, &(8)\\
\frac{1}{\sqrt{R_e}} \nabla \cdot \mathbf{u} &= 0, &(9)\\
\sqrt{R_e} \nabla \cdot \boldsymbol{\omega} &= 0, &(10)\\
\frac{1}{\sqrt{R_e}} \frac{\partial \mathbf{u}}{\partial t} - \mathbf{u} \times \boldsymbol{\omega} - \mathbf{j} \times \mathbf{B} - \sqrt{R_e} \nabla p + \frac{1}{\sqrt{R_e}} \nabla \times \boldsymbol{\omega} &= \mathbf{f}, &(11)\\
\frac{1}{\sqrt{S_L}} \nabla \times \mathbf{B} - \sqrt{S_L} \mathbf{j} &= 0, &(12)\\
\frac{1}{\sqrt{S_L}} \nabla \cdot \mathbf{B} &= 0, &(13)\\
\sqrt{S_L} \nabla \cdot \mathbf{j} &= 0, &(14)\\
\frac{1}{\sqrt{S_L}} \frac{\partial \mathbf{B}}{\partial t} + \frac{1}{\sqrt{R_e S_L}} (\mathbf{u} \cdot \nabla \mathbf{B} - \mathbf{B} \cdot \nabla \mathbf{u}) + \frac{1}{\sqrt{S_L}} \nabla \times \mathbf{j} &= \mathbf{g}. &(15)
\end{aligned}$$

The above system is denoted by $\mathcal{L}(u) = f$, where $u = (\mathbf{u}, \boldsymbol{\omega}, p, \mathbf{B}, \mathbf{j})^T$ represents a vector of all of the dependent variables that should not be confused with the vector fluid velocity, $\mathbf{u}$. Then, the L^2 norm of the residual of this system is minimized. This is referred to as the nonlinear functional,

$$\mathcal{F}(u) = ||\mathcal{L}(u) - f||_0. \tag{16}$$

In general, we wish to find the argmin of (16) in some solution space $\mathcal{V}$. In the context of this paper, we choose $\mathcal{V}$ to be an H^1 product space with boundary conditions that are chosen to satisfy the physical constraints of the problem as well as the assumptions needed for the FOSLS framework. In practice, a series of nested finite subspaces, $\mathcal{V}^h$, are used to approximate the solution in $\mathcal{V}$. However, in the Newton-FOSLS approach [20, 21], system (8)-(15) is first linearized using a Newton step before a FOSLS functional is formed and minimized. This results in the weak form of the problem that produces symmetric positive definite algebraic systems when the problem is restricted to a finite-dimensional subspace, $\mathcal{V}^h$. In addition, proving continuity and coercivity of the resulting bilinear form is equivalent to having H^1 equivalence of the FOSLS functional. Moreover, the FOSLS functional yields a sharp a posteriori local error estimate, which is used to make the algorithm more robust and, under the right conditions, produces algebraic systems that are solved easily by multilevel iterative solvers. Our choice here is algebraic multigrid (AMG) [3–8], which, when applied to the FOSLS discretization, has been shown to be an optimal ($O(n)$) solver [1, 2, 6]. Using the formulation above, and with appropriate boundary conditions, H^1 equivalence of the linearized FOSLS functional is shown in [11]. Therefore, the FOSLS functionals can be a good measure of the error, or at least the semi-norm of the error, in the solution space. Thus, they can be used to develop an efficient solution algorithm and as aids in the adaptive refinement process. By measuring the functional norm of the error in each element of the domain, information on where refinement is needed is readily available.

3 Solution Algorithm

In [10], an algorithm is devised to solve a system of nonlinear equations, $\mathcal{L}(u) = f$. Starting on a coarse grid, given an initial guess, the system is linearized and the linearized FOSLS functional is then minimized on a finite element space. At this point, several AMG V-cycles are performed until there is little to gain in accuracy-per-computational-cost. The system is then relinearized and the minimum of the new linearized FOSLS functional is searched for in the same manner. After each set of linear solves, the relative difference between the computed linearized functional and the nonlinear functional is checked. If they are close and near the minimum of the linearized functional, then it is concluded that we are close enough to the minimum of the nonlinear functional and, hence, we have a good approximation to the solution on the given grid. Next, the approximation is interpolated to a finer grid and the problem is solved on that grid. This process is repeated until an acceptable error has been reached, or until we have run out of computational resources, such as memory. If, as in the case of the MHD equations, it is a time-dependent problem, the whole process is performed at each time step. This algorithm is summarized in the flow chart, figure 1.

3.1 Adaptive Refinement

In the nested iteration algorithm, we decide when to stay on a current mesh and iterate further or interpolate to a finer grid. In [10], it was assumed that the grid is uniformly refined. In other words, it was assumed that there are 2^d more points on the next grid than the one before, where d is the dimension of the problem. This is generally not the case when the grids are refined

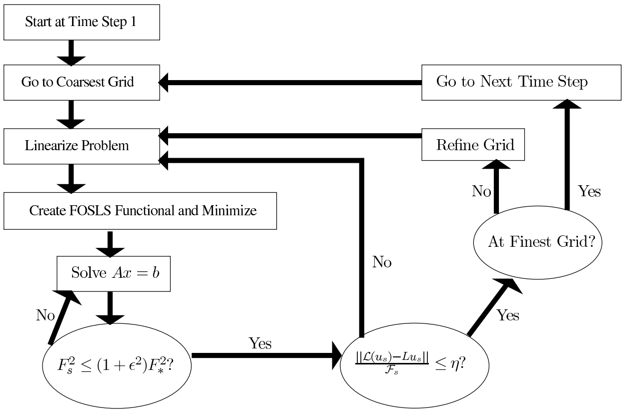

Fig. 1. Flow chart of nested iteration algorithm.

locally. On a given mesh, after enough Newton steps and linear iterations have been performed, the nonlinear functional is calculated in each element. This indicates in which region of the domain the functional and, hence, the error is large compared to the rest of the domain. Then, the best use of new degrees of freedom is to concentrate them where the error is large. Since the goal of the algorithm is to increase the accuracy-per-computational-cost, we do not want to over solve in areas where the error is already small. The adaptive scheme that we describe here is an efficiency based refinement method, called ACE, that was developed in [12–14]. This scheme estimates both the reduction in the functional and the computational cost that would result from any given refinement pattern. These estimates are used to establish a refinement pattern that attempts to optimize the Accuracy-per-Computational cost (Efficiency), which gives rise to the acronym ACE.

The square of the functional value on each element, ϵ_i, is computed and is ordered such that the local functional value is decreasing:

$$\epsilon_1 \geq \epsilon_2 \geq \ldots \geq \epsilon_{N_l}, \tag{17}$$

where N_l is the total number of elements on level l. Next, we predict the reduction of the squared functional and the estimated computational work that would result if we were to refine a given percentage of the elements with the most error. Denote the percentage by $r \in (0,1]$ and the number of elements to be refined by $r * N_l$. Define $f(r)$ to be the squared functional value in the $r * N_l$ elements with largest error, that is,

$$f(r) = \frac{\sum_{i<r*N_l} \epsilon_i}{\sum_{i=1}^{N_l} \epsilon_i}. \tag{18}$$

Here, $f(r)$ is monotonically increasing, concave down, with values on $[0,1]$. For example, figure 2 shows the $f(r)$ that would result from uniformly distributed error and the $f(r)$ that describes error in only a few elements.

Then, we predict that the squared functional is reduced by the factor

$$\gamma(r) = 1 - f(r) + (\frac{1}{2})^{2p} f(r). \tag{19}$$

Here, p is the order of the finite element basis. In other words, the predicted error reduction only comes from the regions where the elements have been refined. The predicted error in the

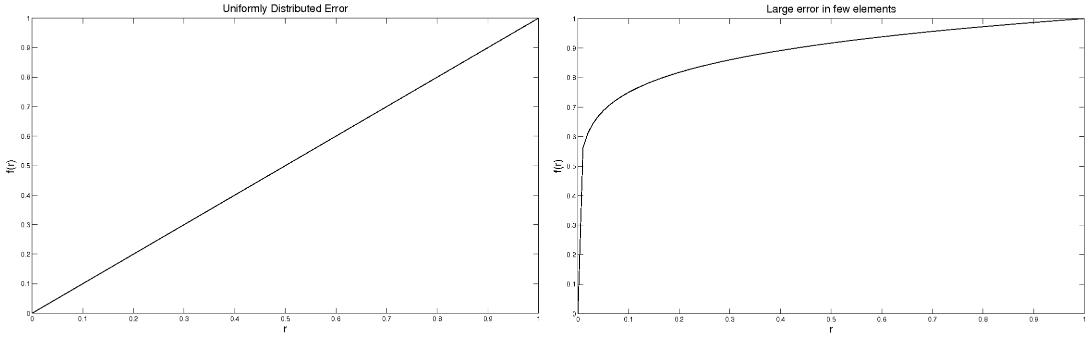

Fig. 2. Error distribution among elements. Left plot shows uniformly distributed error. Right plot shows distribution where a few elements contain most of the error.

remaining elements is left unchanged. In general, this underestimates the reduction. In practice, the unrefined elements can have some error reduction as well.

On the other hand, the predicted work is assumed to be a function of the total number of elements on the refined grid. Since each refined element yields 2^d children, we predict the number of elements on the refined grid to be

$$N_{l+1} = N_l(1 - r + 2^d r). \tag{20}$$

The work required to solve the linear system on the refined grid is assumed to be a function of the number of elements, N_{l+1}, the error reduction factor, $\gamma(r)$, and the AMG convergence factor, ρ. Assume that the cost of one AMG V-cycle is C_1 work units and the overall setup cost for FOSLS and AMG is C_0 work units. The number of V-cycles needed to reduce the error by $\gamma(r)$ using AMG is

$$n = \frac{log\gamma}{log\rho}. \tag{21}$$

Thus, the overall predicted work of solving on a refined grid is

$$W(r) = (C_0 + C_1\frac{log\gamma}{log\rho})N_{l+1}. \tag{22}$$

Replacing N_{l+1} using (26) yields

$$W(r) = (C_0 + C_1\frac{log\gamma}{log\rho})(1 - r + 2^d r)N_l. \tag{23}$$

With these relations in mind, we choose the r that minimizes the predicted effective functional reduction,

$$\gamma(r)^{eff} = \gamma(r)^{1/W(r)}. \tag{24}$$

Therefore, we refine only $r * N_l$ elements on each level that gives us the best error reduction for the added cost.

In addition, one could allow for multiple refinements of each element. For example, the ACE scheme could call for $r_1 N_l$ elements to be refined once and $r_2 N_l$ elements to be refined twice. This changes the predicted error reduction, $\gamma(r_1, r_2)$, to

$$\gamma(r_1, r_2) = 1 - f(r_1) + \left(\frac{1}{2}\right)^{2p}(f(r_1) - f(r_2)) + \left(\frac{1}{2}\right)^{4p} f(r_2). \tag{25}$$

Then, the new number of elements on the refined grid is

$$N_{l+1} = N_l(1 - r_1 + 2^d(r_1 - r_2) + 2^{2d} r_2), \tag{26}$$

and the predicted work estimate, $W(r_1, r_2)$ becomes

$$W(r) = (C_0 + C_1 \frac{log\gamma}{log\rho})(1 - r_1 + 2^d(r_1 - r_2) + 2^{2d} r_2)N_l. \tag{27}$$

Now, the optimal pair, $0 \leq r_1 \leq r_2 \leq 1$ is found to minimize the effective error reduction, equation (24). This allows for more aggressive refinement. In practice, it has been found that using even more refinement, such as triple refinement, is unnecessary.

The ACE scheme fits in nicely with the nested iteration approach. Not only do we try to get the best accuracy-per-computational-cost for each linearization and each AMG cycle, but we also take this into account when interpolating to finer grids. A more detailed explanation of the ACE algorithm can be found in [13, 14]. Numerical results in section 4 show that using adaptive refinement yields the same accuracy in the MHD test problems while using fewer degrees of freedom.

4 Numerical Results

In this section, we investigate an MHD test problem to show that the nested iteration Newton-FOSLS-AMG method works well with the addition of the ACE adaptive mesh refinement. We show that these methods are capable of solving the complex nonlinear systems in a minimal amount of work units, or fine-grid relaxation equivalents. The full algorithm, as in figure 1, was applied to a tokamak test problem [22–25]. From the papers by Chacón, Knoll, and Finn [22] and Philip [24], a reduced set of MHD equations is obtained. These equations simulate a "large aspect-ratio" tokamak, with non-circular cross-sections. Here, the magnetic B-field along the z-direction, or the toroidal direction, is very large and mostly constant. In this context, we are able to look at plasma behavior in the poloidal cross-section. This was described in the companion paper, [9]. The reduced model is equivalent to the 2D version of equations (8)-(15). The x-direction denotes the periodic poloidal direction in the tokamak, whereas the y dimension represents a thin annulus in the poloidal cross section. In this 2D setting, vorticity, ω, and current density, j, are both scalar variables. We now apply our methodology to a test problems known as the island coalescence problem.

4.1 Test Problem: Island Coalescence

This test problem simulates an island coalescence in the current density arising from perturbations in an initial current density sheet. A current density sheet in the toroidal direction of the tokamak is perturbed, resulting in an instability that causes a reconnection in the magnetic field lines and merging of two islands in the current density field. This produces 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 [26, 27, 24] for more details. We choose a low enough resistivity (i.e. Lundquist number above $50,000$) in order to observe the interesting physics. For the following simulations, we define

$$\begin{aligned} \Omega &= [-1, 1] \times [-1, 1], \\ R_e &= S_L = 50,001. \end{aligned}$$

The initial conditions at equilibrium are

$$\mathbf{B}_0(x,y) = \frac{1}{\cosh(2\pi y) + k\cos(2\pi x)} \begin{pmatrix} \sinh(2\pi y) \\ k\sin(2\pi x) \end{pmatrix}, \tag{28}$$

$$\mathbf{u}_0(x,y) = \mathbf{0}, \tag{29}$$

$$\omega_0(x,y) = 0, \tag{30}$$

$$j_{30}(x,y) = \nabla \times \mathbf{B}_0 = \frac{2\pi(k^2-1)}{(\cosh(2\pi y) + 0.2\cos(2\pi x))^2}, \tag{31}$$

$$p_0(x,y) = \frac{(1-k^2)}{2}\left(1 + \frac{1}{(\cosh(2\pi y) + 0.2\cos(2\pi x))^2}\right), \tag{32}$$

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

$$\delta\mathbf{B}_0(x,y) = \begin{pmatrix} -\epsilon\frac{1}{\pi}\cos(\pi x)\sin(\pi\frac{y}{2}) \\ \frac{1}{2}\epsilon\frac{1}{\pi}\cos(\pi\frac{y}{2})\sin(\pi x) \\ 0 \end{pmatrix}, \tag{33}$$

$$\delta j_{30}(x,y) = \epsilon\cos(\pi\frac{y}{2})\cos(\pi x), \tag{34}$$

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. We also have $\mathbf{n}\cdot\mathbf{u}$ and $\mathbf{n}\cdot\mathbf{B}$ known on the top and bottom. Again, the FOSLS formulation, (8)-(15), is H^1 elliptic.

Results The problem was run to time $10\tau_A$ with a timestep of $0.1\tau_A$ using a BDF-2 implicit time-stepping scheme. By the 80th time step, or time $8\tau_A$, the islands have coalesced and the large peak in current density has occurred at the reconnection point. Using both uniform refinement and the ACE scheme, we were able to capture the instability. With ACE employed, the grids evolve over time to refine in areas with steeper gradients. In this problem, as time progresses, a steep gradient occurs at the reconnection point. This is seen in the bottom graph in figure 3. We expect, then, that most of the refinement occurs in this region, which is indeed the case.

To get an idea of how much more efficient adaptive refinement is, we compare the work performed using ACE to that of using uniform refinement. The work at one time step is calculated by first determining the work of all the V-cycles on a given refinement level. Then these values times the number of matrix nonzeros for the level are summed over all grids and divided by the number of nonzeros on the finest refinement level for the given problem. In table 1, the work unit values given are with respect to the finest level of the given refinement scheme. They are an average over all time steps. To compare the two schemes, the average work unit value is multiplied by the fine-grid nonzeros for that scheme and then the ratio is taken. This ratio is defined as the Work Ratio in table 1. Similarly, the Element Ratio column is the ratio of elements on the finest grid of the adaptive scheme compared to the number of elements on the finest grid of the uniform scheme.

The results show that using adaptive refinement greatly reduces the amount of work needed, compared to that of using uniform refinement as is done in [10]. ACE requires 10% of the work that uniform refinement requires. The physics is more localized in this problem, especially by

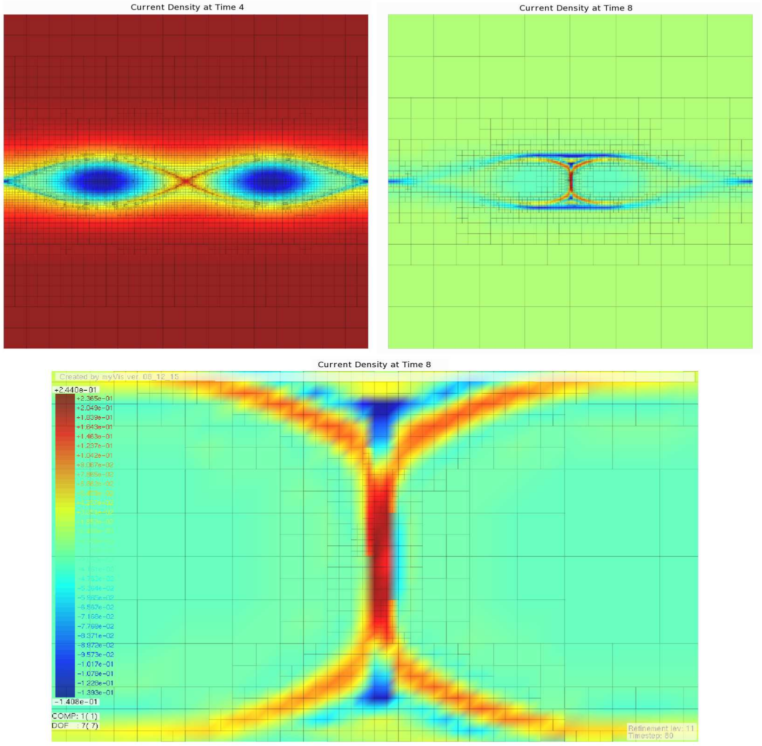

Fig. 3. Numerical solution using 10 levels of adaptive refinement. $S_L = R_e = 50,001$. Top Left: Current Density at Time $4\tau_A$. Top Right: Current Density at Time $8\tau_A$. Bottom: Zoomed in plot of current density peak at Time $8\tau_A$.

Uniform		ACE			
Work Units	Nonzeros	Work Units	Nonzeros	Work Ratio	Element Ratio
66.87	51,781,975	78.23	7,283,047.4	0.16	0.07

Table 1. Average number of work units per timestep using uniform refinement versus ACE refinement. A total of 100 time steps were performed.

time $8\tau_A$ and, thus, the refinement is more localized. Looking at one specific time step, as in table 2, one can see how the nested iteration algorithm along with ACE solves the problem efficiently. By the finest levels, only one Newton step and a handful of work units are needed. The values in parenthesis show the relative work with respect to a grid that was uniformly refined for the given number of levels. Thus, using nested iteration Newton-FOSLS-AMG with ACE yields a good approximation to the solution of the island coalescence problem in less than 10 work units, or the equivalent of 10 relaxation sweeps on a 128 × 128 bi-quadratic uniform grid.

Level	Elements	Nonzeros	Newton Steps	ρ	Work Units	Nonlinear Functional
2	4	17,493	2	0.35	0.054 (0.006)	4.919
3	16	63,063	2	0.55	0.355 (0.038)	1.287
4	40	155,575	1	0.40	0.427 (0.046)	0.604
5	79	299,145	1	0.46	0.682 (0.073)	0.284
6	175	640,871	1	0.58	3.328 (0.356)	0.112
7	337	1,161,741	1	0.50	3.929 (0.420)	0.049
8	658	2,263,261	1	0.79	19.548 (2.090)	0.023
9	1,078	3,624,089	1	0.67	20.789 (2.222)	0.013
10	1,696	5,535,089	1	0.71	29.495 (3.153)	0.008
			Total		78.697 (8.398)	

Table 2. Results for a single time step. AMG convergence factor, ρ, total work units (total work units with respect to uniform refinement in parenthesis), and the nonlinear functional norm are given.

5 Discussion

We showed that using an efficiency-based adaptive refinement scheme, such as ACE, along with the FOSLS finite element method and nested iteration, yields a highly effective method for solving the complicated, current-vorticity form of resistive MHD. Real world MHD applications are solved very efficiently when the focus is on accuracy-per-computational-cost. The use of FOSLS greatly aids this process. Its sharp, a posteriori error estimate allows parameters to be computed that are used to estimate the current accuracy-per-computational-cost. From this, judgements as to what further computation is necessary are made. Such decision making facilitates an efficient local adaptive refinement process, which in turn reduces the amount of cost needed.

Several aspects still need to be studied. First, the linear system solvers are what dictate the overall efficiency of the NI-Newton-FOSLS method. For this paper, the algebraic systems are solved with a classical algebraic multigrid method. Deteriorations in the algebraic convergence for increased timestep size as well as Reynolds and Lundquist numbers are observed. In fact, in table 2, the convergence factor ρ increases slightly as h gets smaller. The current AMG algorithm can be improved in several ways. One might develop an improved AMG for the above type of systems of PDEs. This might involve the use of newly developed adaptive multigrid algorithms described more in [28–30]. In the test problems addressed in this paper, tensor product grids are used, with local refinement. Therefore, one might employ a geometric multigrid solver instead that takes into account the hierarchy of grids used, or one that is based on block structured grids. Specifically, using the same multigrid method to solve on a uniform mesh as for an adaptively refined mesh is not ideal. One would prefer a solver that takes into account the unstructured nature of the grid.

In addition, many aspects of the adaptive refinement algorithm can be improved. In the above tests, the ACE algorithm requires more levels than uniform refinement to reach the same accuracy because it does not refine as aggressively as uniform refinement. Currently, we are examining modifications to ACE that dictate the number of unknowns on the refined grid. If this number is fixed, then the ACE algorithm chooses where to put these new nodes. This is accomplished by allowing ACE to refine an element more than once at each step. This approach allows for fewer refinement steps and, hence, fewer computations overall.

Finally, there are many other MHD problems to be tested, as well as other time-dependent problems in fluid dynamics that have large nonlinearities. Doing most of the hard work on the coarser grids allows us to solve these problems more efficiently. Using a first-order system least squares formulation, we were able to resolve the above MHD physics, and we believe that, with a careful formulation, it can be used for many other time-dependent nonlinear systems. This, with the addition of a parallel implementation, could allow us to tackle even more complicated problems such as Extended or Hall MHD, as well as other complex fluid problems.

Acknowledgments. This work was sponsored by the Department of Energy under grant numbers DE-FG02-03ER25574 and DE-FC02-06ER25784, Lawrence Livermore National Laboratory under contract numbers B568677, and the National Science Foundation under grant numbers DMS-0621199, DMS-0749317, and DMS-0811275.

References

1. Cai, Z., Lazarov, R., Manteuffel, T., McCormick, S.: First-Order System Least Squares for Second-Order Partial Differential Equations. SIAM J. Numer. Anal **31** (1994) 1785–1799
2. Cai, Z., Manteuffel, T., McCormick, S.: First-Order System Least Squares for Second-Order Partial Differential Equations. II,. SIAM J. Numer. Anal **34** (1997) 425–454
3. Brandt, A., McCormick, S.F., Ruge, J.: Algebraic Multigrid (AMG) for automatic multigrid solutions with application to geodetic computations. Report, Inst. for Computational Studies, Fort Collins, CO (1982)
4. Brandt, A., McCormick, S.F., Ruge, J. In: Algebraic Multigrid (AMG) for sparse matrix equations. Cambridge University Press (1984)
5. Brandt, A.: Algebraic Multigrid Theory: The Symmetric Case. Appl. Math. Comput. **19**(1-4) (1986) 23–56
6. Briggs, W.L., Henson, V.E., McCormick, S.F.: A Multigrid Tutorial. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA (2000)
7. Oosterlee, C., Schuller, A., Trottenberg, U.: Multigrid. Academic Press (2000)
8. Ruge, J., St*ü*ben, K.: Algebraic Multigrid (AMG). Multigrid Methods (McCormick, S.F., ed) (1986)
9. Adler, J., Manteuffel, T., McCormick, S.F., Ruge, J.: First-order system least squares for incompressible resistive Magnetohydrodynamics. SIAM J. Sci. Comp. **to appear** (2009)
10. Adler, J., Manteuffel, T., McCormick, S., Ruge, J., Sanders, G.: Nested Iteration and First-Order System Least Squares for Incompressible, Resistive Magnetohydrodynamics. SIAM J. on Sci. Comp. (SISC) (submitted 2009)
11. Adler, J.: Nested Iteration and First Order Systems Least Squares on Incompressible Resistive Magnetohydrodynamics. PhD thesis, University of Colorado at Boulder (2009)
12. Berndt, M., Manteuffel, T., McCormick, S.F.: Local error estimates and adaptive refinement for first-order system least squares (FOSLS). E.T.N.A. **6** (1998) 35–43
13. DeSterck, H., Manteuffel, T., McCormick, S., Nolting, J., Ruge, J., Tang, L.: Efficiency-based h- and hp-refinement strategies for finite element methods. J. Num. Lin. Alg. Appl. **15** (2008) 249–270
14. Nolting, J.: Efficiency-based Local Adaptive Refinement for FOSLS Finite Elements. PhD thesis, University of Colorado at Boulder (2008)
15. Nicholson, D.R.: Introduction to Plasma Theory. John Wiley and Sons, New York (1983)
16. Ullrich, P.: Dynamics and Properties of the Magnetohydrodynamics Equations. unpublished (2005)

17. Bochev, P., Cai, Z., Manteuffel, T., McCormick, S.: Analysis of Velocity-Flux First-Order System Least-Squares Principles for the Navier-Stokes Equations: Part I. SIAM J. Numer. Anal **35** (1998) 990–1009
18. Bochev, P., Cai, Z., Manteuffel, T., McCormick, S.: Analysis of Velocity-Flux First-Order System Least-Squares Principles for the Navier-Stokes Equations: Part II. SIAM J. Numer. Anal **36** (1999) 1125–1144
19. Heys, J., Lee, E., Manteuffel, T., McCormick, S.: An Alternative Least-Squares Formulation of the Navier-Stokes Equations with Improved Mass Conservation. J. Comp. Phys **226**(1) (September 2007) 994–1006
20. Codd, A.: Elasticity-Fluid Coupled Systems and Elliptic Grid Generation (EGG) Based on First-Order System Least Squares (FOSLS). PhD thesis, University of Colorado at Boulder (2001)
21. Codd, A., Manteuffel, T., McCormick, S.: Multilevel First-Order System Least Squares for Nonlinear Elliptic Partial Differential Equations. SIAM J. Numer. Anal **41** (2003) 2197–2209
22. Chacon, L., Knoll, D.A., Finn, J.M.: An Implicit, Nonlinear Reduced Resistive MHD Solver. J. of Computational Physics **178** (2002) 15–36
23. Chacon, L., Knoll, D.A., Finn, J.M.: Nonlinear Study of the Curvature-Driven Parallel Velocity Shear-Tearing Instability. Physics of Plasmas **9** (2002) 1164–1176
24. Philip, B., Chacon, L., Pernice, M.: Implicit Adaptive Mesh Refinement for 2D Reduced Resistive Magnetohydrodynamics. J. Comp. Phys **227**(20) (October 2008) 8855–8874
25. Strauss, H.: Nonlinear, Three-Dimensional Magnetohydrodynamics of Noncircular Tokamaks. Physics of Fluids **19** (1976) 134–140
26. Bateman, G.: MHD Instabilities. The MIT Press (1978)
27. Knoll, D.A., Chacon, L.: Coalescence of Magnetic Islands, Sloshing, and the Pressure Problem. Physics of Plasmas **13**(1) (2006)
28. Brezina, M., Falgout, R., MacLachlan, S., Manteuffel, T., McCormick, S., Ruge, J.: Adaptive smoothed aggregation (αsa) multigrid. SIAM Review (SIGEST) **47** (2005) 317–346
29. Brezina, M., Falgout, R., Maclachlan, S., Manteuffel, T., McCormick, S., Ruge, J.: Adaptive algebraic multigrid. SIAM J. on Sci. Comp. (SISC) **27** (2006) 1261–1286
30. MacLachlan, S.: Improving Robustness in Multiscale Methods. PhD thesis, University of Colorado at Boulder (2004)
31. Ruge, J.: Fospack users manual, version 1.0. unpublished (2000)