

## FIRST-ORDER SYSTEM LEAST SQUARES FOR INCOMPRESSIBLE RESISTIVE MAGNETOHYDRODYNAMICS\*

J. H. ADLER<sup>†</sup>, T. A. MANTEUFFEL<sup>†</sup>, S. F. MCCORMICK<sup>†</sup>, AND J. W. RUGE<sup>†</sup>

**Abstract.** Magnetohydrodynamics (MHD) is a fluid theory that describes plasma physics by treating the plasma as a fluid of charged particles. Hence, the equations that describe the plasma form a nonlinear system that couples Navier–Stokes equations with Maxwell’s equations. This paper shows that the first-order system least squares (FOSLS) finite element method is a viable discretization for these large MHD systems. To solve this system, a nested-iteration–Newton–FOSLS–AMG approach is taken. Most of the work is done on the coarse grid, including most of the linearizations. We show that at most one Newton step and a few V-cycles are all that are needed on the finest grid. Here, we describe how the FOSLS method can be applied to incompressible resistive MHD and how it can be used to solve these MHD problems efficiently. A 3D steady state and a reduced 2D time-dependent test problem are studied. The latter equations can simulate a “large aspect-ratio” tokamak. The goal is to resolve as much physics from the test problems with the least amount of computational work. We show that this is achieved in a few dozen work units or fine grid residual evaluations.

**Key words.** magnetohydrodynamics, algebraic multigrid, FOSLS, nested iteration

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

**DOI.** 10.1137/080727282

**1. Introduction.** In many physical applications, complex systems of partial differential equations (PDEs) must be solved. These equations can be time-dependent and nonlinear, and they can involve several dependent variables in many dimensions. In some of these applications, analytic solutions are not available, and robust numerical simulations are needed to solve the problems that arise. This paper presents one such application, magnetohydrodynamics (MHD), which consists of a set of nonlinear time-dependent PDEs on large domains. Specifically, in simulations of fusion, physicists study the effects of plasma instabilities in large fusion reactors such as tokamaks [21]. This paper develops first-order system least squares (FOSLS) [7, 8] on the MHD equations. The overall goal is to develop a numerical scheme that approximates the solution of the MHD equations accurately, resolving the desired properties, but with the least amount of computational cost. The equations are first linearized and then discretized by the FOSLS finite element method. This approach has been shown to work well on similar types of PDEs [1, 2, 7, 8, 11, 12, 14, 18, 23].

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) [4, 5, 16, 20], which, when applied to the FOSLS discretization, has been shown to be an optimal ( $O(n)$ ) solver [5, 7, 8]. This paper demonstrates that the Newton–FOSLS–AMG solver in the context of nested itera-

---

\*Received by the editors June 14, 2008; accepted for publication (in revised form) April 23, 2009; published electronically February 5, 2010. This work was supported by the Department of Energy under grants DE-FG02-03ER25574 and DE-FC02-06ER25784, Lawrence Livermore National Laboratory under contract B568677, and the National Science Foundation under grants DMS-0621199, DMS-0749317, and DMS-0811275.

<http://www.siam.org/journals/sisc/32-1/72728.html>

<sup>†</sup>Department of Applied Mathematics, Campus Box 526, University of Colorado at Boulder, Boulder, CO 80309-0526 (adlerjh@colorado.edu, tmanteuf@colorado.edu, stevem@colorado.edu, jruge@colorado.edu).

tion yields solutions to the MHD equations in a small number of work units (WUs) that is bounded independently of grid size, while using a limited amount of computational resources. Here, we define a WU to be one computation of the residual on the finest grid (i.e., one matrix-vector multiplication on the finest grid). The FOSLS methodology is what makes this possible.

In section 2 of this paper, we describe the incompressible resistive MHD equations and introduce the FOSLS methodology. Section 3 describes how we reformulate the MHD equations to make them more amenable to the FOSLS formulation. Here, we describe how we organize the FOSLS formulation to make the system work well with our AMG iterative solver. Section 4 briefly describes the overall solution algorithm, including the nested-iteration strategy and treatment of the nonlinearities. We emphasize how the nested-iteration approach, with the aid of the FOSLS error estimates, is crucial to solving the equations efficiently. Finally, in sections 5 and 6, we describe two test problems that confirm that the FOSLS discretization resolves solutions to the MHD equations in a small number of WUs, independently of the discretization parameters.

**2. Background.** The study of plasma physics involves the movement of charged particles. Plasmas are treated by a hierarchy of models [15]. Ideally, each particle would be tracked and all the forces acting on it would be computed, which, of course, would involve an enormous amount of computation and is, therefore, impractical. Instead, in kinetic theory, an ensemble average of the particles is taken and the plasma is treated as a distribution of particles of different types. This model, known as the Vlasov equation, is an exact description of a plasma if all types of particles and collisions are included. However, it is a 7D hyperbolic conservative equation and can be difficult to solve efficiently. By taking velocity moments of these equations, an infinite hierarchy of equations is obtained in just four dimensions (space and time). A truncation, or “closure,” of this hierarchy is used to get a two-fluid theory for the plasma. This is, in effect, averaging the quantities of motion, and the system is treated as a fluid of positive ions and negative electrons. When these fluids are averaged together, we arrive at the MHD equations. These equations treat the plasma as one fluid of charged particles. Hence, the equations that describe the plasma are Navier–Stokes coupled with Maxwell’s equations. A form of MHD, known as extended MHD, introduces new terms that allow the fluid to be treated, once again, with a two-fluid theory. We do not address extended MHD in this paper.

**2.1. Incompressible, resistive MHD.** As stated above, the system is a coupling of the incompressible Navier–Stokes equations 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,  $\mathbf{E}$ .

With  $\rho$  the mass density,  $-q$  the charge of an electron, and  $\nu$  the viscosity, the Navier–Stokes part of the coupled equations is

$$(2.1) \quad \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = \frac{\mathbf{j} \times \mathbf{B}}{\rho} + \frac{q\mathbf{E}}{\rho} - \frac{\nabla p}{\rho} + \nu \nabla^2 \mathbf{u},$$

$$(2.2) \quad \nabla \cdot \mathbf{u} = 0.$$

With  $\mu_0$  the permeability of free space,  $\rho_c$  the charge density, and  $c$  the speed of light, the Maxwell’s part is

$$(2.3) \quad \frac{\partial \mathbf{B}}{\partial t} = -\nabla \times \mathbf{E},$$

$$(2.4) \quad \nabla \times \mathbf{B} = \mu_0 \mathbf{j} + \frac{1}{c^2} \frac{\partial \mathbf{E}}{\partial t},$$

$$(2.5) \quad \nabla \cdot \mathbf{B} = 0,$$

$$(2.6) \quad \nabla \cdot \mathbf{E} = \rho_c.$$

In most practical applications of MHD, only low-frequency behavior is studied, and, thus, the  $\frac{1}{c^2} \frac{\partial \mathbf{E}}{\partial t}$  term can be ignored in (2.1), as well as the displacement current,  $\frac{1}{c^2} \frac{\partial \mathbf{E}}{\partial t}$ , in (2.4) [15, 22]. In addition, a resistive form of Ohm’s law is used to eliminate the electric field,  $\mathbf{E}$ , from the equations,

$$(2.7) \quad \mathbf{j} = \sigma(\mathbf{E} + \mathbf{u} \times \mathbf{B}).$$

After a nondimensionalization, the following equations for incompressible resistive MHD are obtained (i.e., Navier–Stokes equations coupled with Maxwell’s equations) [15, 22]:

$$(2.8) \quad \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} - \mathbf{j} \times \mathbf{B} + \nabla p - \frac{1}{R_e} \nabla^2 \mathbf{u} = \mathbf{f},$$

$$(2.9) \quad \frac{\partial \mathbf{B}}{\partial t} - \mathbf{B} \cdot \nabla \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{B} + \frac{1}{S_L} (\nabla \times \mathbf{j}) = \mathbf{g},$$

$$(2.10) \quad \nabla \times \mathbf{B} = \mathbf{j},$$

$$(2.11) \quad \nabla \cdot \mathbf{B} = 0,$$

$$(2.12) \quad \nabla \cdot \mathbf{u} = 0,$$

$$(2.13) \quad \nabla \cdot \mathbf{j} = 0.$$

Here,  $R_e$  is the fluid Reynolds number and  $S_L$  is the Lundquist number. Both of these are assumed to be constants and adjusted for different types of physical behavior.

**2.2. FOSLS.** To solve the nonlinear system of equations, which for discussion is denoted by  $\mathcal{L}(u) = f$ , a FOSLS discretization is used [7, 8]. First, consider a linear first-order system, denoted by  $Lu = f$ . Then the linear problem is recast as the minimization of a functional constructed by taking the  $L^2$  norm of the residual of each equation. This is written as

$$(2.14) \quad u_* = \arg \min_{u \in \mathcal{V}} G(u; f) := \arg \min_{u \in \mathcal{V}} \|Lu - f\|_0^2,$$

where  $u_*$  is the solution in an appropriate Hilbert space  $\mathcal{V}$ .

Thus,  $u_*$  satisfies  $G'(u_*)[v] = 0$ , which is the Fréchet derivative of  $G$  in the direction  $v \in \mathcal{V}$ . This results in the following weak form of the problem: Find  $u_* \in \mathcal{V}$  such that

$$(2.15) \quad \langle Lu_*, Lv \rangle = \langle f, Lv \rangle \quad \forall v \in \mathcal{V},$$

where  $\langle \cdot, \cdot \rangle$  is the usual  $L^2$  inner product. Desirable properties of the bilinear form  $\langle Lu, Lv \rangle$  are

$$(2.16) \quad \text{continuity} \quad \langle Lu, Lv \rangle \leq c_2 \|u\|_{\mathcal{V}} \|v\|_{\mathcal{V}} \quad \forall u, v \in \mathcal{V},$$

$$(2.17) \quad \text{coercivity} \quad \langle Lu, Lu \rangle \geq c_1 \|u\|_{\mathcal{V}}^2 \quad \forall u \in \mathcal{V}.$$

These properties imply the existence of a unique solution,  $u_* \in \mathcal{V}$ , for the weak problem (2.15).

Next, we approximate  $u_*$  by restricting (2.14) to a finite dimensional space,  $\mathcal{V}^h \subseteq \mathcal{V}$ , which leads to (2.15) restricted to  $\mathcal{V}^h$ . Since  $\mathcal{V}^h$  is a subspace of  $\mathcal{V}$ , the discrete problem is also well-posed. Choosing an appropriate basis,  $\mathcal{V}^h = \text{span}\{\Phi_j\}$ , yields an algebraic system of equations involving the matrix  $A$  with elements

$$(A)_{ij} = \langle L\Phi_j, L\Phi_i \rangle.$$

In general, if  $\mathcal{V}$  is a subspace of  $H^1$  (product space), then the linear system is amenable to an iterative solution by multilevel techniques. In particular, AMG has been shown to work well on a wide range of such problems [1, 2, 3, 4, 5, 7, 8, 11, 16, 18, 20, 23].

In addition, note that the functional yields an a posteriori error measure. If  $u^h \in \mathcal{V}^h$ , then

$$(2.18) \quad G(u^h; f) = \|Lu^h - f\|_0^2 = \|Lu^h - Lu_*\|_0^2 = \|Le^h\|_0^2 \approx c\|e^h\|_{\mathcal{V}}^2.$$

Thus, the functional value is equivalent to the error measured in the Hilbert space norm. In general, the constant  $c$  in (2.18) will depend on the continuity and coercivity constants,  $c_2$  and  $c_1$ . These constants will, of course, depend on properties of the PDE as well as on the boundary conditions and the computational domain. For this paper, we study simple domains and boundary conditions which will not make these constants large. In addition, the numerical results show that the problem parameters are not affecting our accuracy either. Addressing these issues analytically would be a topic of another paper and has been studied in other problems such as in [1, 2, 3, 14]. Locally, on any subset of  $\Omega$ , for example, an element, the functional yields an estimate of the error. 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. Since this is the norm we are minimizing, this allows judgments to be made based on estimates of the increase of accuracy that results from an increase in computational cost. More is said about this in section 4.

**2.3. Linearization (Newton–FOSLS).** Since the MHD system is nonlinear, we first linearize it, and then put it into a FOSLS weak form. The system becomes

$$(2.19) \quad \mathcal{L}(u_0 + \hat{u}) = \mathcal{L}(u_0) + \mathcal{L}'(u_0)[\hat{u}],$$

where  $u_0$  is the current approximation and  $\mathcal{L}'(u_0)[\hat{u}]$  is the Fréchet derivative in the direction  $\hat{u}$ . The functional then becomes

$$(2.20) \quad G(\hat{u}, f) = \|\mathcal{L}'(u_0)[\hat{u}] - (f - \mathcal{L}(u_0))\|_0^2.$$

Minimization of the linearized functional yields  $\hat{u}_*$  that satisfies the following weak form: Find  $\hat{u}_* \in \mathcal{V}$  such that

$$(2.21) \quad \langle \mathcal{L}'(u_0)[\hat{u}_*], \mathcal{L}'(u_0)[\hat{v}] \rangle = \langle (f - \mathcal{L}(u_0)), \mathcal{L}'(u_0)[\hat{v}] \rangle \quad \forall \hat{v} \in \mathcal{V}.$$

Once  $\hat{u}$  is found, it is added to the previous iterate to get the next guess,

$$(2.22) \quad u_1 = u_0 + \hat{u}.$$

Thus, in a nonlinear setting, the FOSLS approach can be applied, and, if the linearized functional retains continuity and coercivity, (2.16) and (2.17), it also retains

the desirable properties as described for linear systems. In addition, the nonlinear functional, that is, the  $L^2$  norm of the residual of the nonlinear system, can be computed, as well as the linear functional or the  $L^2$  norm of the linearized system. This allows the two functional values to be compared after each linearization and helps determine if the Newton iterations are converging as expected.

A similar, but somewhat more complicated, approach would be to create the FOSLS functional of the nonlinear problem and linearize this functional instead. This FOSLS–Newton approach generally involves more terms than the Newton–FOSLS method described here. As the approximation approaches the solution, these higher order terms go to zero, and the two methods tend to be the same. FOSLS–Newton may be more robust in some applications, but the Newton–FOSLS approach is simpler and has been successful in a number of applications [11, 12]. We confine our presentation to it in this paper.

In the next section, we show how the MHD equations can be rewritten to make them amenable to solution by the Newton–FOSLS approach.

**3. A current-vorticity formulation for reduced resistive MHD.** The first task is to write the MHD equations as a first-order system, which we do based on a vorticity-velocity-pressure formulation for the Navier–Stokes part of the equations [14]. This formulation has fewer dependent variables than the grad-velocity-pressure formulation described in [1, 2] and is chosen because many applications of fusion reactors provide explicit vorticity boundary conditions. These boundary conditions make the vorticity form of the Navier–Stokes part of the functional  $H^1$  elliptic along regular branches of the solution. From prior work on Maxwell’s equations, it is also known that a current–magnetic field formulation is  $H^1$  elliptic [3]. Thus, we believe that this combined current-vorticity formulation is a good choice for the full MHD system.

Vorticity,  $\boldsymbol{\omega} = \nabla \times \mathbf{u}$ , is introduced, and, due to incompressibility ( $\nabla \cdot \mathbf{u} = 0$ ), it is noted that

$$\nabla^2 \mathbf{u} = -\nabla \times \nabla \times \mathbf{u} = -\nabla \times \boldsymbol{\omega}.$$

Also,

$$\mathbf{u} \cdot \nabla \mathbf{u} = \frac{1}{2} \nabla |\mathbf{u}|^2 - \mathbf{u} \times (\nabla \times \mathbf{u}) = \frac{1}{2} \nabla |\mathbf{u}|^2 - \mathbf{u} \times \boldsymbol{\omega}.$$

If a new pressure variable is defined so that

$$\tilde{p} = p + \frac{1}{2} |\mathbf{u}|^2,$$

then the equations in three dimensions can be rewritten as

$$(3.1) \quad \frac{\partial \mathbf{u}}{\partial t} - \mathbf{u} \times \boldsymbol{\omega} - (\mathbf{j} \times \mathbf{B}) + \nabla(\tilde{p}) + \frac{1}{R_e} \nabla \times \boldsymbol{\omega} = \mathbf{f},$$

$$(3.2) \quad \frac{\partial \mathbf{B}}{\partial t} - (\mathbf{B} \cdot \nabla \mathbf{u}) + (\mathbf{u} \cdot \nabla \mathbf{B}) + \frac{1}{S_L} (\nabla \times \mathbf{j}) = \mathbf{g},$$

$$(3.3) \quad \nabla \times \mathbf{B} - \mathbf{j} = \mathbf{0},$$

$$(3.4) \quad \boldsymbol{\omega} - \nabla \times \mathbf{u} = \mathbf{0},$$

$$(3.5) \quad \nabla \cdot \boldsymbol{\omega} = 0,$$

$$(3.6) \quad \nabla \cdot \mathbf{B} = 0,$$

$$(3.7) \quad \nabla \cdot \mathbf{u} = 0,$$

$$(3.8) \quad \nabla \cdot \mathbf{j} = 0.$$

**3.1. Linearized equations.** Given an approximate solution,  $u^n$ , we write the true solution as  $u^n + \hat{u}$ , plug this into the system of equations, and discard any quadratic terms in  $\hat{u}$ . This yields the linearized system

$$(3.9) \quad \frac{\partial \hat{\mathbf{u}}}{\partial t} - \hat{\mathbf{u}} \times \boldsymbol{\omega}^n - \mathbf{u}^n \times \hat{\boldsymbol{\omega}} - (\mathbf{j}^n \times \hat{\mathbf{B}}) - (\hat{\mathbf{j}} \times \mathbf{B}^n) + \nabla(\hat{p}) + \frac{1}{R_e} \nabla \times \hat{\boldsymbol{\omega}} = \mathbf{f}^n,$$

$$(3.10) \quad \frac{\partial \hat{\mathbf{B}}}{\partial t} - (\mathbf{B}^n \cdot \nabla \hat{\mathbf{u}}) + (\hat{\mathbf{u}} \cdot \nabla \mathbf{B}^n) - (\hat{\mathbf{B}} \cdot \nabla \mathbf{u}^n) + (\mathbf{u}^n \cdot \nabla \hat{\mathbf{B}}) + \frac{1}{S_L} (\nabla \times \hat{\mathbf{j}}) = \mathbf{g}^n,$$

$$(3.11) \quad \nabla \times \hat{\mathbf{B}} - \hat{\mathbf{j}} = -\nabla \times \mathbf{B}^n + \mathbf{j}^n,$$

$$(3.12) \quad \hat{\boldsymbol{\omega}} - \nabla \times \hat{\mathbf{u}} = -\boldsymbol{\omega}^n + \nabla \times \mathbf{u}^n,$$

$$(3.13) \quad \nabla \cdot \hat{\boldsymbol{\omega}} = -\nabla \cdot \boldsymbol{\omega}^n,$$

$$(3.14) \quad \nabla \cdot \hat{\mathbf{B}} = -\nabla \cdot \mathbf{B}^n,$$

$$(3.15) \quad \nabla \cdot \hat{\mathbf{u}} = -\nabla \cdot \mathbf{u}^n,$$

$$(3.16) \quad \nabla \cdot \hat{\mathbf{j}} = -\nabla \cdot \mathbf{j}^n,$$

where  $\mathbf{f}^n$  and  $\mathbf{g}^n$  have been modified appropriately.

The first-order operator is then written as

$$(3.17) \quad L = \begin{pmatrix} \nabla \times & -\nabla & -I & 0 & 0 & 0 & 0 & 0 \\ \nabla \cdot & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{dt} I + \boldsymbol{\omega}^n \times & 0 & \frac{1}{R_e} \nabla \times -\mathbf{u}^n \times \nabla & -\mathbf{j}^n \times & 0 & \mathbf{B}^n \times & 0 & 0 \\ 0 & 0 & \nabla \cdot & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \nabla \times & -\nabla & -I & 0 \\ 0 & 0 & 0 & 0 & \nabla \cdot & 0 & 0 & 0 \\ -\mathbf{B}^n \cdot \nabla + \nabla \mathbf{B}^n \cdot & 0 & 0 & \frac{1}{dt} I - \nabla \mathbf{u}^n \cdot + \mathbf{u}^n \cdot \nabla & 0 & \frac{1}{S_L} \nabla \times -\nabla & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \nabla \cdot & 0 & 0 \end{pmatrix} \begin{pmatrix} \hat{\mathbf{u}} \\ \alpha \\ \hat{\boldsymbol{\omega}} \\ \hat{p} \\ \hat{\mathbf{B}} \\ \beta \\ \hat{\mathbf{j}} \\ \gamma \end{pmatrix}.$$

Slack variables  $\alpha$ ,  $\beta$ , and  $\gamma$  are introduced here to make the system square and are chosen with appropriate boundary conditions, which, along with the form of the right-hand side, ensure that they are equal to zero. These are just dummy variables used for analysis and are not used during computations. Notice that the system is composed of a sequence of Curl-Grad-Div systems along the main diagonal. Most of the off-diagonal-block terms either are zero or do not involve derivatives. If all of the off-diagonal-block terms were zeroth-order, and if we could show that the infinity norm of each previous iterate (i.e.,  $\|\mathbf{u}^n\|_\infty$ ) is bounded, then, because of the two boundary conditions in each block, it would be straightforward to establish  $H^1$  equivalence of the FOSLS functional. However, the off-diagonal-block derivatives that appear in the last block-row complicate the analysis, and  $H^1$  equivalence remains an open question.

**3.2. Scalings.** In this section, we describe a scaling of the system (3.17) to make it more amenable to solution by AMG. The scaling we describe does not change the finite element approximation, but rather the character of the algebraic system of equations. We can rescale either the specific variables or each equation. The former is a right scaling of the system, whereas the latter is a left scaling. We choose to do the scaling with a constant across the domain. Therefore, scaling on the right does not change the choice of function from the finite element space. However, scaling on the left essentially changes the norm in which we are minimizing, which does change the choice of function from the finite element space. Our goal is to find the scaling that gives the best AMG convergence. We show below that this turns out to be a right scaling, and thus, we need not worry about the effects of a left scaling.

Since the MHD system is a coupled system of Navier–Stokes and Maxwell’s equations, it is broken up into the two pieces and scalings are analyzed for each. One should note that the nonlinear cross-coupling terms,  $\mathbf{j} \times \mathbf{B}$  and  $\mathbf{u} \cdot \nabla \mathbf{B} - \mathbf{B} \cdot \nabla \mathbf{u}$ , are the terms that describe the stiffest MHD waves, and are therefore important to the physics. These two terms are complementary forces of the magnetic field interacting with the fluid. Therefore, we must always consider the contribution of these two terms in any analysis of the system. For this section, we will be testing the convergence of the system to see how well AMG performs. We would like to compare scalings of each block of the system with and without these terms added. In the end, these terms will be considered, but the goal of this section is to determine what kind of effect they will have, specifically on the linear solvers.

For now, consider a 2D system, since this is easier to analyze and no slack variables are needed. See section 6 for a derivation of the 2D equations and note that, in this context,  $\omega$  and  $j$  become scalar variables. The hat notation is now dropped.

Maxwell’s equations become

$$\begin{pmatrix} -dy & dx & -1 \\ dx & dy & 0 \\ \frac{1}{dt} & 0 & \frac{1}{S_L} dy \\ 0 & \frac{1}{dt} & -\frac{1}{S_L} dx \end{pmatrix} \begin{pmatrix} B_1 \\ B_2 \\ j \end{pmatrix}.$$

Navier–Stokes equations become

$$\begin{pmatrix} -dy & dx & -1 & 0 \\ dx & dy & 0 & 0 \\ \frac{1}{dt} & -\omega^n & \frac{1}{Re} dy - u_2^n & dx \\ \omega^n & \frac{1}{dt} & -\frac{1}{Re} dx + u_1^n & dy \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ \omega \\ p \end{pmatrix}.$$

The Maxwell system has the added coupling,  $\mathbf{u} \cdot \nabla \mathbf{B} - \mathbf{B} \cdot \nabla \mathbf{u}$ , in the last two equations, and the Navier–Stokes system has the added coupling,  $-\mathbf{j} \times \mathbf{B}$ , in the last two equations. In the FOSLS setting, we are in effect solving the  $L^*L$  system. This is called the “formal normal” and, ideally, this is a simple block-diagonal system, such as a set of uncoupled Poisson equations. The goal is to find a scaling that achieves the best AMG convergence.



To study the components, the discrete matrix,  $A$ , is created from the Newton-FOSLS discretization for each part of the system separately. The linearization is done around a known equilibrium state, and all right-hand sides are zero. One backward-Euler timestep is taken, and the convergence of AMG on this mock system is observed. The system is solved on a biquadratic finite element space on a tensor product mesh. The iterations are stopped when the linear system residual is reduced by  $10^{-6}$ . As an effective convergence factor,  $\left(\frac{F_{final}}{F_{initial}}\right)^{\frac{1}{V-cycles}}$  is computed, where  $F_{final}$  is the final and  $F_{initial}$  is the initial FOSLS functional value. This allows us to individually study the Maxwell, Navier-Stokes, and hybrid systems that introduce pieces of the couplings. For all the following tests,  $R_e = S_L = 1000$ , and the timestep  $dt = 1$ .

**3.2.1. Maxwell’s.** First, for the Maxwell system, consider the following scalings:

$$(Scaling\ 1) \quad L = \begin{pmatrix} -\frac{1}{\sqrt{S_L}}dy & \frac{1}{\sqrt{S_L}}dx & -\sqrt{S_L} \\ \frac{1}{\sqrt{S_L}}dx & \frac{1}{\sqrt{S_L}}dy & 0 \\ \frac{1}{\sqrt{S_L}} & 0 & \frac{1}{\sqrt{S_L}}dy \\ 0 & \frac{1}{\sqrt{S_L}} & -\frac{1}{\sqrt{S_L}}dx \end{pmatrix}.$$

This leads to

$$L^*L = \begin{pmatrix} \frac{1}{S_L} - \frac{1}{S_L}\Delta & 0 & \frac{1}{S_L}dy - dy \\ 0 & \frac{1}{S_L} - \frac{1}{S_L}\Delta & dx - \frac{1}{S_L}dx \\ dy - \frac{1}{S_L}dy & \frac{1}{S_L}dx - dx & S_L - \frac{1}{S_L}\Delta \end{pmatrix}.$$

This system, however, can easily be made diagonal:

$$(Scaling\ 2) \quad L = \begin{pmatrix} -\frac{1}{\sqrt{S_L}}dy & \frac{1}{\sqrt{S_L}}dx & -\sqrt{S_L} \\ \frac{1}{\sqrt{S_L}}dx & \frac{1}{\sqrt{S_L}}dy & 0 \\ 1 & 0 & dy \\ 0 & 1 & -dx \end{pmatrix},$$

$$\Rightarrow L^*L = \begin{pmatrix} 1 - \frac{1}{S_L}\Delta & 0 & 0 \\ 0 & 1 - \frac{1}{S_L}\Delta & 0 \\ 0 & 0 & S_L - \Delta \end{pmatrix}.$$

We see in Table 3.1 that the latter scaling produces good results, as expected, whereas the first one does not do as well.

TABLE 3.1  
Maxwell system with no nonlinear coupling.

Equations	V-cycles used	Average residual reduction factor	Effective convergence factor
Scaling 1	60	0.84	0.826
Scaling 2	5	0.04	0.127

Now, if the nonlinear cross-coupling term,  $\mathbf{u} \cdot \nabla \mathbf{B} - \mathbf{B} \cdot \nabla \mathbf{u}$ , is introduced, we can see how this affects AMG on the Maxwell system. A fixed equilibrium state is used for the values of  $\mathbf{u}^n$ . All scalings lead to a complicated formal-normal, which is no longer diagonal. See Table 3.2.



TABLE 3.2  
*Maxwell system with nonlinear coupling added.*

Equations	V-cycles used	Average residual reduction factor	Effective convergence factor
Scaling 1	72	0.86	0.895
Scaling 2	17	0.5	0.694

While the AMG convergence deteriorated slightly for Scaling 2, AMG still performs better than for Scaling 1. However, it is interesting to note that the introduction of the coupling term does not affect the performance of Scaling 1 very much. Since this latter system is the physical one that we would like to solve, this indicates that Scaling 1 might be the appropriate choice.

**3.2.2. Navier–Stokes.** Next, consider Navier–Stokes, where

$$(Scaling\ 1) \quad L = \begin{pmatrix} -\frac{1}{\sqrt{R_e}}dy & \frac{1}{\sqrt{R_e}}dx & -\sqrt{R_e} & 0 \\ \frac{1}{\sqrt{R_e}}dx & \frac{1}{\sqrt{R_e}}dy & 0 & 0 \\ \frac{1}{\sqrt{R_e}} & -\omega^n & \frac{1}{\sqrt{R_e}}dy - u_2^n & -\sqrt{R_e}dx \\ \omega^n & \frac{1}{\sqrt{R_e}} & -\frac{1}{\sqrt{R_e}}dx + u_1^n & -\sqrt{R_e}dy \end{pmatrix}.$$

For simplicity, consider  $\mathbf{u}^n = \omega^n = 0$ , so that

$$L^*L = \begin{pmatrix} \frac{1}{R_e}(1 - \Delta) & 0 & \frac{1}{R_e}dy - dy & -dx \\ 0 & \frac{1}{R_e}(1 - \Delta) & dx - \frac{1}{R_e}dx & -dy \\ dy - \frac{1}{R_e}dy & \frac{1}{R_e}dx - dx & R_e - \frac{1}{R_e}\Delta & 0 \\ dx & dy & 0 & -R_e\Delta \end{pmatrix}.$$

A different scaling leads to a simpler formal-normal:

$$(Scaling\ 2) \quad L = \begin{pmatrix} -\frac{1}{R_e^{1/4}}dy & \frac{1}{R_e^{1/4}}dx & -R_e^{1/4} & 0 \\ \frac{1}{R_e^{1/4}}dx & \frac{1}{R_e^{1/4}}dy & 0 & 0 \\ R_e^{1/4} & -\sqrt{R_e}\omega^n & \frac{1}{R_e^{1/4}}dy - \frac{1}{\sqrt{R_e}}u_2^n & -\frac{1}{R_e^{1/4}}dx \\ \sqrt{R_e}\omega^n & R_e^{1/4} & -\frac{1}{R_e^{1/4}}dx + \frac{1}{\sqrt{R_e}}u_1^n & -\frac{1}{R_e^{1/4}}dy \end{pmatrix}.$$

Again, with  $\mathbf{u}^n = \omega^n = 0$ , we have

$$L^*L = \begin{pmatrix} \sqrt{R_e} - \frac{1}{R_e}\Delta & 0 & 0 & -dx \\ 0 & \sqrt{R_e} - \frac{1}{R_e}\Delta & 0 & -dy \\ 0 & 0 & \sqrt{R_e} - \frac{1}{R_e}\Delta & 0 \\ dx & dy & 0 & -\frac{1}{\sqrt{R_e}}\Delta \end{pmatrix}.$$

First, with  $\mathbf{u}^n$  and  $\omega^n$  not equal to zero, we see the results in Table 3.3.

Then, for the results in Table 3.4, we have the  $\mathbf{u} \times \omega$  term removed completely.

It appears that Scaling 1 is doing slightly better here, though both methods are fairly comparable. Again, it is interesting to note that the introduction of the nonlinear term does not affect the performance of Scaling 1. This leads us to believe that this scaling should handle the coupled system better.

TABLE 3.3  
*Navier–Stokes system with nonlinear coupling added.*

Equations	V-cycles used	Average residual reduction factor	Effective convergence factor
Scaling 1	34	0.74	0.71
Scaling 2	40	0.77	0.83

TABLE 3.4  
*Navier–Stokes system with no nonlinear coupling.*

Equations	V-cycles used	Average residual reduction factor	Effective convergence factor
Scaling 1	34	0.74	0.71
Scaling 2	41	0.78	0.81

**3.2.3. Coupled system.** Based on the above results, it would appear that combining Scaling 2 for Maxwell’s and Scaling 1 for Navier–Stokes should yield the best convergence for the whole system with no coupling. We show all possible combinations for the full MHD system.

First, with the  $\mathbf{u} \cdot \nabla \mathbf{B} - \mathbf{B} \cdot \nabla \mathbf{u}$  term removed from the full system, we obtain the results in Table 3.5.

TABLE 3.5  
*Maxwell and Navier–Stokes system combined with no coupling.*

Max scaling	N/S scaling	V-cycles	Average residual reduction factor	Effective convergence factor
1	1	75	0.89	0.883
1	2	75	0.90	0.914
2	1	35	0.74	0.726
2	2	75	0.89	0.890

As expected, Scaling 2 for Maxwell’s and Scaling 1 for Navier–Stokes work the best. However, with the  $\mathbf{u} \cdot \nabla \mathbf{B} - \mathbf{B} \cdot \nabla \mathbf{u}$  term added back in we notice, in Table 3.6, that Scaling 1 for both blocks works best.

TABLE 3.6  
*Fully coupled system.*

Max scaling	N/S scaling	V-cycles	Average residual reduction factor	Effective convergence factor
1	1	51	0.82	0.807
1	2	75	0.90	0.911
2	1	62	0.85	0.852
2	2	75	0.90	0.916

Thus, after trying various scalings, it was determined that what worked best for the individual Maxwell and Navier–Stokes systems was not the best in the end for the fully coupled system. This scaling produces more consistent results when the nonlinear cross-coupling is added. Since these coupling terms are what drives the physics, specifically at the fastest time scales, this is the scaling we must choose. Finally, Scaling 1 was obtained only by a change of variables and not a scaling of the equations. As described above, this does not affect the finite element solution.

**3.3. Current-vorticity model.** Thus, our final formulation in three dimensions is obtained by letting  $\mathbf{u} \rightarrow \frac{1}{\sqrt{R_e}}\mathbf{u}$ ,  $\boldsymbol{\omega} \rightarrow \sqrt{R_e}\boldsymbol{\omega}$ ,  $p \rightarrow -\sqrt{R_e}p$ ,  $\mathbf{B} \rightarrow \frac{1}{\sqrt{S_L}}\mathbf{B}$ ,  $\mathbf{j} \rightarrow \sqrt{S_L}\mathbf{j}$ :

$$(3.18) \quad \frac{1}{\sqrt{R_e}}\nabla \times \mathbf{u} - \sqrt{R_e}\boldsymbol{\omega} = 0,$$

$$(3.19) \quad \frac{1}{\sqrt{R_e}}\nabla \cdot \mathbf{u} = 0,$$

$$(3.20) \quad \sqrt{R_e}\nabla \cdot \boldsymbol{\omega} = 0,$$

$$(3.21) \quad \frac{1}{\sqrt{R_e}}\frac{\partial \mathbf{u}}{\partial t} - \mathbf{u} \times \boldsymbol{\omega} - \mathbf{j} \times \mathbf{B} - \sqrt{R_e}\nabla p + \frac{1}{\sqrt{R_e}}\nabla \times \boldsymbol{\omega} = \mathbf{f},$$

$$(3.22) \quad \frac{1}{\sqrt{S_L}}\nabla \times \mathbf{B} - \sqrt{S_L}\mathbf{j} = 0,$$

$$(3.23) \quad \frac{1}{\sqrt{S_L}}\nabla \cdot \mathbf{B} = 0,$$

$$(3.24) \quad \sqrt{S_L}\nabla \cdot \mathbf{j} = 0,$$

$$(3.25) \quad \frac{1}{\sqrt{S_L}}\frac{\partial \mathbf{B}}{\partial t} + \frac{1}{\sqrt{R_e S_L}}(\mathbf{u} \cdot \nabla \mathbf{B} - \mathbf{B} \cdot \nabla \mathbf{u}) + \frac{1}{\sqrt{S_L}}\nabla \times \mathbf{j} = \mathbf{g}.$$

Here, we have 16 equations and 13 unknowns. The slack variables make up the three unknowns in system (3.17). Since they are zero, they are not included in any computations.

This current-vorticity formulation consists of two block matrices of a div/curl system coupled together with the square-root scaling of the Reynolds and Lundquist numbers distributed symmetrically across the system.

**4. Nested-iteration–Newton–FOSLS–AMG.** Now that a FOSLS formulation has been established, it can be applied to some test problems. First, we describe the strategy for solving the algebraic systems. The main component of this algorithm is a nested-iteration approach. Most of the work is done on coarse grids, where computation is much cheaper, and then moved up to the finer grids.

An initial guess is used on a very coarse grid. Using this initial guess, the system is linearized as described above. The least squares functional is then formed, and the minimization results in an algebraic system,  $Ax = b$ . Several AMG V-cycles are used to solve the algebraic system. If necessary, several more linearization steps are done and the process is repeated. After the Newton steps have converged and the algebraic system is solved “well enough,” the solution is interpolated to a finer grid where it is again linearized, FOSLSized, and solved with AMG [18]. In addition, we minimize the number of linearizations and multigrid V-cycles on each refinement level to solve the problem accurately and with the least amount of work. A strategy has been developed for this, parts of which are described in [23], but we only briefly go into it here.

**4.1. Nested-iteration efficiency.** The goal is to find out how many Newton steps and linear iterations are needed at each refinement level to solve the problem with the most accuracy-per-computational-cost. Results below show that only a few Newton steps and V-cycles are needed on coarser grids to get the solution on the

finest grid. This supports the hypothesis that while the majority of work should be done on coarser grids, it is not necessary to solve the discrete problem on these coarse grids below the level of discretization error. In fact, at some point, doing more work on the coarser grid is not as efficient as just moving to a finer grid. This is where the FOSLS methodology proves to be important. Due to the a posteriori error estimate of the functional, we can decide how well both the linear and nonlinear systems are being solved. The linear and nonlinear functional merely need to be computed. With these values, along with the AMG convergence factors, stopping parameters are computed for the linear iterations as well as for the Newton iterations. For instance, with the use of the linear functional and the AMG convergence factor, we can decide if we can improve our accuracy-per-computational-cost of the linear iterations by performing another V-cycle on the current grid, or by relinearizing the system and starting over. Using knowledge of how much the functional is reduced by performing another linear iteration on a current grid, and taking into account how much more work it would cost to do the same iteration on a finer grid, we can determine a good stopping parameter. Work here is extended from [23] in that, if the functional value of the current iterate is within a given tolerance of the functional minimum, then the linear iterations are stopped. For the nonlinear problem, the linear and nonlinear functionals are compared after each Newton step, and it is determined whether another linearization is needed or if proceeding to a finer grid would provide the most accuracy-per-computational-cost. Here, if the linear functional is within a given tolerance of the nonlinear functional of the current Newton step, no more linearizations are performed. Experiments have shown that these two tolerances should be the same order. A more in-depth explanation of this procedure and how the stopping parameters are computed using the FOSLS's functionals will be the topic of a future paper. The following flow chart, Figure 4.1, should give an idea of how the overall algorithm will work.  $F_s^2$  is the linear functional at linear iteration  $s$ , and  $\mathcal{F}_s^2$  is the nonlinear functional at Newton step  $s$ .

The goal of this paper, however, is to show that, with the FOSLS formulation and the methods described above, the complex MHD systems are solved efficiently, in a minimal number of WUs.

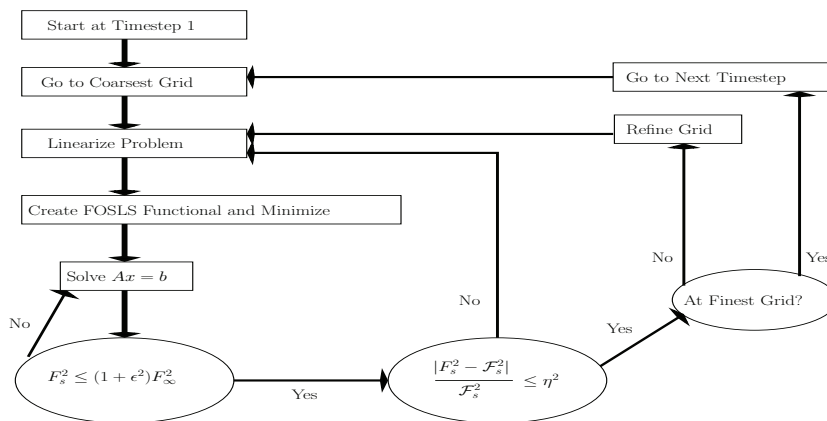


FIG. 4.1. Nested-iteration algorithm.

**5. 3D test problem.** Our first test of the Newton–FOSLS approach on incompressible resistive MHD is a steady-state test problem with manufactured solutions. The first-order system is as in (3.18)–(3.25) with the time-derivatives removed.

Let the true solution be as follows:

$$\mathbf{u}(x, y, z) = \begin{pmatrix} \sin(\pi x)(\cos(\pi y) - \cos(\pi z)) \\ \sin(\pi y)(\cos(\pi z) - \cos(\pi x)) \\ \sin(\pi z)(\cos(\pi x) - \cos(\pi y)) \end{pmatrix}, \quad \boldsymbol{\omega} = \begin{pmatrix} 2\pi \sin(\pi y) \sin(\pi z) \\ 2\pi \sin(\pi z) \sin(\pi x) \\ 2\pi \sin(\pi x) \sin(\pi y) \end{pmatrix},$$

$$\mathbf{B}(x, y, z) = \begin{pmatrix} \sin(\pi x)(\cos(\pi y) - \cos(\pi z)) \\ \sin(\pi y)(\cos(\pi z) - \cos(\pi x)) \\ \sin(\pi z)(\cos(\pi x) - \cos(\pi y)) \end{pmatrix}, \quad \mathbf{j}(x, y, z) = \begin{pmatrix} 2\pi \sin(\pi y) \sin(\pi z) \\ 2\pi \sin(\pi z) \sin(\pi x) \\ 2\pi \sin(\pi x) \sin(\pi y) \end{pmatrix},$$

$$p = 1 - z.$$

On a cube domain,  $\Omega = [0, 1]^3$ , we have the following boundary conditions for a perfect conductor.

On all faces:

$$\begin{aligned} \mathbf{n} \cdot \mathbf{u} &= 0, \\ \mathbf{n} \cdot \mathbf{B} &= 0, \\ \mathbf{n} \times (\mathbf{j} - S_L(\mathbf{u} \times \mathbf{B})) &= 0, \\ \mathbf{n} \times \boldsymbol{\omega} &= 0. \end{aligned}$$

On the inflow and outflow boundaries:

$$p = \begin{cases} 1 & \text{inflow wall } z = 0, \\ 0 & \text{outflow } z = 1. \end{cases}$$

This allows the right-hand sides of the MHD equations to be built and gives a well-posed problem, for which there is a known solution.

**5.1. Nested-iteration tests.** The problem is first solved on a grid of  $h = 1/2$ . The system is over solved, in that as many Newton steps as needed are performed for the linear functional to converge to the nonlinear functional to within five digits. At each Newton step, the linear system is solved with as few V-cycles as possible to reach discretization error. The solution is then moved up to the next level,  $h = 1/4$ , and the process is repeated. By the time the finest grid is reached, only one Newton step and a few V-cycles are needed. The following results in Table 5.1 start with a random initial guess.

As we get to finer and finer grids, one can see that the number of Newton steps goes to one with only a few V-cycles being used. We next determine how much work is actually being done.

**5.1.1. Work.** Again, a work unit (WU) is defined to be the cost of one matrix-vector multiplication at the finest level. Multigrid theory tells us that the cost of one V(1,1) cycle in three dimensions at that level is  $16/7$  WUs [5]. Since going to a coarser level ( $h$  is reduced by a half) decreases the number of unknowns by  $1/8$ , the total work at a given level is  $(\frac{1}{8^L})(\frac{16k}{7})$  WUs, where  $L = 0$  is the finest level and  $k$  is the number of V-cycles performed. Ignoring setup cost, we can crudely consider the

TABLE 5.1

*Nested iteration tests for three-dimensional problem. Shows number of Newton steps and V-cycles used on each level or refinement.*

$h$	Newton step	Linear functional	Nonlinear functional	V-cycles used
1/2	1	15.758	15.910	4
1/2	2	14.823	14.909	3
1/2	3	14.712	14.760	3
1/2	4	14.710	14.727	3
1/2	5	14.715	14.720	2
1/2	6	14.717	14.718	2
1/2	7	14.718	14.718	2
1/4	1	8.195	8.207	5
1/4	2	8.146	8.144	4
1/4	3	8.142	8.142	3
1/8	1	4.664	4.663	10
1/8	2	4.657	4.657	5
1/16	1	3.126	3.126	15

work of one Newton step with  $k$  V-cycles at that level to be  $\alpha \frac{16k}{7}$  WUs, so the total work at a given level is

$$(5.1) \quad \sum_0^m \alpha \left( \frac{1}{8^L} \right) \left( \frac{16k(m)}{7} \right),$$

where  $m$  is the number of Newton steps,  $k(m)$  is the number of V-cycles for that Newton step, and  $\alpha$  is such that a Newton step is  $\alpha$  times more work than a V-cycle.

For now, assume  $\alpha = 1$ . In reality, the Newton step is more costly and depends on the amount of nonlinearity and the FOSLS setup. Thus, we want to perform fewer of them. For this test problem, where the finest level is 1/16 and discretization error is reached on each level, we have

- 0.085 WUs for  $h = 1/2$ ,
- 0.429 WUs for  $h = 1/4$ ,
- 4.286 WUs for  $h = 1/8$ ,
- 34.286 WUs for  $h = 1/16$ .

This leads to about 39 WUs to get the results obtained. This shows that, by performing most of the linearizations on coarser grids, it is possible to get better coarse-grid estimates to the solution for finer grids and, thus, less work is needed on the finer grids to get the desired solution. The above results show that, by the time the finest grid is reached, the complex nonlinear system is solved with only one Newton linearization and a few V-cycles. Therefore, nested iteration is crucial to solving these systems efficiently. This also shows that the FOSLS discretization allowed the large 16 by 13 system to be solved correctly, thus making it a viable method for MHD. Next, we see how well this process does for a fusion application.

**6. 2D reduced MHD.** The algorithm above was next applied to a more realistic test problem [9, 10, 17, 21]. From the papers by Chacon, Knoll, and Finn [9] and Philip, Chacon, and Pernice [17], a reduced set of MHD equations is obtained. These equations simulate a “large aspect-ratio” tokamak, with noncircular cross-sections. Here, the  $B$ -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 direction.

Consider a regular torus, as in Figure 6.1. With the strong magnetic field in the toroidal direction and with  $R \gg r$ , we study the fluid motion in the poloidal direction.

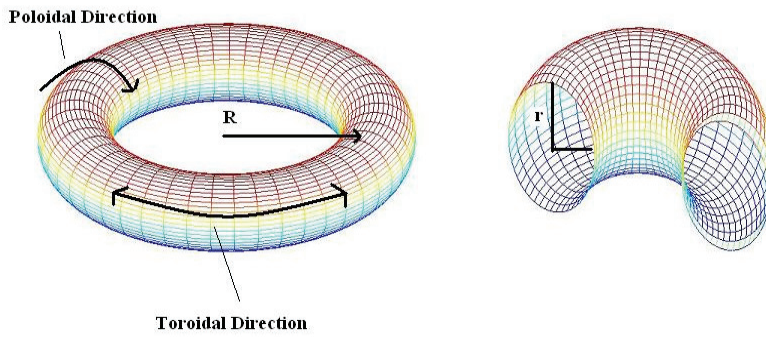


FIG. 6.1. *Torus-shaped tokamak.*

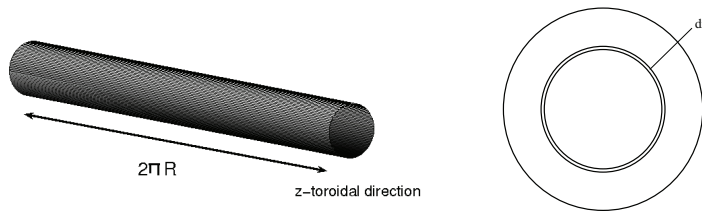


FIG. 6.2. *Cross-section in poloidal direction.*

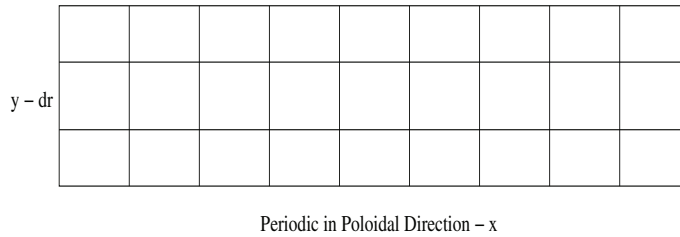


FIG. 6.3. *Cartesian grid upon which the problem is solved.*

The tokamak is also approximated as a cylinder as in Figure 6.2. Finally, we take a small slice of the poloidal cross-section and look at the instabilities in this region. The annulus is cut and is modeled on a Cartesian grid, which is periodic in  $x$ . See Figure 6.3.

In this context, assume

$$\mathbf{B} = \begin{pmatrix} B_1(x, y) \\ B_2(x, y) \\ B_3 \gg 1 \end{pmatrix} \Rightarrow \mathbf{j} = \begin{pmatrix} 0 \\ 0 \\ j_3(x, y) \end{pmatrix},$$

$$\mathbf{u} = \begin{pmatrix} u_1(x, y) \\ u_2(x, y) \\ 0 \end{pmatrix} \Rightarrow \boldsymbol{\omega} = \nabla \times \mathbf{u} = \begin{pmatrix} 0 \\ 0 \\ \omega_3(x, y) \end{pmatrix}.$$



A poloidal velocity stream function,  $\Phi$ , and a poloidal flux,  $\Psi$ , are introduced such that

$$-\nabla^\perp \Psi = \mathbf{B}, \quad -\nabla^\perp \Phi = \mathbf{u}.$$

This enforces the incompressibility of  $\mathbf{u}$  and  $\mathbf{B}$ . The reduced equations become

$$(6.1) \quad \frac{\partial \Psi}{\partial t} + \mathbf{u} \cdot \nabla \Psi - \frac{1}{\sigma \mu_0} \nabla^2 \Psi = -E_0,$$

$$(6.2) \quad \rho \frac{\partial \omega_3}{\partial t} + \rho \mathbf{u} \cdot \nabla \omega_3 - \rho \nu \nabla^2 \omega_3 - \mathbf{B} \cdot \nabla j_3 = -S_w,$$

$$(6.3) \quad \nabla^2 \Phi = \omega_3,$$

$$(6.4) \quad \nabla^2 \Psi = \mu_0 j_3.$$

Here,  $B_3$  is large and constant. The functions  $E_0$  and  $S_w$  are source functions used to balance the decay of the equilibrium solution due to transport terms. They are obtained by plugging the initial conditions into the equation and dropping the full time derivative,  $\frac{D}{Dt} = \frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla$ .

By applying  $-\nabla^\perp$  to (6.1), we arrive at

$$(6.5) \quad \frac{\partial \mathbf{B}}{\partial t} - \nabla \times (\mathbf{u} \times \mathbf{B}) + \left( \nabla \times \frac{\mathbf{j}}{\sigma} \right) = \nabla^\perp E_0,$$

which is exactly (2.9) in the original formulation plus the source term. Also note that, since  $\nabla \cdot \mathbf{u} = 0$  and  $\nabla \cdot \mathbf{B} = 0$ , we have  $\nabla \times (\mathbf{u} \times \mathbf{B}) = \mathbf{B} \cdot \nabla \mathbf{u} - \mathbf{u} \cdot \nabla \mathbf{B}$ .

By taking the curl of the Navier–Stokes part of the original formulation (equation (2.8)) and using the above assumptions, (6.2) is obtained. Thus, the reduced model is equivalent to a 2D version of the coupled Navier–Stokes and Maxwell equations derived in section 2. The 2D form of the formulation, (3.18)–(3.25) described in section 3, can then be used. We now perform some simulations on a test problem that arises from this formulation. The one considered here is known as the tearing mode instability.

**6.1. Test problem: Tearing modes.** This test problem simulates the tearing mode instability arising from perturbations in an initial current sheet of a tokamak generator. In this case, a current sheet in the toroidal direction of the tokamak is perturbed, resulting in an instability that causes a reconnection in the magnetic field lines. The result is a formation of an island in the contours of the current field and a “tearing” in the contours of the vorticity. For the following simulations,

$$\Omega = [0, 4] \times [0, 1],$$

$$R_e = S_L = 1000,$$

$$\lambda = 0.2,$$

$$\epsilon = 0.1.$$

The initial conditions at equilibrium are

$$(6.6) \quad \mathbf{B}_0(x, y) = \begin{pmatrix} -\tanh(\frac{1}{\lambda}(y - 1/2)) + \frac{4}{3\lambda}(y - 1/2)^3 \sec h^2\left(\frac{1/2}{\lambda}\right) \\ 0 \end{pmatrix},$$

$$(6.7) \quad \mathbf{u}_0(x, y) = \mathbf{0},$$

$$(6.8) \quad \omega_0(x, y) = 0,$$

$$(6.9) \quad j_{30}(x, y) = \nabla \times \mathbf{B}_0 = -\frac{4}{\lambda}(y - 1/2)^2 \sec h^2\left(\frac{1/2}{\lambda}\right) + \frac{1}{\lambda} \left( \sec h^2\left(\frac{y - 1/2}{\lambda}\right) \right),$$

$$(6.10) \quad p_0(x, y) = c - \frac{B_{10}^2}{2} \\ = c - \frac{1}{2} \left( -\tanh\left(\frac{1}{\lambda}(y - 1/2)\right) + \frac{4}{3\lambda}(y - 1/2)^3 \sec h^2\left(\frac{1/2}{\lambda}\right) \right)^2.$$

These initial conditions are perturbed away from equilibrium as follows:

$$(6.11) \quad \delta \mathbf{B}_0(x, y) = \begin{pmatrix} -\epsilon \pi \cos(\pi y) \sin(\pi x/2 - \pi/2) \\ \frac{1}{2} \epsilon \pi \sin(\pi y) \cos(\pi x/2 - \pi/2) \\ 0 \end{pmatrix},$$

$$(6.12) \quad \delta j_{30}(x, y) = -\frac{5}{4} \epsilon \pi^2 \sin(\pi y) \sin(\pi x/2 - \pi/2),$$

$$(6.13) \quad \delta p_0(x, y) = c + \frac{5}{8} \epsilon^2 \pi^2 \sin^2(\pi y) \sin^2(\pi x/2 - \pi/2).$$

The boundary conditions are periodic in  $x$  and homogeneous Dirichlet for the current and vorticity. Again, this allows for the FOSLS formulation we prescribe to be  $H^1$  elliptic for the Navier–Stokes and the Maxwell blocks. We, therefore, expect our algorithm to perform well on this tearing mode problem and to resolve the physical phenomena.

**6.2. Results.** The problem was run to time 200 with a timestep of 5 Alfvén units using a backward-Euler implicit scheme. At this point the tearing mode should be in full gear. The full nested-iteration algorithm was applied only on the first timestep. For all following timesteps, the linearizations and algebraic solves were all done on the finest grid. Future work involves incorporating the nested-iteration algorithm to all timesteps. Stopping criteria were used for the AMG V-cycles and the linearizations so that the problem was solved to within a specified fraction of the discretization error. In other words, minimal work was done to obtain the solution on each grid. This was briefly described in section 4. As the results show, this was sufficient for the tearing mode on a  $32 \times 128$  fine grid with quadratic elements. On the finest grid and for later timesteps, only one Newton step was needed for the linearization to converge, and an average of four V-cycles were needed for the linear solves to converge, as shown in Table 6.1. All tests were done on an AMD Opteron shared memory machine running Suse Linux and with 64GB of shared memory.

As one can see in Figure 6.4, the main properties of the tearing mode have been captured. Computing the total work, as was done for the 3D problem, gives about 32 WUs on the first timestep and a total of less than 450 WUs for the full problem at time 200. This is due to the fact that most of the work was done on coarser levels. Few computations were done on the fine level, which would have been more costly. For later timesteps, about 11 WUs were used at each step. The solution did not change much at later timesteps and, therefore, not much work was needed.

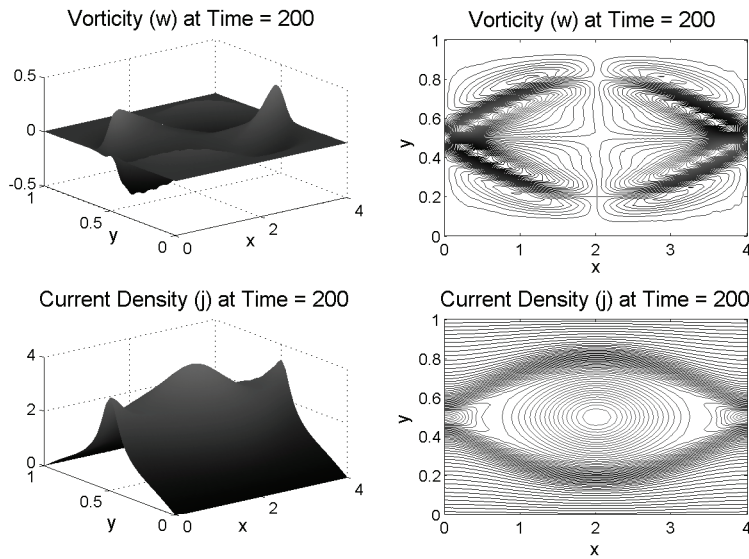


FIG. 6.4. Numerical solution after 200 timesteps.  $h = 1/32$ ,  $p = 2$ . Top: vorticity. Bottom: current density.

Most importantly, though, these results show that the FOSLS formulation is capable of capturing complex instabilities in a fusion reactor simulation. FOSLS gives a discretization that AMG can solve well and efficiently, especially in a nested-iteration algorithm. It also gives sharp error estimates that aid in continuously calculating the accuracy-per-computational-cost, helping to better solve the system.

TABLE 6.1

Tearing mode problem solved with a timestep of  $5\tau_A$ . Finest grid:  $32 \times 128$  quadratic elements  $\Rightarrow$  116,935 DOF.  $\epsilon^2 = \eta^2 = 0.1$ . Time to solution: < 14 minutes.

Timesteps	Grid	Newton steps	Average V-cycles used
1	1	2	4
1	2	3	5.67
1	3	2	7.5
1	4	1	6
1	5	2	14
1	6	1	4
40(Time = 200)	6	1	4

In addition, our algorithm is first order in time, which we would expect from a backward-Euler scheme. This is shown in Figure 6.5. The algorithm is run out to a time of  $60\tau_A$  with various timestep sizes. Each of these solutions is compared to the solution when a very small timestep,  $0.04\tau_A$ , is used. The analysis of the stability of our timestepping is complicated by the fact that our system is actually a set of differential algebraic equations. In other words, we have a set of constraint equations in addition to our time-evolution equations. A complete analysis of this will be done in a future paper, but the preliminary results here show that we are solving these constraint equations well enough to still guarantee stability and convergence of the timestepping scheme.

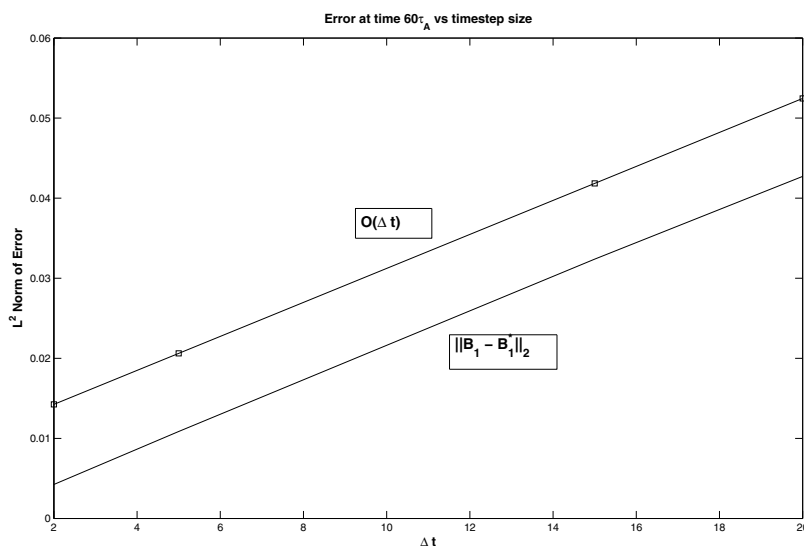


FIG. 6.5. Error at timestep  $60\tau_A$  versus timestep size. Shows  $O(\Delta t)$  convergence of backward-Euler scheme.

**7. Discussion.** We have shown that the FOSLS finite element method, along with full multigrid, is highly effective for the complex current-vorticity form of resistive MHD. Real world MHD applications can be solved very efficiently when the focus is on accuracy-per-computational-cost. The crux of the above methods uses the fact that we do most of the difficult computations, such as the linearizations, on coarser grids where computing is cheap. As a result, when the desired grid is reached, we obtain an approximation to the solution in only a few work units. The use of FOSLS greatly aided this process. Its sharp a posteriori error estimate allows parameters to be computed that can estimate the current accuracy-per-computational-cost. From this, judgments can be made on what further computation is necessary. Should the method perform another V-cycle, stop iterating, and relinearize the problem, or quit on the current grid altogether and move to a finer level? Such decision making is important for complex systems, such as the MHD equations.

Several aspects still need to be studied. First, if we use previous data to determine when to move to a finer grid or to keep iterating, we must be certain to solve the previous data “well enough” to get good estimates. For instance, if we are using data from the previous timestep to determine the parameters of the current timestep, do we need to solve the problem on the previous timestep better than predicted to correctly obtain those parameters? This issue also leads to a second matter to consider. In the FOSLS setting, the PDEs are a set of differential algebraic equations. In 2D MHD as above, there are four time-propagating equations with four constraint or algebraic equations. How does the fact that the constraint equations are not solved exactly affect the timestepping of the first four equations? Analysis of the equations with various implicit backward differencing timestepping schemes remains to be done. It is important to confirm that the timestepping is still as accurate as in standard methods.

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. FOSLS was 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.

## REFERENCES

- [1] P. BOCHEV, Z. CAI, T. A. MANTEUFFEL, AND S. F. MCCORMICK, *Analysis of velocity-flux first-order system least-squares principles for the Navier–Stokes equations: Part I*, SIAM J. Numer. Anal., 35 (1998), pp. 990–1009.
- [2] P. BOCHEV, T. A. MANTEUFFEL, AND S. F. MCCORMICK, *Analysis of velocity-flux least-squares principles for the Navier–Stokes equations: Part II*, SIAM J. Numer. Anal., 36 (1999), pp. 1125–1144.
- [3] J. H. BRAMBLE, T. V. KOLEV, AND J. E. PASCIAK, *A least-squares approximation method for the time-harmonic Maxwell equations*, J. Numer. Math., 13 (2005), pp. 237–263.
- [4] A. BRANDT, *Algebraic multigrid theory: The symmetric case*, Appl. Math. Comput., 19 (1986), pp. 23–56.
- [5] W. L. BRIGGS, V. E. HENSON, AND S. F. MCCORMICK, *A Multigrid Tutorial*, 2nd ed., SIAM, Philadelphia, 2000.
- [6] P. N. BROWN, P. S. VASSILEVSKI, AND C. S. WOODWARD, *On mesh-independent convergence of an inexact Newton–multigrid algorithm*, SIAM J. Sci. Comput., 25 (2003), pp. 570–590.
- [7] 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.
- [8] Z. CAI, T. A. MANTEUFFEL, 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.
- [9] L. CHACON, D. A. KNOLL, AND J. M. FINN, *An implicit, nonlinear reduced resistive MHD solver*, J. Comput. Phys., 178 (2002), pp. 15–36.
- [10] L. CHACON, D. A. KNOLL, AND J. M. FINN, *Nonlinear study of the curvature-driven parallel velocity shear-tearing instability*, Physics Plasmas, 9 (2002), pp. 1164–1176.
- [11] A. CODD, *Elasticity-Fluid Coupled Systems and Elliptic Grid Generation (EGG) Based on First-Order System Least Squares (FOSLS)*, Ph.D. thesis, University of Colorado at Boulder, Boulder, CO, 2001.
- [12] 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.
- [13] H. DE STERCK, T. MANTEUFFEL, S. MCCORMICK, J. NOLTING, J. RUGE, AND L. TANG, *Efficiency-based h- and hp-refinement strategies for finite element methods*, Numer. Linear Algebra Appl., 15 (2008), pp. 89–114.
- [14] J. J. HEYS, E. LEE, T. MANTEUFFEL, AND S. F. MCCORMICK, *An alternative least-squares formulation of the Navier–Stokes equations with improved mass conservation*, J. Comput. Phys., 226 (2007), pp. 994–1006.
- [15] D. R. NICHOLSON, *Introduction to Plasma Theory*, John Wiley and Sons, New York, 1983.
- [16] C. W. OOSTERLEE, A. SCHULLER, AND U. TROTTENBERG, *Multigrid*, Academic Press, New York, 2000.
- [17] B. PHILIP, L. CHACON, AND M. PERNICE, *Implicit adaptive mesh refinement for 2D reduced resistive magnetohydrodynamics*, J. Comput. Phys., 227 (2008), pp. 8855–8874.
- [18] O. ROEHRLE *Multilevel First-Order System Least Squares for Quasilinear Elliptic Partial Differential Equations*, Ph.D. thesis, University of Colorado at Boulder, Boulder, CO, 2004.
- [19] J. RUGE, *Fospack Users Manual, Version 1.0*, University of Colorado at Boulder, Boulder, CO, 2000.
- [20] J. RUGE AND K. STÜBEN, *Algebraic multigrid*, in *Multigrid Methods*, S. F. McCormick, ed., SIAM, Philadelphia, 1987, pp. 73–130.
- [21] H. R. STRAUSS *Nonlinear, three-dimensional magnetohydrodynamics of noncircular tokamaks*, Phys. Fluids, 19 (1976), pp. 134–140.
- [22] P. ULLRICH, *Dynamics and Properties of the Magnetohydrodynamics Equations*, unpublished notes, University of Waterloo, Waterloo, Ontario, 2005.
- [23] C. WESTPHAL, *First-Order System Least Squares for Geometrically-Nonlinear Elasticity in Nonsmooth Domains*, Ph.D. thesis, University of Colorado at Boulder, Boulder, CO, 2004.