INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2008; **58**:1201–1222
Published online 14 April 2008 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.1755

Implicit symmetrized streamfunction formulations of magnetohydrodynamics

K. S. Kang[1,*,†] and D. E. Keyes[1,2]

[1]*Brookhaven National Laboratory, Computational Science Center, Building 463, Room 255, Upton, NY 11973, U.S.A.*
[2]*Applied Physics & Applied Mathematics, Columbia University, New York, NY 10027, U.S.A.*

SUMMARY

We apply the finite element method to the classic tilt instability problem of two-dimensional, incompressible magnetohydrodynamics, using a streamfunction approach to enforce the divergence-free conditions on the magnetic and velocity fields. We compare two formulations of the governing equations, the standard one based on streamfunctions and a hybrid formulation with velocities and magnetic field components. We use a finite element discretization on unstructured meshes and an implicit time discretization scheme. We use the PETSc library with index sets for parallelization. To solve the nonlinear problems on each time step, we compare two nonlinear Gauss-Seidel-type methods and Newton's method with several time-step sizes. We use GMRES in PETSc with multigrid preconditioning to solve the linear subproblems within the nonlinear solvers. We also study the scalability of this simulation on a cluster. Copyright © 2008 John Wiley & Sons, Ltd.

Received 26 February 2007; Revised 30 November 2007; Accepted 18 December 2007

KEY WORDS: incompressible MHD; streamfunction; finite element; multigrid method

1. INTRODUCTION

Unstructured meshes are popular in computational hydrodynamics and magnetohydrodynamics (MHD) because they gracefully accommodate adaptive local mesh refinement and irregular boundaries [1–4], and finite element methods have been analyzed on unstructured meshes for many

*Correspondence to: K. S. Kang, Brookhaven National Laboratory, Computational Science Center, Building 463, Room 255, Upton, NY 11973, U.S.A.
†E-mail: kskang@bnl.gov, kangkabseok@gmail.com

Contract/grant sponsor: U.S. Department of Energy; contract/grant numbers: DE-AC02-98CH10886, DE-FC02-04ER25595
Contract/grant sponsor: U.S. National Science Foundation; contract/grant number: CCF-03-52334

Copyright © 2008 John Wiley & Sons, Ltd.

problems [5–7]. Implicit time differencing has become popular because it removes the stability imposed time-step constraint, allowing much larger time steps [8–14]. Under implicit time differencing, nonlinear problems must be solved on each time-step. To solve the nonlinear problems, one can use nonlinear Gauss–Seidel iteration or global linearization techniques. Nonlinear Gauss–Seidel lacks a convergence guarantee, but often converges well in time-implicit problems in the limit of small time step. Newton's method based on global linearization [15–19] has a second-order convergence rate and inherits the scalability of the linear solver, which can be quite satisfactory up to thousands of processors [33].

Preconditioned Krylov iteration is a versatile linear solver for Newton [18, 19]. Among the various Krylov methods, GMRES [21] is often selected because it can guarantee convergence even for nonsymmetric, indefinite systems, if sufficient memory can be afforded, and the linear problem that is generated from some linearization of the nonlinear problem is usually nonsymmetric. As a preconditioner, we consider multigrid herein. Multigrid [22–25] represents an important advance in algorithmic efficiency for the solution of large problems. However, care is needed in using multigrid preconditioning for convectively dominated problems since the symmetric and nonsymmetric terms scale differently as the mesh is coarsened. We can overcome this difficulty by applying multigrid to reduced systems that possess symmetry. The preconditioning approach does not affect the accuracy of the final solution, but crucially determines convergence behavior.

MHD is the fluid dynamics of a conducting plasma, coupled with Maxwell's equations. The fluid velocity creates currents, which produce a Lorentz body force on the fluid. Ampere's law relates the currents to the magnetic field. The MHD approximation is that the electric field vanishes in the moving fluid frame, except for possible resistive effects. MHD is described by a higher-order system of partial differential equations (PDEs) than fluid dynamics. It admits additional waves, the shear and transverse Alfvén waves, magnetosonic waves, and their instabilities [3, 4, 26, 27].

Sharp structures can develop in MHD: nearly discontinuous magnetic fields and localized, intense current sheets. If the resolution is inadequate, truncation error leads to artificially high levels of numerical dissipation and magnetic reconnection. Therefore, it is important to refine the grids where current sheets form. Several authors [3, 4, 28] have employed adaptive mesh refinement in this context.

Strauss and Longcope [3] introduced a finite element discretization and Lankalapalli *et al*. [4] used a stabilized finite element formulation for the two-dimensional, incompressible MHD equations. They used the standard streamfunction–vorticity advection form of the equations, as well as the symmetrized current–vorticity advection formulation. The vorticity and current are time advanced and the potentials are found by solving the Poisson equations. In this streamfunction approach, we need the second-order derivatives of potential to obtain current density. To solve this higher-order problem, Jardin [29] uses a triangular finite element with first-derivative continuity. Otherwise, one needs to introduce four auxiliary variables, first-order partial derivatives of streamfunctions, and expand them in finite elements, i.e. four linear mass–matrix problems must be solved for first-order partial derivatives of streamfunctions. As a result, we have to solve eight scalar PDEs consisting of two time-dependent equations, two Poisson equations, and four mass–matrix equations in a standard streamfunction formulation with C^0 continuity. Kang [30] considers an alternative (hybrid) approach for solving the eight equations which uses the velocity and the magnetic field component that are the derivatives of potential functions to reduce the number of equations to six, consisting of two time-dependent equations and four Poisson equations. This approach is preferable for implicit time differencing and Newton's method to solve resulting nonlinear problems because its Jacobian matrix is smaller and contains four Poisson equations.

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

In this paper, we apply the finite element method on an unstructured mesh with an implicit time difference scheme to two-dimensional, incompressible MHD, using a hybrid streamfunction approach to enforce the divergence-free conditions on the magnetic and velocity fields. We focus on the behavior of the nonlinear solvers, which are two variants of nonlinear Gauss–Seidel, Newton's method. Preconditioned GMRES with multigrid preconditioning is employed to solve the linear subproblems within nonlinear solvers. In implementation, we use the PETSc library [31] with index sets for parallelization and study the scalability of this method on a cluster.

The remainder of the paper is organized as follows. Section 2 introduces the streamfunction formulations of MHD equations. Section 3 defines the finite element discretization in space and implicit finite difference scheme in time. In Section 4, we describe the three nonlinear solvers. Numerical results of the hybrid streamfunction approach for a tilt instability on finite domain are presented in Section 5, where the following are investigated:

1. effect of domain size, with respect to application of far-field boundary conditions;
2. convergence behavior of nonlinear and linear solvers according to time-step size;
3. parallel scalability.

We summarize in Section 6.

2. MHD EQUATIONS AND A STREAMFUNCTION FORMULATION

In this section, we review the two-dimensional incompressible MHD system and its stream function formulation as in Strauss and Longcope [3] and the hybrid streamfunction formulation [30].

MHD is governed by the Navier–Stokes equations with the magnetic force that is the macroscopic Lorentz force:

$$\mathbf{C}\times\mathbf{B}$$

where the relationship between the magnetic field $\mathbf{B}$ and the current density $\mathbf{C}$ is obtained from Ampére's law:

$$\nabla\times\mathbf{B}=\mathbf{C}$$

Hence, we have the Navier–Stokes equations with low Reynolds number for incompressible MHD

$$\rho\left(\frac{\partial}{\partial t}\mathbf{v}+\mathbf{v}\cdot\nabla\mathbf{v}\right)=(\nabla\times\mathbf{B})\times\mathbf{B}+v\rho\nabla^2\mathbf{v} \tag{1}$$

with the incompressibility condition

$$\nabla\cdot\mathbf{v}=0 \tag{2}$$

In an incompressible fluid, we can assume, without loss of generality, a homogenous density distribution normalized to $\rho=1$.

The dynamics of the magnetic field follows from Faraday's law

$$\frac{\partial}{\partial t}\mathbf{B}=-\nabla\times\mathbf{E}$$

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

and Ohm's law for an infinitely conducting fluid

$$\mathbf{E}+\mathbf{v}\times\mathbf{B}=0$$

with electric field **E**. Eliminating the electric field, we have

$$\frac{\partial}{\partial t}\mathbf{B}=\nabla\times(\mathbf{v}\times\mathbf{B}) \tag{3}$$

Therefore, the incompressible MHD equations appear in coordinate invariant, dimension invariant form, with the addition of the solenoidal constraint on the magnetic field for completeness, as

$$\frac{\partial}{\partial t}\mathbf{B}=\nabla\times(\mathbf{v}\times\mathbf{B}) \tag{4}$$

$$\frac{\partial}{\partial t}\mathbf{v}=-\mathbf{v}\cdot\nabla\mathbf{v}+(\nabla\times\mathbf{B})\times\mathbf{B}+\mu\nabla^2\mathbf{v} \tag{5}$$

$$\nabla\cdot\mathbf{v}=0 \tag{6}$$

$$\nabla\cdot\mathbf{B}=0 \tag{7}$$

where μ is the viscosity.

In two dimensions, to enforce incompressibility, it is common to introduce streamfunctions:

$$\mathbf{v}=\left(\frac{\partial\phi}{\partial y},-\frac{\partial\phi}{\partial x}\right),\quad \mathbf{B}=\left(\frac{\partial\psi}{\partial y},-\frac{\partial\psi}{\partial x}\right) \tag{8}$$

The MHD system (4)–(7) can be rewritten in terms of two scalar variables: the velocity streamfunction ϕ and the magnetic flux function ψ, i.e.,

$$\frac{\partial}{\partial t}\Omega+[\Omega,\phi]=[C,\psi]+\mu\nabla^2\Omega \tag{9}$$

$$\frac{\partial}{\partial t}\psi+[\psi,\phi]=0 \tag{10}$$

$$\nabla^2\phi=\Omega \tag{11}$$

$$C=\nabla^2\psi \tag{12}$$

where the two-dimensional Laplacian is

$$\nabla^2=\frac{\partial^2}{\partial x^2}+\frac{\partial^2}{\partial y^2}$$

the bracketed term is the commutator

$$[a,b]=\frac{\partial a}{\partial x}\frac{\partial b}{\partial y}-\frac{\partial a}{\partial y}\frac{\partial b}{\partial x}$$

the vorticity is $\Omega=\partial v_1/\partial y-\partial v_2/\partial x$, and the current density is $C=\partial B_1/\partial y-\partial B_2/\partial x$.

The left-hand side of (9), along with (11), is the familiar vorticity–streamfunction formulation of two-dimensional incompressible hydrodynamics. The evolution of the magnetic and velocity fields is treated in a nonsymmetric way in the standard formulation above. The velocity is advanced

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

through the vorticity, whereas the magnetic field is advanced via the magnetic potential. We consider an alternative symmetrical formulation the equations, in which the current and the vorticity are time advanced [3]. Instead of solving Equations (10) and (12), we take the Laplacian of (10) and use (11), (12), (8), and

$$\begin{aligned}\nabla^2[\psi,\phi]&=[\nabla^2\psi,\phi]+[\psi,\nabla^2\phi]+2\left(\frac{\partial^2\psi}{\partial x^2}\frac{\partial^2\phi}{\partial x\partial y}-\frac{\partial^2\psi}{\partial x\partial y}\frac{\partial^2\phi}{\partial x^2}+\frac{\partial^2\psi}{\partial x\partial y}\frac{\partial^2\phi}{\partial y^2}-\frac{\partial^2\psi}{\partial y^2}\frac{\partial^2\phi}{\partial x\partial y}\right)\\&=[C,\phi]-[\Omega,\psi]-2\left[\frac{\partial\phi}{\partial x},\frac{\partial\psi}{\partial x}\right]-2\left[\frac{\partial\phi}{\partial y},\frac{\partial\psi}{\partial y}\right]\end{aligned}$$

This yields an equation for the current density C, analogous to (9) for the vorticity:

$$\frac{\partial}{\partial t}\Omega+[\Omega,\phi]=[C,\psi]+\mu\nabla^2\Omega \tag{13}$$

$$\frac{\partial}{\partial t}C+[C,\phi]=[\Omega,\psi]+2\left[\frac{\partial\phi}{\partial x},\frac{\partial\psi}{\partial x}\right]+2\left[\frac{\partial\phi}{\partial y},\frac{\partial\psi}{\partial y}\right] \tag{14}$$

$$\nabla^2\phi=\Omega \tag{15}$$

$$\nabla^2\psi=C \tag{16}$$

The equations are now symmetrical, in the sense that the source functions Ω and C are time advanced, and the potentials ϕ and ψ are obtained at each time step by solving the Poisson equations (15) and (16).

To solve (14), we have to compute the first- and second-order partial derivatives of potentials, i.e., $\partial\phi/\partial x$, $\partial\phi/\partial y$, $\partial\psi/\partial x$, $\partial\psi/\partial y$, and their partial derivatives. These first partial derivatives can be obtained as solutions of mass equations or by differentiation of element basis functions. The second partial derivatives can be obtained from these first-order partial derivatives. To obtain the partial derivatives from mass equations, we have to introduce four auxiliary variables, which is the standard approach and requires the solution of eight equations at each step. To obtain the partial derivatives by differentiation, we have to use higher-order spaces [29] or special tricks on first-order spaces, because the second derivatives of first-order spaces vanish elementwise.

We consider the alternative (hybrid) approach introduced in [30] and use the velocity $\mathbf{v}$ and the magnetic field $\mathbf{B}$

$$\mathbf{v}=(v_1,v_2)=\left(\frac{\partial\phi}{\partial y},-\frac{\partial\phi}{\partial x}\right),\quad \mathbf{B}=(B_1,B_2)=\left(\frac{\partial\psi}{\partial y},-\frac{\partial\psi}{\partial x}\right)$$

in Equations (13)–(16), and obtain the following eight equations:

$$\frac{\partial\Omega}{\partial t}+(v_1,v_2)\cdot\nabla\Omega=(B_1,B_2)\cdot\nabla C+\mu\nabla^2\Omega \tag{17}$$

$$\frac{\partial C}{\partial t}+(v_1,v_2)\cdot\nabla C=(B_1,B_2)\cdot\nabla\Omega+2([v_1,B_1]+[v_2,B_2]) \tag{18}$$

$$-\nabla^2 v_1=-\frac{\partial\Omega}{\partial y} \tag{19}$$

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

$$-\nabla^2 v_2 = \frac{\partial \Omega}{\partial x} \tag{20}$$

$$-\nabla^2 B_1 = -\frac{\partial C}{\partial y} \tag{21}$$

$$-\nabla^2 B_2 = \frac{\partial C}{\partial x} \tag{22}$$

$$\nabla^2 \phi = \Omega \tag{23}$$

$$\nabla^2 \psi = C \tag{24}$$

The last two Poisson equations do not need to be solved to advance the solution. If the potentials are needed at a specific time, they are obtained by solving Equations (23) and (24).

The MHD equations conserve energy and magnetic flux [3, 26]. Since the magnetic flux function is advected with the flow, any function of ψ is a constant of the motion. The energy, $\mathscr{E}$, can be shown to be conserved by premultiplying the Ω evolution equation by ϕ, and the ψ evolution equation by C, and integrating by parts, or working directly from the primitive form of the equations. One obtains

$$\frac{\partial}{\partial t}\mathscr{E} = -\mu \int_K \Omega^2 \, \mathrm{d}^2 x$$

where

$$\mathscr{E} = \frac{1}{2}\int_K (\mathbf{v}^2 + \mathbf{B}^2)\, \mathrm{d}^2 x$$

assuming either Dirichlet boundary conditions, with ϕ, ψ constant on the boundary, Neumann conditions with the normal derivatives of ϕ, ψ equal to zero, or periodic boundary conditions.

To solve (17)–(24) on a finite domain, we have to impose boundary conditions on each variable. From the energy and magnetic flux conservation properties and uniqueness of solutions, we can choose Dirichlet boundary conditions for Ω, ϕ, and ψ in (17), (23), and (24). For current density C in (18), we can choose the Neumann boundary condition because, without resistivity, (18) is a first-order PDE. We have to impose compatible boundary conditions for the velocity and the magnetic field with Ω, C, ϕ, and ψ.

3. FINITE ELEMENT FORMULATIONS

In this section, we consider the discretization of (17)–(22). We use a first-order finite element discretization in space and complete with an implicit discretization of time.

Let $H^1(K)$ denote a standard Sobolev space of functions on domain K. To derive the variational (weak) formulation, we need the trial spaces that satisfy any Dirichlet conditions and test spaces that vanish on the Dirichlet part of boundary and have no restriction on the Neumann part of boundary. For $A = \Omega, v_1, v_2, B_1, B_2$, let $H^{1,A}$ denote the subset of $H^1(K)$ that satisfies the boundary condition of A, and $H^{1,A'}$ denote the subspace of $H^1(K)$ that vanishes on the Dirichlet boundary. Multiplying by the test functions and integrating by parts in each equation and using the appropriate boundary

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

conditions to determine the variational form of (17)–(22), we find $\mathbf{X}=(\Omega, C, v_1, v_2, B_1, B_2)\in H^{1,\Omega}\times H^1\times H^{1,v_1}\times H^{1,v_2}\times H^{1,B_1}\times H^{1,B_2}$ such that, for all $u\in H^{1,\Omega'}$, $w\in H^1$, $p_1\in H^{1,v_1'}$, $p_2\in H^{1,v_2'}$, $q_1\in H^{1,B_1'}$, $q_2\in H^{1,B_2'}$:

$$M_t(\Omega,u)+(\mathbf{v}\cdot\nabla\Omega,u)+\mu a(\Omega,u)=(B\cdot\nabla C,u) \tag{25}$$

$$M_t(C,w)+(\mathbf{v}\cdot\nabla C,w)=(B\cdot\nabla\Omega,w)+2P(v_1,B_1,w)+2P(v_2,B_2,w) \tag{26}$$

$$a(v_1,p_1)=-\left(\frac{\partial\Omega}{\partial y},p_1\right) \tag{27}$$

$$a(v_2,p_2)=\left(\frac{\partial\Omega}{\partial x},p_2\right) \tag{28}$$

$$a(B_1,q_1)=-\left(\frac{\partial C}{\partial y},q_1\right) \tag{29}$$

$$a(B_2,q_2)=\left(\frac{\partial C}{\partial x},q_2\right) \tag{30}$$

where

$$M_t(u,w)=\int_K\left(\frac{\partial}{\partial t}u\right)w\ \mathrm{d}x,\quad (u,w)=\int_K uw\,\mathrm{d}x$$

$$a(u,w)=\int_K\nabla u\cdot\nabla w\,\mathrm{d}x-\int_{\partial K}\frac{\partial v}{\partial n}w\ \mathrm{d}s\quad\text{and}\quad P(u,v,w)=\int_K[u,v]w\,\mathrm{d}x$$

To discretize the time derivative, we use the backward Euler method leading to an implicit scheme that removes the stability imposed time-step constraint, allowing much larger time steps. This approach is first-order accurate in time and is chosen merely for convenience, since the principal objective herein is the study of the solution of the nonlinear system. Higher-order backward difference formula treatments [32] have a similar algebraic structure, with additional history terms on the right-hand sides of the two prognostic equations.

Let $\mathscr{K}_h$ be given a triangulation of domain K with the maximum diameter h of the element triangles. Define the linear finite element spaces:

$$V_h=\{v\in L^2(K): v \text{ is linear on each element triangle of } \mathscr{K}_h \text{ and continuous in } K\}$$

Let V_h^A, $A=\Omega, v_1, v_2, B_1, B_2$, be the subsets of V_h that satisfy the boundary conditions of A on every boundary point of $\mathscr{K}_h$ and $V_h^{A'}$ be subspaces of V_h and $H^{1,A'}$. For notational convenience, let $\mathbf{X}=(\Omega, C, v_1, v_2, B_1, B_2)$, the six-vector of unknown fields and define the residual function components for the nonlinear problem at each time step:

$$F_1^n(\mathbf{X},u)=\frac{1}{\Delta t}(\Omega,u)+(\mathbf{v}\cdot\nabla\Omega,u)+\mu a(\Omega,u)-(\mathbf{B}\cdot\nabla C,u)-\frac{1}{\Delta t}(\Omega^{n-1},u)$$

$$F_2^n(\mathbf{X},w)=\frac{1}{\Delta t}(C,w)+(\mathbf{v}\cdot\nabla C,w)-(\mathbf{B}\cdot\nabla\Omega^n,w)-2P(v_1,B_1,w)$$
$$-2P(v_2,B_2,w)-\frac{1}{\Delta t}(C^{n-1},w)$$

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

$$F_3^n(\mathbf{X}, p_1) = a(v_1, p_1) + \left(\frac{\partial \Omega}{\partial y}, p_1\right)$$

$$F_4^n(\mathbf{X}, p_2) = a(v_2, p_2) - \left(\frac{\partial \Omega}{\partial x}, p_2\right)$$

$$F_5^n(\mathbf{X}, q_1) = a(B_1, q_1) + \left(\frac{\partial C}{\partial y}, q_1\right)$$

$$F_6^n(\mathbf{X}, q_2) = a(B_2, q_2) - \left(\frac{\partial C}{\partial x}, q_2\right)$$

For $k = 3, 4, 5, 6$, F_k^n is a linear constraint independent of time. Hence, we can express, for $k = 3, 4, 5, 6$,

$$F_k^n(\mathbf{X}, p) = F_k(\mathbf{X}, p)$$

We can express the discretized MHD problems as follows: For each n, find the solutions $\mathbf{X}^n = (\Omega_h^n, C_h^n, v_{1,h}^n, v_{2,h}^n, B_{1,h}^n, B_{2,h}^n) \in V_h^\Omega \times V_h \times V_h^{v_1} \times V_h^{v_2} \times V_h^{B_1} \times V_h^{B_2}$ that satisfy

$$\mathbf{F}^n(\mathbf{X}^n, \mathbf{Y}) = 0 \tag{31}$$

for all $\mathbf{Y} = (u, w, p_1, p_2, q_1, q_2) \in V_h^{\Omega'} \times V_h \times V_h^{v_1'} \times V_h^{v_2'} \times V_h^{B_1'} \times V_h^{B_2'}$, where

$$\mathbf{F}^n(\mathbf{X}, \mathbf{Y}) = \begin{pmatrix} F_1^n(\mathbf{X}, u) \\ F_2^n(\mathbf{X}, w) \\ F_3(\mathbf{X}, p_1) \\ F_4(\mathbf{X}, p_2) \\ F_5(\mathbf{X}, q_1) \\ F_6(\mathbf{X}, q_2) \end{pmatrix} \tag{32}$$

4. NONLINEAR AND LINEAR SOLVERS

In this section, we consider three nonlinear solvers for (32) and linear solvers within the nonlinear problem.

Equation (32) is a nonlinear problem in the six variables consisting of two time-dependent equations and four Poisson equations. However, if we consider the equations separately, each equation is a linear problem with respect to one variable. From this observation, it is natural to consider the nonlinear Gauss–Seidel iteration (GS1) that solves each linear equation in turn for its proper variable with recent approximate solutions.

The first nonlinear Gauss–Seidel (GS1): For each time n, initialize $\mathbf{X}^{n,0} = (\Omega_h^{n,0}, C_h^{n,0}, v_{1,h}^{n,0}, v_{2,h}^{n,0}, B_{1,h}^{n,0}, B_{2,h}^{n,0})$ and compute, for $k = 1, 2, \ldots,$

$$\mathbf{X}^{n,k} = \mathbf{X}^{n,k-1} + \delta \mathbf{X} \tag{33}$$

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

until $\mathbf{X}^{n,k}$ satisfies

$$\|F(\mathbf{X}^{n,k}, \mathbf{Y})\| \leqslant \gamma_n$$

where $\delta\mathbf{X} = (\delta\Omega, \delta C, \delta v_1, \delta v_2, \delta B_1, \delta B_2)$ is the solution of

$$\frac{\partial F_1^n}{\partial \Omega}(\delta\Omega, v_{1,h}^{n,k-1}, v_{2,h}^{n,k-1}, u) = -F_1^n((\Omega_h^{n,k-1}, C_h^{n,k-1}, v_{1,h}^{n,k-1}, v_{2,h}^{n,k-1}, B_{1,h}^{n,k-1} B_{2,h}^{n,k-1}), u) \tag{34}$$

$$\frac{\partial F_2^n}{\partial C}(\delta C, v_{1,h}^{n,k-1}, v_{2,h}^{n,k-1}, w) = -F_2^n((\Omega_h^{n,k}, C_h^{n,k-1}, v_{1,h}^{n,k-1}, v_{2,h}^{n,k-1}, B_{1,h}^{n,k-1} B_{2,h}^{n,k-1}), w) \tag{35}$$

$$a(\delta v_1, p_1) = -F_3((\Omega_h^{n,k}, v_{1,h}^{n,k-1}), p_1) \tag{36}$$

$$a(\delta v_2, p_2) = -F_4((\Omega_h^{n,k}, v_{2,h}^{n,k-1}), p_2) \tag{37}$$

$$a(\delta B_1, q_1) = -F_5((C_h^{n,k}, B_{1,h}^{n,k-1}), q_1) \tag{38}$$

$$a(\delta B_2, q_2) = -F_6((C_h^{n,k}, B_{2,h}^{n,k-1}), q_2) \tag{39}$$

where

$$\frac{\partial F_1}{\partial \Omega}(\delta\Omega, v_1, v_2, u) = \frac{1}{\Delta t} M(\delta\Omega, u) + ((v_1, v_2) \cdot \nabla \delta\Omega, u) + \mu a(\delta\Omega, u)$$

$$\frac{\partial F_2}{\partial C}(\delta C, v_1, v_2, w) = \frac{1}{\Delta t} M(\delta C, w) + ((v_1, v_2) \cdot \nabla \delta C, w)$$

$$\|F(\mathbf{X}, \mathbf{Y})\|^2 = \|F_1^n(\mathbf{X}, u)\|^2 + \|F_2^n(\mathbf{X}, w)\|^2 + \|F_3(\mathbf{X}, p_1)\|^2 + \|F_4(\mathbf{X}, p_2)\|^2 + \|F_5(\mathbf{X}, q_1)\|^2 + \|F_6(\mathbf{X}, q_2)\|^2$$

and γ_n is a given convergence tolerance.

Remark 4.1
In GS1, problems (36)–(39) are Poisson problems for which a multigrid solver can be used.

The first two equations in (32) comprise a pair of time-dependent linear equation with respect to two variables Ω and C, which can be solved simultaneously and remaining four Poisson equations separately. This is a variant nonlinear Gauss–Seidel iteration (GS2).

The second nonlinear Gauss–Seidel (*GS2*): For each time n, initialize $\mathbf{X}^{n,0}$ and compute, for $k = 1, 2, \ldots,$

$$\mathbf{X}^{n,k} = \mathbf{X}^{n,k-1} + \delta\mathbf{X} \tag{40}$$

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

until $\mathbf{X}^{n,k}$ satisfies

$$\|F(\mathbf{X}^{n,k}, \mathbf{Y})\| \leqslant \gamma_n$$

where $\delta\mathbf{X} = (\delta\Omega, \delta C, \delta v_1, \delta v_2, \delta B_1, \delta B_2)$ is the solution of

$$\frac{\partial(F_1^n, F_2^n)}{\partial(\Omega, C)}(\delta\Omega, \delta C, u, w | v_{1,h}^{n,k-1}, v_{2,h}^{n,k-1}, B_{1,h}^{n,k-1} B_{2,h}^{n,k-1}) = \begin{pmatrix} -F_1^n(\mathbf{X}^{n,k-1}, u) \\ -F_2^n(\mathbf{X}^{n,k-1}, w) \end{pmatrix} \tag{41}$$

where

$$\frac{\partial(F_1^n, F_2^n)}{\partial(\Omega, C)}(\delta\Omega, \delta C, u, w | v_1, v_2, B_1, B_2)$$

$$= \begin{pmatrix} \dfrac{\partial F_1}{\partial \Omega}(\delta\Omega, v_1, v_2, u) & -((B_1, B_2)\cdot\nabla\delta C, u) \\ -((B_1, B_2)\cdot\nabla\delta\Omega, w) & \dfrac{\partial F_2}{\partial C}(\delta C, v_1, v_2, w) \end{pmatrix}$$

followed by (36)–(39).

Remark 4.2
The implicit time difference ensures absolutely stable numerics, for any time step and level of mesh refinement, at price of solving globally connected systems.

Next we consider Newton's method. Newton's method has, asymptotically, second-order convergence for nonlinear problems and optimality with respect to mesh refinement, but requires computation of the Jacobian of nonlinear problem, which can be complicated.

Newton's method (*NM*): For each time n, initialize $\mathbf{X}^{n,0}$ and compute, for $k = 1, 2, \ldots,$

$$\mathbf{X}^{n,k} = \mathbf{X}^{n,k-1} + \delta\mathbf{X} \tag{42}$$

until $\mathbf{X}^{n,k}$ satisfies

$$\|F(\mathbf{X}^{n,k}, \mathbf{Y})\| \leqslant \gamma_n$$

where $\delta\mathbf{X} = (\delta\Omega, \delta C, \delta v_1, \delta v_2, \delta B_1, \delta B_2)$ is the solution of

$$\frac{\partial \mathbf{F}^n}{\partial \mathbf{X}}(\delta\mathbf{X}, \mathbf{Y}) = \begin{pmatrix} -F_1^n(\mathbf{X}^{n,k-1}, u) \\ -F_2^n(\mathbf{X}^{n,k-1}, w) \\ -F_3(\mathbf{X}^{n,k-1}, p_1) \\ -F_4(\mathbf{X}^{n,k-1}, p_2) \\ -F_5(\mathbf{X}^{n,k-1}, q_1) \\ -F_6(\mathbf{X}^{n,k-1}, q_2) \end{pmatrix} \tag{43}$$

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

where $\partial \mathbf{F}^n / \partial \mathbf{X}$ be the Jacobian of (25), i.e.

$$J_k = \begin{pmatrix} \frac{\partial F_1^n}{\partial \Omega} & \frac{\partial F_1^n}{\partial C} & \frac{\partial F_1^n}{\partial v_1} & \frac{\partial F_1^n}{\partial v_2} & \frac{\partial F_1^n}{\partial B_1} & \frac{\partial F_1^n}{\partial B_2} \\ \frac{\partial F_2^n}{\partial \Omega} & \frac{\partial F_2^n}{\partial C} & \frac{\partial F_2^n}{\partial v_1} & \frac{\partial F_2^n}{\partial v_2} & \frac{\partial F_2^n}{\partial B_1} & \frac{\partial F_2^n}{\partial B_2} \\ \frac{\partial F_3}{\partial \Omega} & 0 & \frac{\partial F_3}{\partial v_1} & 0 & 0 & 0 \\ \frac{\partial F_4}{\partial \Omega} & 0 & 0 & \frac{\partial F_4}{\partial v_2} & 0 & 0 \\ 0 & \frac{\partial F_5}{\partial C} & 0 & 0 & \frac{\partial F_5}{\partial B_1} & 0 \\ 0 & \frac{\partial F_6}{\partial C} & 0 & 0 & 0 & \frac{\partial F_6}{\partial B_2} \end{pmatrix} \tag{44}$$

Remark 4.3
From the above, we observe that the first two block rows have parabolic operators with identity matrices scaled by $1/\Delta t$ on the diagonal, and the last four have Laplacians.

In all three nonlinear solvers, we need to solve large sparse linear problems. Among the various Krylov methods available, we use GMRES because it is well-known solver for nonsymmetric, nonpositive definite systems. However, GMRES can be memory intensive (storage increases linearly with the number of GMRES iterations per Jacobian solve) and expensive (computational complexity of GMRES increases with the square of the number of GMRES iterations per Jacobian solve). Restarted GMRES can in principle deal with these limitations; however, it lacks a theory of convergence, and stalling is frequently observed in real applications.

Preconditioning consists in operating on the system matrix J_k where

$$J_k \delta x_k = -F(x_k) \tag{45}$$

with an operator P_k^{-1} (preconditioner) such that $J_k P_k^{-1}$ (right preconditioning) or $P_k^{-1} J_k$ (left preconditioning) is well conditioned. In this study, we use left preconditioning. The choice of P_k^{-1} does not affect the accuracy of the Newton correction, but crucially determines convergence behavior of GMRES and, hence, the efficiency of the algorithm.

In this study, we use multigrid, which is well known as a scalable preconditioner, as well as a scalable solver in unaccelerated form, for many problems. Multigrid is excellent for symmetric diagonally dominant problems (36)–(39). The Equations (34), (35), (41), and (43) possess nonsymmetry that depends up on the magnitudes of (v_1, v_2) and (B_1, B_2) and on the time-step sizes.

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

We consider the diagonal term of (41) and (43) as a reduced system, i.e.

$$J_{R,k}=\begin{pmatrix} \dfrac{\partial F_1^n}{\partial \Omega} & 0 \\ 0 & \dfrac{\partial F_2^n}{\partial C} \end{pmatrix}, \quad J_{R,k}=\begin{pmatrix} \dfrac{\partial F_1^n}{\partial \Omega} & 0 & 0 & 0 & 0 & 0 \\ 0 & \dfrac{\partial F_2^n}{\partial C} & 0 & 0 & 0 & 0 \\ 0 & 0 & \dfrac{\partial F_3}{\partial v_1} & 0 & 0 & 0 \\ 0 & 0 & 0 & \dfrac{\partial F_4}{\partial v_2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \dfrac{\partial F_5}{\partial B_1} & 0 \\ 0 & 0 & 0 & 0 & 0 & \dfrac{\partial F_6}{\partial B_2} \end{pmatrix} \tag{46}$$

Next, we consider the symmetrized diagonal term of (46) as a reduced system, i.e.

$$J_{S,k}=\tfrac{1}{2}(J_{R,k}+J_{R,k}^{\mathrm{T}}) \tag{47}$$

The reduced systems $J_{R,k}$ and $J_{S,k}$ are constructed analytically. The reduced system $J_{S,k}$ is interesting for multigrid when $J_{R,k}$ is highly nonsymmetric. In the numerical experiments in the following section, we compare the behaviors of two multigrid preconditionings applying these reduced systems.

To implement the finite element solver for two-dimensional, incompressible MHD on parallel machines, we use the PETSc library [31], which is well developed for solving nonlinear PDE problems and implements a geometric multigrid preconditioner with GMRES. To obtain finer level from coarse level, we use subdivision process by connecting mid-points of sides of each

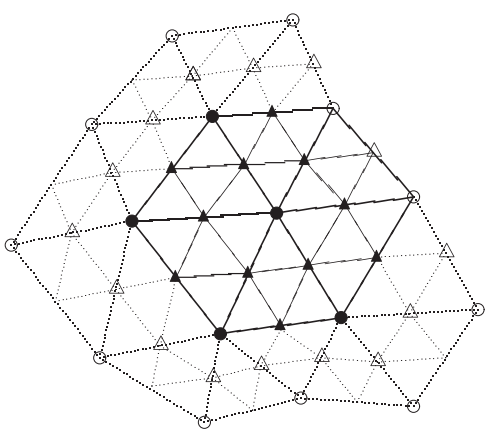

Figure 1. Real points and ghost points sub-domain Ω_j on processor j (solid line Ω_j, thick lines and circles for coarser level, and thin lines and triangles for finer level. Filled circles and triangles are real points on processor j and unfilled circles and triangles are ghost points on processor j.)

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

triangles in discretized triangles. We assign sub-domains to each processors only coarsest level on unstructured finite element discretization and obtain finer levels on each sub-domains. To do this, we use PETSc index sets for global orderings and ghost points for local orderings. In Figure 1, we show real points and ghost points in a sub domain and two levels of discretization. Each processor handles all variable and equations that are defined on each sub-domains and all levels in the discretization.

5. NUMERICAL EXPERIMENTS: TILT INSTABILITY

In this section, we apply the methods outlined to the tilt instability problem and present numerical results as the effect of domain resolution, the convergence behavior of nonlinear and linear problems, and scalability on a cluster.

To solve (13)–(16), we need initial and boundary conditions for Ω, C, ϕ, and ψ. The initial condition is a perturbation of the state:

$$\psi = \begin{cases} [2/kJ_0(k)]J_1(kr)\dfrac{y}{r}, & r<1 \\ \left(\dfrac{1}{r-r}\right)\dfrac{y}{r}, & r>1 \end{cases} \tag{48}$$

where J_n be the Bessel function order n, k be any constant that satisfies $J_1(k)=0$, and $r=\sqrt{x^2+y^2}$. Many authors [3, 4, 29, 30] have investigated this initial condition for the tilt instability problem with a perturbation of ϕ such that

$$\Omega(x, y, 0) = 0.0 \tag{49}$$

$$C(x, y, 0) = \begin{cases} 19.0272743J_1(kr)y/r & \text{if } r<1 \\ 0.0 & \text{if } r>1 \end{cases} \tag{50}$$

$$\phi(x, y, 0) = 10^{-3}\mathrm{e}^{-(x^2+y^2)} \tag{51}$$

$$\psi(x, y, 0) = \begin{cases} -1.295961618J_1(kr)y/r & \text{if } r<1 \\ -\left(\dfrac{1}{r}-r\right)y/r & \text{if } r>1 \end{cases} \tag{52}$$

where $k=3.831705970$.

In [3, 4, 30, 33], (13)–(16) are solved on finite domain with Dirichlet boundary conditions $\Omega(x, y, t)=0$, $\phi(x, y, t)=0$, and $\psi(x, y, t)=y-y/(x^2+y^2)$ and homogeneous Neumann boundary condition for C.

In our numerical experiments, we solve (17)–(24) on the finite square domain $K=[-R, R]\times[-R, R]$ and use the above initial and boundary condition for Ω, C, ϕ, and ψ. The initial and boundary conditions for velocity $\mathbf{v}$ and magnetic field $\mathbf{B}$ are compatibly derived from the initial and boundary conditions of Ω, C, ϕ, and ψ, as mentioned in Section 2. The initial condition of the velocity and the magnetic field are as follows:

$$v_1(x, y, 0) = -2y10^{-3}\mathrm{e}^{-(x^2+y^2)}$$

$$v_2(x, y, 0) = 2x10^{-3}\mathrm{e}^{-(x^2+y^2)}$$

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

$$B_1(x,y,0)=\frac{\partial\psi(0)}{\partial y}=\begin{cases}-1.295961618\left(\dfrac{ky^2}{r^2}J_0(kr)+\dfrac{x^2-y^2}{r^3}J_1(kr)\right) & \text{if } r<1\\ 1-\dfrac{x^2-y^2}{r^4} & \text{if } r>1\end{cases}$$

$$B_2(x,y,0)=-\frac{\partial\psi(0)}{\partial x}=\begin{cases}1.295961618\left(\dfrac{kxy}{r^2}J_0(kr)-\dfrac{2xy}{r^3}J_1(kr)\right) & \text{if } r<1\\ -\dfrac{2xy}{r^4} & \text{if } r>1\end{cases}$$

and the boundary conditions of the velocity and the magnetic field are

$$v_1(x,y,t)=0 \quad \text{on } x=\pm R$$

$$\frac{\partial v_1}{\partial n}(x,y,t)=0 \quad \text{on } y=\pm R$$

$$v_2(x,y,t)=0 \quad \text{on } y=\pm R$$

$$\frac{\partial v_2}{\partial n}(x,y,t)=0 \quad \text{on } x=\pm R$$

$$B_1(x,y,t)=1-\frac{x^2-y^2}{r^4} \quad \text{on } x=\pm R$$

$$\frac{\partial B_1}{\partial n}(x,y,t)=\begin{cases}-\dfrac{2y(3x^2-y^2)}{(x^2+y^2)^3} & \text{on } y=-R\\ \dfrac{2y(3x^2-y^2)}{(x^2+y^2)^3} & \text{on } y=R\end{cases}$$

$$B_2(x,y,t)=-\frac{2xy}{r^4} \quad \text{on } y=\pm R$$

$$\frac{\partial B_2}{\partial n}(x,y,t)=\begin{cases}-\dfrac{2y(3x^2-y^2)}{(x^2+y^2)^3} & \text{on } x=-R\\ \dfrac{2y(3x^2-y^2)}{(x^2+y^2)^3} & \text{on } x=R\end{cases}$$

For the multigrid preconditioner, we start with the coarsest grid of the square domain $[-3,3]\times[-3,3]$, which has 136 triangles, 216 edges, and 31 vertices, and generate finer grids by recursively subdividing the coarsest grid. Figure 2 shows the triangulations of coarsest grid and level 5 grid that is generated by four recursive subdivisions. Numerical simulation results are illustrated in Figure 3.

The tilt instability problem is defined on an unbounded domain, but we simulate this problem on the bounded square domain. Hence, we are obligated to choose a sufficiently large domain compared with the size of the current perturbations to keep finite boundary effects beneath those of other errors. To investigate the effect of the size of domains, we compare our two formulations the

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

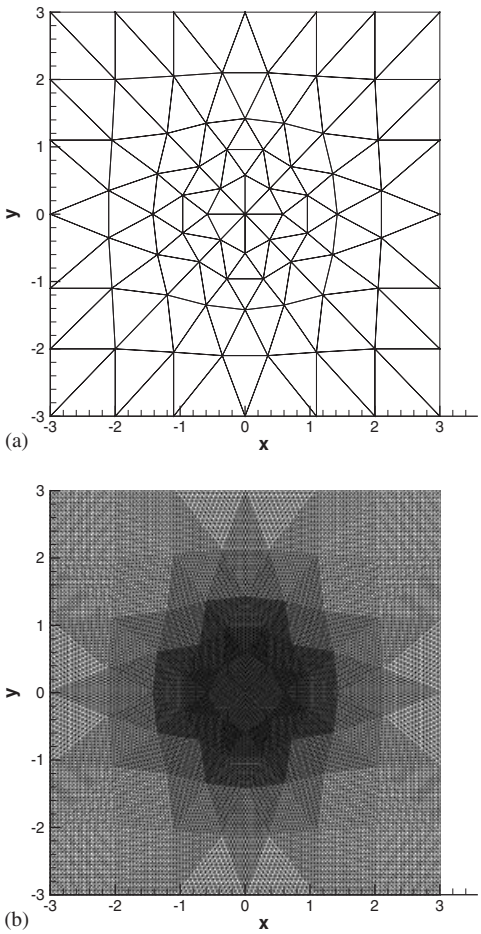

Figure 2. Unstructured nested meshes on coarsest and finest levels: (a) coarsest grid and (b) level 5 grid.

standard formulation, using ϕ, ψ, and its derivatives, and the *hybrid* formulation, using **v** and **B**, on the square domains with various R. These numerical simulation results are illustrated in Figure 4, which shows the contours of ψ at $t=7.0$, and in Figure 5, which shows the kinetic energies. The average growth rate γ of kinetic energy is shown in Table I. These results show that the solutions of two formulations are closer when the domain is enlarged with the standard streamfunction formula converging from above and the hybrid streamfunction formula converging below.

Henceforth, we consider the convergence behaviors of several nonlinear and linear solvers as a function of time-step sizes for the hybrid streamfunction formulation only. We use the convergence tolerance $\gamma_n = 10^{-8}$ for nonlinear problem on each time step and the relative convergence tolerance 10^{-6} for linear problem within the nonlinear problem. In linear problems, we use the Jacobi smoother with two times smoothing for pre- and post-smoothing steps because the Jacobi smoother is simple to implement on parallel machines and do not change the behavior when the number of processors is changed. In Table II, we report the number of nonlinear iterations of nonlinear

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

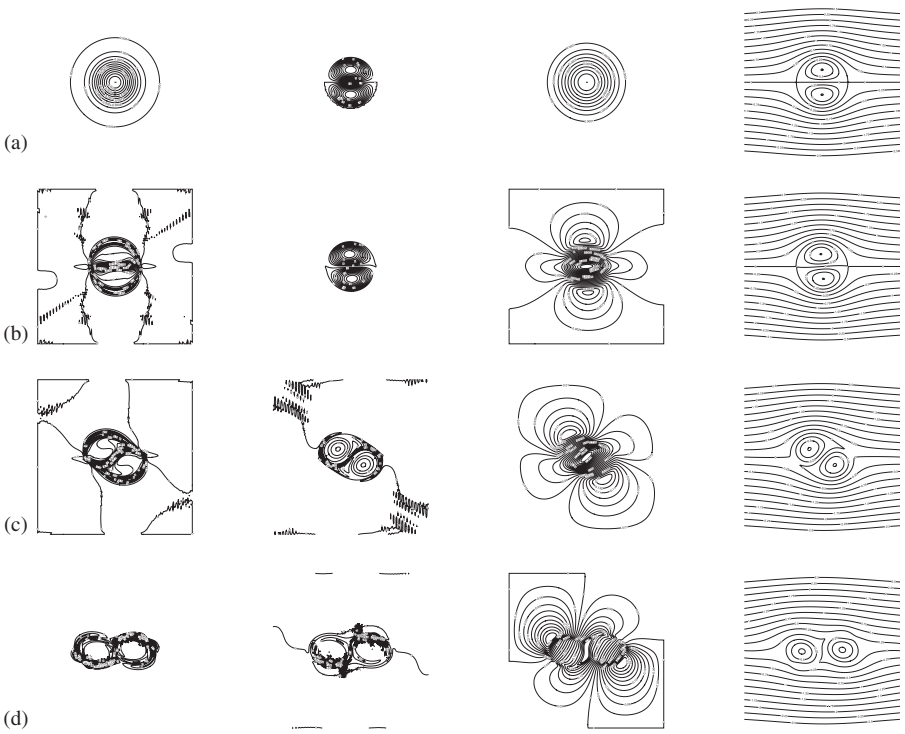

Figure 3. Contours of Ω, C, ϕ, and ψ at time $t=0.0$ (a), 4.0 (b), 6.0 (c), and 7.0 (d) of hybrid streamfunction formulation.

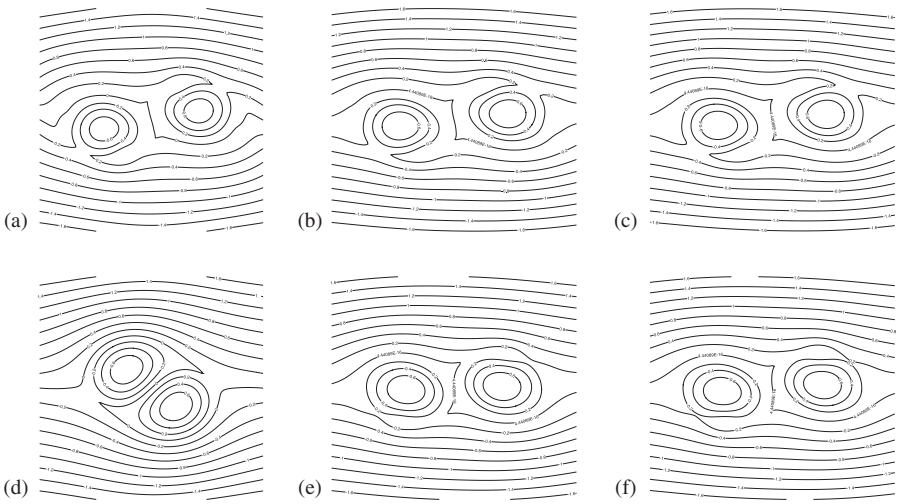

Figure 4. Results of finite boundaries: contours of ψ at $t=7.0$. Standard streamfunction formula: (a) $R=2$; (b) $R=3$; and (c) $R=3.5$ and hybrid streamfunction formula: (d) $R=2$; (e) $R=3$; and (f) $R=3.5$.

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

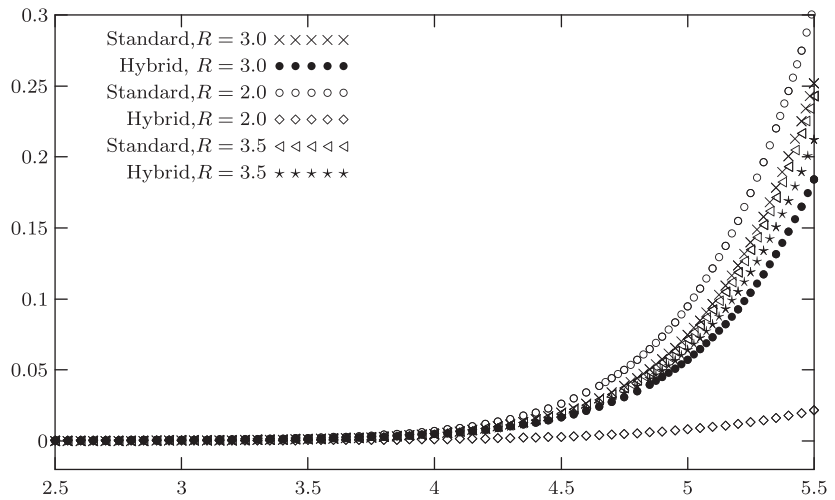

Figure 5. Results of finite boundaries: kinetic energy as a function of time $2.5 < t < 5.5$.

Table I. Average growth rate γ of kinetic energy from $t = 0.0$ to 6.0.

Standard streamfunction			Hybrid streamfunction		
$R=2$	$R=3$	$R=3.5$	$R=2$	$R=3$	$R=3.5$
2.167	2.152	2.148	1.744	2.102	2.125

Table II. The average number of nonlinear iterations for early and late time according to time-step size dt for the hybrid streamfunction formulation.

Starting time	dt	GS1	GS2	NM
$t=0.0$	0.0005	3	3	2
	0.001	4	3	3
	0.002	5	3	3
	0.005	12	4	3
	0.01	*	4	4
	0.02	*	6	5
$t=6.0$	0.0005	4	4	4
	0.001	4	4	4
	0.002	5	5	5
	0.005	8	6	5
	0.01	*	8	7
	0.02	*	13	11

solvers as a function of time-step sizes for the fixed starting time and fixed mesh level 5. We choose $t = 0$ and 6 as the starting time because many simulations have trouble at start up time and the magnitude of the velocity (v_1, v_2) and the magnetic field (B_1, B_2) increase with time which

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

cause increasing nonsymmetry. These numerical results show that GS2 and Newton method are more nonlinearly robust than GS1.

To illustrate convergence behavior of the linear solvers, we report the average number of linear iterations in one time step according to preconditioning in Table III. Numerical results show that multigrid preconditioning applied to the symmetrized reduced system is robust at both $t=0.0$ and 6.0, but multigrid applied to the reduced system is robust only at $t=0.0$, where it is very similar to the symmetrized case, because the values of velocity are small. These results show the value of deriving the multigrid preconditioning from the symmetrized reduced system.

In Table IV, we report the average number of nonlinear and linear iterations over the 10 steps from $t=0.0$ to 0.05 with $\mathrm{d}t=0.005$ according to the number of levels used in multigrid. The methods GS2(S) and Newton method (S) have very similar convergence behaviors.

To assess scalability, we measure the simulation time from $t=0.0$ to 0.5 with $\mathrm{d}t=0.005$ for the second nonlinear Gauss–Seidel and Newton methods with symmetrized diagonal term for the hybrid streamfunction formulation (17)–(22). We execute on the heterogeneous BGC (Galaxy

Table III. The average number of linear iterations over all Newton steps for early and late time according to time-step size for the hybrid streamfunction formulation.

Starting time	dt	GS1	GS2	GS2(R)	GS2(S)	NM(R)	NM(S)
$t=0.0$	0.0005	4.4	4	4.3	4.3	5	5
	0.001	4.4	4	5.3	5.3	6	6
	0.002	4.4	4.3	6.6	6.6	7	7
	0.005	3.4	*	11	11	12	12
	0.01	*	*	18	18	18.5	18.5
	0.02	*	*	28.8	28.8	31.6	31.6
$t=6.0$	0.0005	4.4	4.4	4	4	5	5
	0.001	4.4	4	4.5	5	5	5
	0.002	4.4	4	5	5.2	6.8	6.25
	0.005	3.9, 4.2	*	7.8	8.8	*	10
	0.01	*	*	*	15.2	*	17
	0.02	*	*	*	27.3	*	33.3

For GS1, the result for each unblocked prognostic equation is reported separately.

Table IV. Average number of iterations with $\mathrm{d}t=0.005$ for the hybrid streamfunction formulation of the second nonlinear Gauss–Seidel and Newton method with symmetrized diagonal term.

Solvers	Level	Nonlinear	Linear
GS2(S)	4	4	7.9
	5	3.1	11.2
	6	3	16.1
	7	3.4	19.1
NM(S)	4	3	8
	5	3	11.7
	6	3	16.4
	7	3.4	20.1

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

Cluster) machine at Brookhaven National Laboratory, which consists of 256 Intel P3 and P4 dual processor nodes running at speeds up to 2.4 GHz with 1 Gbyte of memory per node and Cheetah, a

Table V. Average solution time according to the number of processors (in weak scaling).

			Number of processors						
Machine	Solver	DoF/P	2	4	8	16	32	64	128
BGC	GS2(S)	2000	133		386		1180		
		8000	425		1205		2263		
	NM(S)	2000	77.6		199		616		
		8000	335		599		1429		
Cheetah	GS2(S)	1000		37.02		47.90		75.49	
		2000	67.37		76.58		110.7		239.5
		4000		130.1		216.1		311.7	
		8000	255.8		351.0		503.0		
	NM(S)	1000		14.27		17.86		28.37	
		2000	23.08		30.84		41.21		82.09
		4000		50.41		91.43		117.5	
		8000	110.3		149.8		204.1		

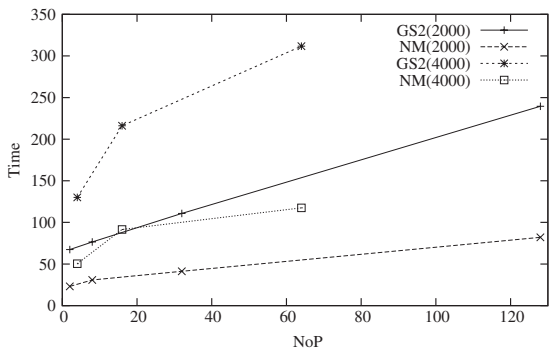

Figure 6. Solution time for a fixed number of degree of freedom per processor (weak scaling).

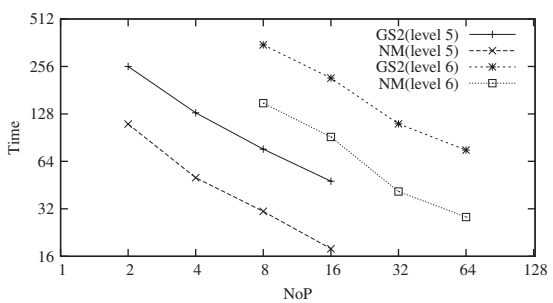

Figure 7. Solution time for fixed level according to the number of processors (strong scaling).

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

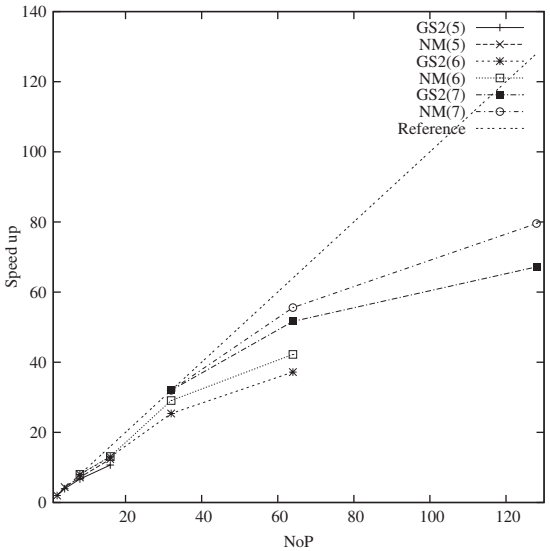

Figure 8. Speed-up for fixed level according to the number of processors.

4.5 Tflop/s IBM pSeries system at Oak Ridge National Laboratory that consists of 27 p690 nodes, each node in turn consisting of thirty-two 1.3 GHz Power4 processors and connected via IBM's Federation interconnect. We report the simulation times according to the number processors, with fixed problem size per processor, in Table V and plot the weak scalability in Figure 6, the strong scalability in Figure 7, and speed-up in Figure 8. These various profiles show that Newton's method has a better scalability and is faster in absolute terms than the Gauss–Seidel iteration.

6. CONCLUSIONS

We study the hybrid streamfunction approach method for the two-dimensional, incompressible MHD with the finite element method for the canonical problem of the tilt instability. We show that nonlinear Gauss–Seidel (GS2) and Newton's methods have similar convergence behaviors and that multigrid preconditioning applied on the symmetrized reduced system provides good linear convergence.

ACKNOWLEDGEMENTS

This article has been authored by Brookhaven Science Associates, LLC under Contract No. DE-AC02-98CH10886 with the U.S. Department of Energy. The United States Government retains, and the publisher, by accepting the article for publication, acknowledges, a world-wide license to publish or reproduce the publication form of this manuscript, or allow others to do so, for the United States Government purposes. We gratefully acknowledge support from the National Science Foundation under grant CCF-03-52334 and the Department of Energy under cooperative agreement DE-FC02-04ER25595.

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld

REFERENCES

1. Gropp WD, Kaushik DK, Keyes DE, Smith BF. Performance modeling and tuning of an unstructured mesh CFD application. *Proceedings of SC2000*. IEEE Computer Society: Los Alamitos, 2000.
2. Mavriplis DJ. An assessment of linear versus non-linear multigrid methods for unstructured mesh solvers. *Journal of Computational Physics* 2002; **175**:302–325.
3. Strauss HR, Longcope DW. An adaptive finite element method for magnetohydrodynamics. *Journal of Computational Physics* 1998; **147**:318–336.
4. Lankalapalli S, Flaherty JE, Shephard MS, Strauss H. Adaptive finite element analysis and error estimation for magnetohydrodynamics. *Journal of Computational Physics* 2006. DOI: 10.1016/j.jcp.2006.12.010.
5. Brenner SC, Scott LR. *The Mathematical Theory of Finite Element Methods*. Springer: New York, 1994.
6. Ciarlet PG. *The Finite Element Method for Elliptic Problems*. North-Holland: Amsterdam, New York, Oxford, 1988.
7. Girault V, Raviart PA. *Finite Element Methods for Navier–Stokes Equations*, *Theory and Algorithms*. Springer: Berlin, New York, 1986.
8. Chacon L, Knoll DA, Finn JM. An implicit, nonlinear reduced resistive MHD solver. *Journal of Computational Physics* 2002; **178**:15–36.
9. Gropp WD, Kaushik DK, Keyes DK, Smith BF. High performance parallel implicit CFD. *Parallel Computing* 2001; **27**:337–362.
10. Hujeirat A, Rannacher R. On the efficiency and robustness of implicit methods in computational astrophysics. *New Astronomy Reviews* 2001; **45**:425–447.
11. Jones OS, Shumlak U, Eberhardt DS. An implicit scheme for nonideal magnetohydrodynamics. *Journal of Computational Physics* 1997; **130**:231–242.
12. Keyes DE. Terascale implicit methods for partial differential equations. *Contemporary Mathematics*, vol. 306. AMS: Providence, 2001; 29–84.
13. Knoll DA, Lapenta G, Brackbill JU. A multilevel iterative field solver for implicit, kinetic, plasma simulation. *Journal of Computational Physics* 1999; **149**:377–388.
14. Turek A. A comparison study of time-stepping techniques for the incompressible Navier–Stokes equations: from fully implicit non-linear schemes to semi-implicit projection methods. *International Journal for Numerical Methods in Fluids* 1996; **22**:987–1011.
15. Cai X-C, Keyes DE. Nonlinearly preconditioned inexact Newton algorithms. *SIAM Journal on Scientific Computing* 2002; **24**:183–200.
16. D'Amico D. A Newton–Raphson approach for nonlinear diffusion equations in radiation hydrodynamics. *Journal of Quantitative Spectroscopy and Radiative Transfer* 1995; **54**:655–669.
17. Eisenstat SC, Walker HF. Globally convergent inexact Newton methods. *SIAM Journal on Optimization* 1994; **4**:393–422.
18. Knoll DA, Keyes DE. Jacobian-free Newton–Krylov methods: a survey of approach and applications. *Journal of Computational Physics* 2004; **193**:357–397.
19. Knoll DA, Rider MJ. A multigrid preconditioned Newton–Krylov method. *SIAM Journal on Scientific Computing* 1999; **21**:691–710.
20. Kaushik DK, Keyes DE, Smith BF. NKS methods for compressible and incompressible flows on unstructured grids. In *Proceedings of the 11th International Conference on Domain Decomposition Methods*, Lai C-H *et al*. (eds). Springer: Berlin, 1998; 513–520.
21. Saad Y, Schultz M. GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal on Scientific and Statistical Computing* 1986; **7**:856–869.
22. Bramble J. *Multigrid Methods*. Pitman: London, 1993.
23. Brandt A. *Multigrid Techniques*: *1984 Guide*, *with Applications to Fluid Dynamics*. GMD, Forschungszentrum Informationstechnik: St. Augustin, Germany, 1984.
24. Hackbush W. *Multigrid Methods and Applications*. Springer: Berlin, 1985.
25. Wesseling P. *An Introduction to Multigrid Methods*. Wiley: Chichester, U.K., 1992.
26. Biskamp D. *Nonlinear Magnetohydrodynamics*. Cambridge University Press: Cambridge, 1997.
27. Davidson PA. *An Introduction to Magnetohydrodynamics*. Cambridge University Press: Cambridge, 2001.
28. Philip B, Pernice M, Chacón L. Solution of reduced resistive magnetodynamics using implicit adaptive mesh refinement. In *Domain Decomposition Methods in Science and Engineering*, vol. XVI, Widlund OB, Keyes DE (eds). Springer: Berlin, Heidelberg, New York, 2006; 723–729.

Copyright © 2008 John Wiley & Sons, Ltd.

29. Jardin SC. A triangular finite element with first-derivative continuity applied to fusion MHD applications. *Journal of Computational Physics* 2004; **200**:133–152.
30. Kang KS. New stream function approach method for magnetohydrodynamics. In *Domain Decomposition Methods in Science and Engineering*, vol. XVI, Widlund OB, Keyes DE (eds). Springer: Berlin, Heidelberg, New York, 2006; 619–626.
31. Balay S, Gropp WD, McInnes LC, Smith BF. PETSc Users Manual. *Technical Report ANL-95/11—Revision 2.1.3*, Argonne National Laboratory, 2001.
32. Brenan KE, Campbell SL, Petzold LR. *Numerical Solution of Initial Value Problems in Differential–Algebraic Equations*. SIAM: Philadelphia, 1995.
33. Richard RL, Sydora RD, Ashour-Abdalla M. Magnetic reconnection driven by current repulsion. *Physics of Fluids B* 1990; **2**:488–494.

Copyright © 2008 John Wiley & Sons, Ltd.

DOI: 10.1002/fld