New Streamfunction Approach for Magnetohydrodynamics

Kab Seok Kang

Brookhaven National Laboratory, Computational Science Center, Building 463, Room 255, Upton NY 11973, USA. kskang@bnl.gov

Summary. We apply the finite element method to two-dimensional, incompressible MHD, using a streamfunction approach to enforce the divergence-free conditions on the magnetic and velocity fields. This problem was considered by Strauss and Longcope [SL98]. In this paper, we solve the problems with magnetic and velocity fields instead of the velocity stream function, magnetic flux, and their derivatives. Considering the multiscale nature of the tilt instability, we study the effect of domain resolution in the tilt instability problem. We use a finite element discretization on unstructured meshes and an implicit scheme. We use the PETSc library with index sets for parallelization. To solve the nonlinear MHD problem, 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 program on a cluster.

1 MHD and its streamfunction approach

Magnetohydrodynamics (MHD) is the fluid dynamics of conducting fluid or plasma, coupled with Maxwell's equations. The fluid motion induces currents, which produce Lorentz body forces 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. In this study, we consider finite element methods on an unstructured mesh for two-dimensional, incompressible MHD, using a streamfunction approach to enforce the divergence-free condition on magnetic and velocity fields and an implicit time difference scheme to allow much lager time steps. Strauss and Longcope [SL98] applied adaptive finite element method with explicit time difference scheme to this problem.

The incompressible MHD equations are:

$$\begin{array}{ll} \frac{\partial}{\partial t}\mathbf{B} = \nabla \times (\mathbf{v} \times \mathbf{B}), & \frac{\partial}{\partial t}\mathbf{v} = -\mathbf{v} \cdot \nabla \mathbf{v} + (\nabla \times \mathbf{B}) \times \mathbf{B} + \mu \nabla^2 \mathbf{v}, \\ \nabla \cdot \mathbf{v} = 0, & \nabla \cdot \mathbf{B} = 0, \end{array} \qquad (1)$$

where $\mathbf{B}$ is the magnetic field, $\mathbf{v}$ is the velocity, and μ is the viscosity. To enforce incompressibility, it is common to introduce stream functions: $\mathbf{v} = \left(\frac{\partial\phi}{\partial y}, -\frac{\partial\phi}{\partial x}\right), \quad \mathbf{B} = \left(\frac{\partial\psi}{\partial y}, -\frac{\partial\psi}{\partial x}\right)$. By reformulation for symmetric treatment of the fields, in the sense that the source function Ω and C are time advanced, and the potentials ϕ and ψ are obtained at each time step by solving Poisson equations, we obtain

$$\begin{array}{l} \frac{\partial}{\partial t}\Omega + [\Omega,\phi] = [C,\psi] + \mu\nabla^2\Omega, \\ \frac{\partial}{\partial t}C + [C,\phi] = [\Omega,\psi] + 2\left[\frac{\partial\phi}{\partial x}, \frac{\partial\psi}{\partial x}\right] + 2\left[\frac{\partial\phi}{\partial y}, \frac{\partial\psi}{\partial y}\right] \\ \nabla^2\phi = \Omega, \qquad \nabla^2\psi = C, \end{array} \tag{2}$$

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

To solve (2), we have to compute the partial derivatives of potentials. These partial derivatives can be obtained by solutions of linear problems. To do this, we have to introduce four auxiliary variables. It requires the solution of eight equations at each step.

We use the velocity $\mathbf{v}$ and the magnetic field $\mathbf{B}$ to reduce the number of equations to solve (2). To this aim, we put $\mathbf{v} = (v_1, v_2) = \left(\frac{\partial\phi}{\partial y}, -\frac{\partial\phi}{\partial x}\right), \quad \mathbf{B} = (B_1, B_2) = \left(\frac{\partial\psi}{\partial y}, -\frac{\partial\psi}{\partial x}\right)$ in equation (2) and get the following system:

$$\begin{array}{l} \frac{\partial\Omega}{\partial t} + (v_1,v_2)\cdot\nabla\Omega = (B_1,B_2)\cdot\nabla C + \mu\nabla^2\Omega, \\ \frac{\partial C}{\partial t} + (v_1,v_2)\cdot\nabla C = (B_1,B_2)\cdot\nabla\Omega + 2([v_1,B_1] + [v_2,B_2]), \\ -\nabla^2 v_1 = -\frac{\partial\Omega}{\partial y}, \quad -\nabla^2 v_2 = \frac{\partial\Omega}{\partial x}, \quad -\nabla^2 B_1 = -\frac{\partial C}{\partial y}, \quad -\nabla^2 B_2 = \frac{\partial C}{\partial x}, \\ \nabla^2\phi = \Omega, \qquad \nabla^2\psi = C. \end{array} \tag{3}$$

In the eight equations in (3), the last two equations for potentials don't need to be solved to get the solutions in each time step. If the potentials are needed at a specific time, they are obtained by solving the last two equations in (3). To solve the Poisson's equations for $\mathbf{v}$ and $\mathbf{B}$ in (3), we have to impose boundary conditions which are compatible to boundary conditions of ϕ, ψ, Ω, and C.

2 Finite discretization

To solve (3), we use the first-order backward difference (Euler) derivative scheme leading to an implicit scheme which removes the numerically imposed time-step constraint, allowing much larger time steps. This approach is first order accurate in time and is chosen merely for convenience.

Let H^1 denote $H^1(K)$, $H^{1,A}$ denote the subset of $H^1(K)$ which satisfy the boundary condition of A, and $H^{1,A'}$ denote he subspace of $H^1(K)$ whose elements have zero values on the Dirichlet boundary of $A = \Omega, v_1, v_2, B_1, B_2$. Multiply the test functions and integrate by parts in each equation and use the appropriate boundary conditions, to derive the variational form of (3) as follows: 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

$$\mathbf{F}^n(\mathbf{X}, \mathbf{Y}) = 0 \tag{4}$$

for all $\mathbf{Y} = (u, w, p_1, p_2, q_1, q_2) \in H^{1,\Omega'} \times H^1 \times H^{1,v_1'} \times H^{1,v_2'} \times H^{1,B_1'} \times H^{1,B_2'}$, where $\mathbf{F}^n = (F_1^n, F_2^n, F_3, F_4, F_5, F_6)^T$,

$$\begin{aligned}
F_1^n(\mathbf{X}, \mathbf{Y}) &= M_t^n(\Omega, u) + (\mathbf{v} \cdot \nabla\Omega, u) + \mu a(\Omega, u) - (\mathbf{B} \cdot \nabla C, u),\\
F_2^n(\mathbf{X}, \mathbf{Y}) &= M_t^n(C, w) + (\mathbf{v} \cdot \nabla C, w) - (\mathbf{B} \cdot \nabla \Omega^n, w) - 2P(\mathbf{v}, \mathbf{B}, w),\\
F_3(\mathbf{X}, \mathbf{Y}) &= a(v_1, p_1) + \left(\tfrac{\partial\Omega}{\partial y}, p_1\right), \quad F_4(\mathbf{X}, \mathbf{Y}) = a(v_2, p_2) - \left(\tfrac{\partial\Omega}{\partial x}, p_2\right),\\
F_5(\mathbf{X}, \mathbf{Y}) &= a(B_1, q_1) + \left(\tfrac{\partial C}{\partial y}, q_1\right), \quad F_6(\mathbf{X}, \mathbf{Y}) = a(B_2, q_2) - \left(\tfrac{\partial C}{\partial x}, q_2\right),
\end{aligned}$$

$$(u, w) = \int_K uw dx, \quad M_t^n(u, w) = \frac{1}{\Delta t}(u - u^n, w), \quad a(u, w) = \int_K \nabla u \cdot \nabla w dx,$$

$$P((u_1, u_2), (v_1, v_2), w) = \int_K [u_1, v_1] w dx + \int_K [u_2, v_2] w dx.$$

Let $\mathcal{K}_h$ be a given triangulation of domain K with the maximum diameter h of the element triangles. Let V_h be the continuous piecewise linear finite element space. Let V_h^A, $A = \Omega, v_1, v_2, B_1, B_2$, be the subsets of V_h which satisfy the boundary conditions of A on every boundary point of $\mathcal{K}_h$ and $V_h^{A'}$ be subspaces of V_h and $H^{1,A'}$. Then we can write the discretized MHD problems as follows: For each n, find the solutions $\mathbf{X}_h^n = (\Omega_h^n, C_h^n, v_{1,h}^n, v_{2,h}^n, B_{1,h}^n, B_{2,h}^n) \in V_h^\Omega \times V_h \times V_h^{v_1} \times V_h^{v_2} \times V_h^{B_1} \times V_h^{B_2}$ on each discretized times n which satisfy

$$F^n(\mathbf{X}_h^n, \mathbf{Y}_h) = 0 \tag{5}$$

for all $\mathbf{Y}_h \in V_h^{\Omega'} \times V_h \times V_h^{v_1'} \times V_h^{v_2'} \times V_h^{B_1'} \times V_h^{B_2'}$.

3 Nonlinear and linear solvers

(5) 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 linear with respect to one variable. Specifically, the last four equations are linear equations and Poisson problems. From the above observations, we naturally consider a nonlinear Gauss-Seidel iterative solvers (GS1) which solve linear each equation on one variable in (5) in consecutive order with recent approximate solutions.

Poisson solvers are well developed and the first two equations are time dependent problems. From this observation, we consider another nonlinear Gauss-Seidel iterative solvers (GS2) which solve first two equations of (5) as one equation and solve four Poisson equations.

Nonlinear Gauss-Seidel iterative method doesn't guarantee convergence, but converges well in many cases, especially for small time step sizes in time dependent problems.

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

In all three nonlinear solvers, we need to solve linear problems. Krylov iterative techniques are suited because they can be preconditioned for efficiency. Among the various Krylov methods, GMRES (Generalized Minimal RESiduals) is selected because it guarantees convergence with 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 principal 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

$$J_k \delta x_k = -F(x_k) \tag{6}$$

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 because this can be implemented easily. This is straightforward to see when considering the equivalent linear system:

$$P_k^{-1} J_k \delta x_k = -P_k^{-1} F(x_k). \tag{7}$$

Notice that the system in equation (7) is equivalent to the original system (6) for any nonsingular operator P_k^{-1}. Thus, the choice of P_k^{-1} does not affect the accuracy of the final solution, but crucially determines the rate of convergence of GMRES, and hence the efficiency of the algorithm.

In this study, we use multigrid which is well known as a successful preconditioner, as well as a scalable solver in unaccelerated form, for many problems. We consider the symmetrized diagonal term of Jacobian as a reduced system, i.e.,

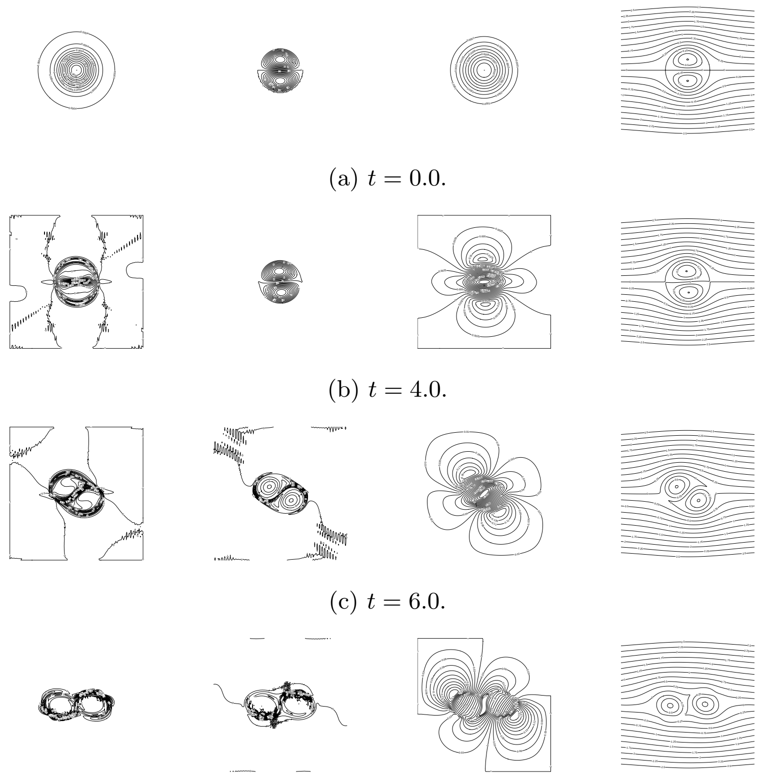
$$J_{S,k} = \frac{1}{2}\left(J_{R,k} + J_{R,k}^T\right), \tag{8}$$

where $J_{R,k}$ is a block diagonal matrix. The reduced system $J_{S,k}$ may be less efficient than $J_{R,k}$ but more numerically stable because it is symmetric and nonsingular.

To implement the finite element solver for two-dimensional, incompressible MHD on parallel machines, we use PETSc library which is well developed for nonlinear PDE problems and easily implements a multigrid preconditioner with GMRES. We use PETSc's index sets for parallelization of our unstructured finite element discretization.

4 Numerical experiments : Tilt Instability

We consider the initial equilibrium state as $\psi = \begin{cases} [2/kJ_0(k)]J_1(kr)\frac{y}{r}, & r < 1, \\ (1/r - r)\frac{y}{r}, & r > 1, \end{cases}$ where J_n is the Bessel function of order n, k is any constant that satisfies $J_1(k) = 0$, and $r = \sqrt{x^2 + y^2}$.



(a) $t = 0.0$.

(b) $t = 4.0$.

(c) $t = 6.0$.

(d) $t = 7.0$.

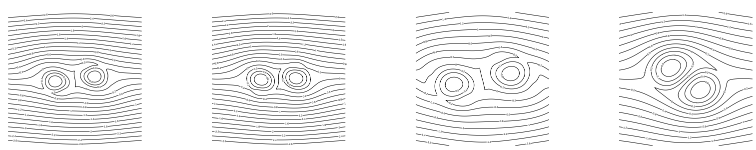
Fig. 1. Contours of Ω, C, ϕ, and ψ at time $t = 0.0, 4.0, 6.0, 7.0$.

In our numerical experiments, we solve on the finite square domain $K = [-R, R] \times [-R, R]$ with the initial condition of the tilt instability problem from the above initial equilibrium and perturbation of ϕ (originating from perturbations of velocity) such that

$$\Omega(0) = 0.0, \qquad C(0) = \begin{cases} 19.0272743 J_1(kr)y/r & \text{if } r < 1 \\ 0.0 & \text{if } r > 1 \end{cases},$$
$$\phi(0) = 10^{-3}e^{-(x^2+y^2)}, \ \psi(0) = \begin{cases} -1.295961618 J_1(kr)y/r & \text{if } r < 1 \\ -(\frac{1}{r} - r)y/r & \text{if } r > 1, \end{cases}$$

where $k = 3.831705970$ and with Dirichlet boundary conditions $\Omega(x,y,t) = 0.0$, $\phi(x,y,t) = 0.0$, and $\psi(x,y,t) = y - \frac{y}{x^2+y^2}$ and Neumann boundary condition for C, i.e., $\frac{\partial C}{\partial n}(x,y,t) = 0.0$. The initial and boundary condition for velocity $\mathbf{v}$ and magnetic field $\mathbf{B}$ are derived from the initial and boundary condition of Ω, C, ϕ, and ψ. Numerical simulation results are illustrated in Fig. 1.

The tilt instability problem is defined on unbounded domain. To investigate the effect of size of domains, we simulate two methods, one uses ϕ, ψ and its derivatives (SL) and the other uses $\mathbf{v}$ and $\mathbf{B}$ (K) on the square domains with $R = 2$ and $R = 3$. These numerical simulation results are reported in Fig. 2, the contours of ψ at $t = 6.0$. The average growth rate γ of kinetic energy is shown in Table 1. These numerical simulation results show that the solutions of two formulations are closer when the domain is enlarged, with the previous approach converging from above and new approach converging from below.



(a) $R = 3$ (SL). (b) $R = 3$(K). (c) $R = 2$ (SL). (d) $R = 2$(K).

Fig. 2. Contours of ψ at $T = 7.0$.

Table 1. Average growth rate γ of kinetic energy from $t = 0.0$ to $t = 6.0$

previous, $R = 2$	previous, $R = 3$	new, $R = 2$	new, $R = 3$
2.167	2.152	1.744	2.102

From here, we consider the convergence behaviors of several nonlinear and linear solvers as a function of time step sizes. In Table 2, we report the number of nonlinear iterations of nonlinear solvers according to time step sizes for the fixed staring time and fixed mesh level 5. We choose $t = 0.0$ and $t = 6.0$ as the starting time because many simulations have trouble at start up time and the magnitudes of the velocity (v_1, v_2) and magnetic field (B_1, B_2) increase with time. These numerical results show that GS2 and Newton method are more nonlinearly robust than GS1.

To investigate another convergence behavior, we report the average number of linear iterations in one time step according to preconditioners in Table 3. Numerical results show that multigrid preconditioner applying on symmetrized reduced system is robust at $t = 0.0$ and $t = 6.0$, but multigrid

Table 2. The average number of nonlinear iterations of one time step according to time step sizes dt.

dt	GS1		GS2		NM	
	$t=0$	$t=6$	$t=0$	$t=6$	$t=0$	$t=6$
0.0005	3	4	3	4	2	4
0.001	4	4	3	4	3	4
0.002	5	5	3	3	5	4(5)
0.005	12	8	4	6	3	5
0.01	*	*	4	8	4	7
0.02	*	*	6	13	5	11

applied to the reduced system is robust only at $t = 0.0$, very similar to the symmetrized case, because the values of velocity are small at $t = 0.0$. These results show that we have to use multigrid preconditioner applied to the symmetrized reduced system to get good convergence.

Table 3. The average number of linear iterations in one time step according to time step sizes

dt	GS2(R)		GS2(S)		NM(R)		NM(S)	
	$t=0$	$t=6$	$t=0$	$t=6$	$t=0$	$t=6$	$t=0$	$t=6$
0.0005	4.3	4	4.3	4	5	5	5	5
0.001	5.3	4.5	5.3	5	6	5	6	5
0.002	6.6	5	6.6	5.2	7	6.8	7	6.25
0.005	11	7.8	11	8.8	12	*	12	10
0.01	18	*	18	15.2	18.5	*	18.5	17
0.02	28.8	*	28.8	27.3	31.6	*	31.6	33.3

In Table 4, we report the average number of nonlinear and linear iterations from $t = 0.0$ to $t = 0.05$ with $dt = 0.005$ according to the levels. These results show that two numerical method GS2(S) and Newton method (S) have very similar behaviors.

Table 4. Average number of iterations according to the number of level.

Solvers	GS2(S)		NM(S)	
level	nonlinear	linear	nonlinear	linear
4	4	7.9	3	8
5	3.1	11.2	3	11.7
6	3	16.1	3	16.4
7	3.4	19.1	3.4	20.1

In Table 5, we report the solution times of one time step according to level and number of processors on the cluster machine BGC (the Brookhaven

Galaxy Cluster) at BNL. We run the program on the same speed (696 MHz) CPU's. This table shows that Newton's method have a better scalabilty GS2.

Table 5. Average solving time of one time step (linear system) according to level and number of processors at $t = 0.0$ and $dt = 0.005$

level	# CPU	GS2(S)	NM(S)
4	1	13.7 (2.39)	12.3 (2.02)
	2	13.3 (2.74)	7.76 (1.40)
5	2	42.5 (11.3)	33.5 (6.62)
	4	29.4 (7.79)	21.0 (4.23)
	8	38.6 (11.12)	19.9 (4.89)
6	8	120.5 (35.9)	59.9 (14.4)
	16	64.5 (19.3)	39.0 (9.65)
	32	118.1 (36.95)	61.6 (17.8)
7	32	226.3(61.8)	142.9 (36.2)

5 Conclusions

We study the new streamfunction approach method for two-dimensional, incompressible Magnetohydrodynamics with finite element method and tilt instability example. We show that nonlinear Gauss-Seidel (GS2) and Newton method have similar numerical behaviors and have to employ multigrid preconditioner on the symmetrized reduced system for good linear convergence.

Acknowledgement

The author would like to thank D. E. Keyes of Columbia University for his valuable advice in the preparation of this work. This work was supported by the United States Department of Energy under contract No. DE-AC02-98CH10886.

References

[SL98] Strauss, H. R. and Longcope, D. W.: An Adaptive Finite Element Method for Magnetohydrodynamics., *Journal of Computational Physics*, 147:318–336, 1998.