Daniel Brauss

Francis Marion University

Research: Numerical Solution of Velocity-Current Magnetohydrodynamic (MHD) Equations

Methods: Finite Element Method

Resources: Supercomputing

Libraries: deal.II

Codes: Parallel with MPI

Goals: Further parallelization and optimization

Daniel Brauss

Francis Marion University

Implementation of a Least Squares, Conjugate Gradient, FEM Solver For Velocity-Current MHD Equations

K. Daniel Brauss

A. J. Meir

Introduction

Magnetohydrodynamics (MHD) studies the movement of conducting fluids under the presence of electromagnetic fields. MHD phenomena range from industrial processes like aluminum casting and crystal growth to studying of the formation of stars in astrophysics. We recall the *velocity –current MHD equations* studied by A.J. Meir and Paul G. Schmidt

$$\eta\Delta\boldsymbol{u} + \rho(\boldsymbol{u}\cdot\nabla)\boldsymbol{u} + \nabla p - \boldsymbol{J}\times\boldsymbol{B} = \boldsymbol{F}$$

$$\sigma^{-1}\boldsymbol{J} + \nabla\phi - \boldsymbol{u}\times\boldsymbol{B} = \boldsymbol{E} \qquad \text{in } \Omega$$

$$\nabla\cdot\boldsymbol{u} = 0 \text{ and } \nabla\cdot\boldsymbol{J} = 0$$

with boundary conditions

$$\boldsymbol{u} = \boldsymbol{g} \quad \text{and} \quad \phi = k \qquad \text{on } \Gamma$$

and Biot-Savart law

$$\boldsymbol{B}(x) = \boldsymbol{B}_o(x) + \boldsymbol{B}(\boldsymbol{J})(x) = \boldsymbol{B}_{ext} - \frac{\mu}{4\pi}\int_{\mathbb{R}^3\setminus\Omega}\frac{x-y}{|x-y|^3}\times\boldsymbol{J}_{ext}(y)\,dy - \frac{\mu}{4\pi}\int_\Omega\frac{x-y}{|x-y|^3}\times\boldsymbol{J}(y)dy.$$

which can be used to eliminate the magnetic field as an unknown.

Approximation to Weak Formulation

We approximate the weak formulation to the equations above with the Galerkin Finite Element Method and finite element subspaces (see below) satisfying corresponding LBB conditions. Given $\boldsymbol{F}\in \boldsymbol{H}^{-1}(\Omega)$, $\boldsymbol{E}\in L^2(\Omega)$, we seek a $\mathbf{u}^h \in V_\Gamma^h \subset \boldsymbol{H}_\Gamma^1(\Omega)$, $p^h \in S^h \subset L^2(\Omega)$, $\mathbf{J}^h \in W^h \subset \boldsymbol{L}^2(\Omega)$, and $\phi^h \in T_\Gamma^h \subset H_\Gamma^1(\Omega)$ (where boundary conditions are approximated on Γ) such that

$$a\left((\boldsymbol{u}^h,\boldsymbol{J}^h),(\boldsymbol{v}^h,\boldsymbol{K}^h)\right) + c\left((\boldsymbol{u}^h,\boldsymbol{J}^h),(\boldsymbol{u}^h,\boldsymbol{J}^h),(\boldsymbol{v}^h,\boldsymbol{K}^h)\right) + b\left((\boldsymbol{v}^h,\boldsymbol{K}^h),(p^h,\phi^h)\right) = (\boldsymbol{F},\boldsymbol{v}^h) + (\boldsymbol{E},\boldsymbol{K}^h) \quad \boldsymbol{v}^h\in V_0^h \text{ and } \boldsymbol{K}^h\in W^h$$

$$b\left((\boldsymbol{u}^h,\boldsymbol{J}^h),(q^h,\psi^h)\right) = 0 \quad q^h\in S^h \text{ and } \psi^h\in T_0^h$$

where the parameter h is the size of cells making up the grid subdividng $\overline{\Omega}$ into elements and

$$a\big((\boldsymbol{u},\boldsymbol{J}),(\boldsymbol{v},\boldsymbol{K})\big) = \nu\int_\Omega \nabla\boldsymbol{u}:\nabla\boldsymbol{v}\,dx + \sigma^{-1}\int_\Omega \boldsymbol{J}\cdot\boldsymbol{K}\,dx \qquad \boldsymbol{u},\boldsymbol{v}\in H^1(\Omega),\ \boldsymbol{J},\boldsymbol{K}\in L^2(\Omega)$$

Algorithm

To solve the Least Squares problem, we implement the algorithm below that is based on work by Roland Glowinksi and colleagues.

Conjugate Gradient Algorithm

1. Initialization : $(\boldsymbol{u}^0,\boldsymbol{J}^0)\in V_0^h\times W^h$ is chosen
2. Solve for $(\boldsymbol{g}_1^0,\boldsymbol{g}_2^0)\in V_0^h\times W^h$ the Stokes-type system

$$a\left((\boldsymbol{g}_1^0,\boldsymbol{g}_2^0),(\boldsymbol{v},\boldsymbol{K})\right) + b\big((\boldsymbol{v},\boldsymbol{K}),(\sigma,\theta)\big) = \langle\mathfrak{I}'\big((\boldsymbol{u}^0,\boldsymbol{J}^0)\big),(\boldsymbol{v},\boldsymbol{K})\rangle \quad \forall(\boldsymbol{v},\boldsymbol{K})\in V_0^h\times W^h$$

$$b\left((\boldsymbol{g}_1^0,\boldsymbol{g}_2^0),(q,\psi)\right) = 0 \quad \forall(q,\psi)\in S^h\times T_0^h$$

3. Set $(\boldsymbol{w}_1^0,\boldsymbol{w}_2^0) = (\boldsymbol{g}_1^0,\boldsymbol{g}_2^0)$$
4. Begin <u>Loop</u>: *Given* $(\boldsymbol{u}^n,\boldsymbol{J}^n),(\boldsymbol{g}_1^n,\boldsymbol{g}_2^n)\neq(\mathbf{0},\mathbf{0}),(\boldsymbol{w}_1^n,\boldsymbol{w}_2^n)\neq(\mathbf{0},\mathbf{0})$
   a. Determine $\rho_n\in\mathbb{R}$ such that
   $$\mathfrak{I}\big((\boldsymbol{u}^n,\boldsymbol{J}^n)-\rho_n(\boldsymbol{w}_1^n,\boldsymbol{w}_2^n)\big) \le \mathfrak{I}\big((\boldsymbol{u}^n,\boldsymbol{J}^n)-\rho(\boldsymbol{w}_1^n,\boldsymbol{w}_2^n)\big)\ \forall\rho\in\mathbb{R}$$
   b. Set $(\boldsymbol{u}^{n+1},\boldsymbol{J}^{n+1}) = (\boldsymbol{u}^n,\boldsymbol{J}^n)-\rho_n(\boldsymbol{w}_1^n,\boldsymbol{w}_2^n)$
   c. Solve for $(\boldsymbol{g}_1^{n+1},\boldsymbol{g}_2^{n+1})$ the Stokes-type system
   $$a\left((\boldsymbol{g}_1^{n+1},\boldsymbol{g}_2^{n+1}),(\boldsymbol{v},\boldsymbol{K})\right) + b\big((\boldsymbol{v},\boldsymbol{K}),(\sigma,\theta)\big) = \langle\mathfrak{I}'\big((\boldsymbol{u}^{n+1},\boldsymbol{J}^{n+1})\big),(\boldsymbol{v},\boldsymbol{K})\rangle \quad \forall(\boldsymbol{v},\boldsymbol{K})\in V_0^h\times W^h$$
   $$b\left((\boldsymbol{g}_1^{n+1},\boldsymbol{g}_2^{n+1}),(q,\psi)\right) = 0 \quad \forall(q,\psi)\in S^h\times T_0^h$$
   d. If $\frac{(\boldsymbol{g}_1^{n+1},\boldsymbol{g}_2^{n+1})}{(\boldsymbol{g}_1^0,\boldsymbol{g}_2^0)} < \epsilon$ or $\frac{\mathfrak{I}\left((\boldsymbol{u}^{n+1}\boldsymbol{J}^{n+1})\right)}{\mathfrak{I}\left((\boldsymbol{u}^0\boldsymbol{J}^0)\right)} < \epsilon$
   Set $(\boldsymbol{u},\boldsymbol{J})$=$(\boldsymbol{u}^{n+1},\boldsymbol{J}^{n+1})$and Set $\boldsymbol{flag} = on$
   e. Else compute (Fletcher-Reeves or Polak-Ribiere)
   $$\gamma_n = \frac{\left((\boldsymbol{g}_1^{n+1},\boldsymbol{g}_2^{n+1}),(\boldsymbol{g}_1^{n+1},\boldsymbol{g}_2^{n+1})\right)}{\left((\boldsymbol{g}_1^n,\boldsymbol{g}_2^n),(\boldsymbol{g}_1^n,\boldsymbol{g}_2^n)\right)} \quad or \quad \gamma_n = \frac{\left((\boldsymbol{g}_1^{n+1},\boldsymbol{g}_2^{n+1})-(\boldsymbol{g}_1^n,\boldsymbol{g}_2^n),(\boldsymbol{g}_1^{n+1},\boldsymbol{g}_2^{n+1})\right)}{\left((\boldsymbol{g}_1^n,\boldsymbol{g}_2^n),(\boldsymbol{g}_1^n,\boldsymbol{g}_2^n)\right)}$$
   f. Set $(\boldsymbol{w}_1^{n+1},\boldsymbol{w}_2^{n+1}) = (\boldsymbol{g}_1^{n+1},\boldsymbol{g}_2^{n+1}) + \gamma_n(\boldsymbol{w}_1^n,\boldsymbol{w}_2^n)$
5. End <u>Loop</u> if $\boldsymbol{flag} = on$

The algorithm utilizes the formula

$$\langle\mathfrak{I}'\big((\boldsymbol{u}^n,\boldsymbol{J}^n)\big),(\boldsymbol{v},\boldsymbol{K})\rangle = a\big((\boldsymbol{v},\boldsymbol{K}),(\xi_1^n,\xi_2^n)\big) + c\big((\boldsymbol{u}^n,\boldsymbol{J}^n),(\boldsymbol{v},\boldsymbol{K}),(\xi_1^n,\xi_2^n)\big) + c\big((\boldsymbol{v},\boldsymbol{K}),(\boldsymbol{u}^n,\boldsymbol{J}^n),(\xi_1^n,\xi_2^n)\big)$$

The search for $\rho_n\in\mathbb{R}$ utilizes Newton's Method by minimizing the derivative of the polynomial function $j_n = j_n(\rho)$ given by

Discussion and Preliminary Results

The matrix systems arising in the conjugate gradient algorithm have the form of the Stokes equations. We utilize the finite element library deal.II. Deal.II is configured with the software libraries Trilinos and p4est for distributed HPC architectures. The deal.II developers provide well-documented literature and tutorials. To solve the Stokes-type systems, we use the conjugate gradient method with a preconditioner that is based on the theorem below.

Theorem 5.4. *If* u^h, p^h *and* J^h,ϕ^h *satisfy the respective uniform LBB conditions such that spectral equivalence of the Schur complements with corresponding mass matrices hold, then for the preconditioned system*

$$\begin{bmatrix} A_{(u)} & 0 & 0 & 0 \\ 0 & A_{(J)} & 0 & 0 \\ 0 & 0 & Q_{(p)} & 0 \\ 0 & 0 & 0 & Q_{(\nabla\phi)} \end{bmatrix}^{-1} \begin{bmatrix} A_{(u)} & 0 & B_{(p)}^t & 0 \\ 0 & A_{(J)} & 0 & B_{(\phi)}^t \\ B_{(p)} & 0 & 0 & 0 \\ 0 & B_{(\phi)} & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathrm{c}^{(u)} \\ \mathrm{c}^{(J)} \\ \mathrm{c}^{(p)} \\ \mathrm{c}^{(\phi)} \end{bmatrix} = \lambda \begin{bmatrix} \mathrm{c}^{(u)} \\ \mathrm{c}^{(J)} \\ \mathrm{c}^{(p)} \\ \mathrm{c}^{(\phi)} \end{bmatrix}$$

all negative eigenvalues satisfy

$$-1 \le \lambda \le \frac{1}{2} - \frac{\sqrt{1+4\min(\beta_1,\beta_2)}}{2},$$

and all positive eigenvalues satisfy

$$1 \le \lambda \le \frac{1}{2} + \frac{\sqrt{5}}{2}.$$

Further, upon replacing the mass matrices by their corresponding Schur complements, the system has exactly three eigevalues $\lambda = 1, \frac{1}{2}\pm\frac{\sqrt{5}}{2}$.

Below are preliminary results applying the algorithm to the Navier-Stokes equations for the Lid Driven Cavity problem.