Preconditioned JFNK Method for Resistive MHD

Ravi Samtaney

Daniel Reynolds

Carol S. Woodward

FDM3D Workshop

March 19-21, 2007

PPPL, Princeton

Selective Review of Related Work

- Implicit treatment of fast compressive wave
  - *Original idea proposed by Harned & Kerner (JCP 1985)*
  - *Basic idea: Subtract a term which mimics the fast wave behavior from both sides of the $v_\perp$ equation; treat one side explicitly and the other implicitly*
  - *First order in time*
- Implicit treatment of Alfvén wave
  - *Harned & Schnack (JCP 1986)*
  - *Similar to the fast compressive treatment with extra heuristics (e.g. set cross terms of the operator to zero)*
  - *First order in time*
- Newton-Krylov Approaches
  - *Chacon et. al : "Parabolization + Schur complement + Multigrid"*
    - *Works for resistive and two-fluid MHD*
  - *Reynolds et. al: SUNDIALS (cvode, KINSOL)*
    - *Preconditioner: Operator + Directional splitting, Local wave-structure decomposition*

Nonlinearly Implicit : Introduction to Newton-Krylov

- The solution at the next time level to the entire system of equations is expressed as the solution to the following nonlinear equation

$$\mathcal{F}(U^{n+1}) = 0$$

$$\mathcal{F}(U^{n+1}) = U^{n+1} - U^n + (1-\theta)R(U^{n+1}) + \theta R(U^n) = 0$$

The number of unknowns is $8N^2$ for an NxN mesh

- This is solved using Newton's method

$$\delta U^k = -\left[\left(\frac{\partial \mathcal{F}}{\partial U}\right)^{n+1,k}\right]^{-1} \mathcal{F}$$

where $J(U^{n+1,k}) \equiv \left(\frac{\partial \mathcal{F}}{\partial U}\right)^{n+1,k}$ is the Jacobian; and $\delta U^k \equiv U^{n+1,k+1} - U^{n+1,k}$

The size of the Jacobian matrix is $64N^4$

- The linear system at each Newton iteration is solved with a Krylov method in which an approximation to the linear system $J\,\delta U = -\mathfrak{F}$ is obtained by iteratively building a Krylov subspace of dimension m

$$\mathcal{K}(r_0, J) = span\{r_0, Jr_0, J^2 r_0, \cdots, J^{m-1} r_0\}$$

Nonlinearly Implicit : Introduction to Newton-Krylov

- Commonly used Krylov methods which can handle asymmetric matrices
  - *GMRES (Generalized Minimum Residual)*
    - *Long-recurrence Arnoldi orthogonalization method*
    - *Robust, guaranteed convergence, but heavy on memory requirement*
  - *BiCGStab (Bi-conjugate Gradient Stabilized)*
    - *Short-recurrence Lanzcos biorthogonalization procedure*
    - *Residual not guaranteed to decrease monotonically, but less memory requirement*
- Steps in a Newton-Krylov method
  1. *Guess the solution $U^{n+1,0}$ (=U^n)*
  2. *For each Newton iteration k*
     1. *Using a Krylov Method solve for δU^k*
        Solve $J \delta U^k = - F(U^{n+1,k})$ until $\| J \delta U^k + F(U^{n+1,k}) \| < ltol$
  3. *Update the Newton iterate: $U^{n+1,k+1} = U^{n+1,k} + \lambda \delta U^k$*
  4. *Check for convergence $\|F(U^{n+1,k+1})\| < ftol$*
- Newton method converges quadratically. In practice, we use "Inexact" Newton which can exhibit linear or superlinear convergence
- **Jacobian-Free Newton-Krylov:** Krylov methods require only matrix-vector products to build up the Krylov subspace, i.e., only $J \delta U$ is required. Thus, the entire method can be built from evaluations of the nonlinear function F(U)

$$\boxed{J(U^k)\delta U^k \approx \frac{\mathcal{F}(U^{n+1,k} + \sigma \delta U^k) - \mathcal{F}(U^{n+1,k})}{\sigma}}$$

JFNK: Resistive MHD

- Reynolds et al. (JCP 2006) developed a fully implicit MHD method for the fully conservative form of the resistive MHD equations. The nonlinear function was expressed as a a high-order BDF method

$$g(\mathbf{U}^n) \equiv \mathbf{U}^n - \Delta t_n \beta_{n,0} f(\mathbf{U}^n) - \sum_{i=1}^{q_n} \left[\alpha_{n,i} \mathbf{U}^{n-i} + \Delta t_n \beta_{n,i} f(\mathbf{U}^{n-i}) \right]$$

$$\text{where } f(\mathbf{U}) \equiv \nabla \cdot (\mathbf{F}_v(\mathbf{U}) - \mathbf{F}_h(\mathbf{U}))$$

  In this method, $\alpha_{n,i}$ and $\beta_{n,i}$ are fixed parameters for a given method order q_n. The method is stable for any Δt_n for $q_n=\{1,2\}$

- The divergence of fluxes is discretized as

$$\left(\frac{\partial f}{\partial x}\right)_{i,j,k} = \frac{\tilde{f}_{i+\frac{1}{2},j,k} - \tilde{f}_{i-\frac{1}{2},j,k}}{\Delta x}$$

- The numerical fluxes are computed as

$$\tilde{f}_{i+\frac{1}{2},j,k} = \sum_{\nu=-m}^{n} a_\nu \, f_{i+\nu,j,k}$$

  The coefficients a_ν are chosen based on the order of the method. For 4-th order finite difference method: m=1, n=2, $a_{-1}=a_2=1/12$, $a_0=a_1=7/12$

- Resulting code was conservative, and preserved the solenoidal property of B

Main Idea of Hyperbolic Preconditioner

- Ideal MHD is a hyperbolic system of PDEs
- Linearization about previous time step (or Newton iterate)
- Decomposition of systems of coupled PDEs to decoupled systems of equations governing linear wave propagation
  - *Local decomposition into characteristics*
  - *Riemann invariants propagate along characteristics*
  - *Claim: If the fastest waves in the system are parasitical but dynamically insignificant then solving for the fastest wave may be an effective preconditioning method*
  - *Solve for the fastest waves; and reconstruct the solution from the Riemann invariants wherein only the Riemann invariants associated with the fastest waves are updated*

JFNK: Resistive MHD - Preconditioner

- Instead of solving J δ U = -g solve (J P^{-1}) (P δ U) = -g, i.e., right preconditioning is employed
- The preconditioner is split into a hyperbolic and a diffusive component

$$P^{-1} = P_h^{-1} P_d^{-1} = J(\mathbf{U})^{-1} + \mathcal{O}(\Delta t^2)$$

- Denoting by (.) the location of the linear operator action, the ideal MHD Jacobian is

$$\begin{aligned}
J_h(\mathbf{U}) &= I + \overline{\gamma}\left[J_x \partial_x(\cdot) + J_y \partial_y(\cdot) + J_z \partial_z(\cdot)\right] \\
&= I + \overline{\gamma}\left[J_x L_x^{-1} L_x \partial_x(\cdot) + J_y L_y^{-1} L_y \partial_y(\cdot) + J_z L_z^{-1} L_z \partial_z(\cdot)\right] \\
&= I + \overline{\gamma}\left[J_x L_x^{-1} \partial_x \left(L_x(\cdot)\right) - J_x L_x^{-1} \partial_x(L_x)(\cdot)\right. \\
&\qquad + J_y L_y^{-1} \partial_y \left(L_y(\cdot)\right) - J_y L_y^{-1} \partial_y(L_y)(\cdot) \\
&\qquad \left. + J_z L_z^{-1} \partial_z \left(L_z(\cdot)\right) - J_z L_z^{-1} \partial_z(L_z)(\cdot)\right]
\end{aligned}$$

where J_x is the Jacboian of the hyperbolic flux in the x-direction. L_x is the spatially local left eigenvector matrix for J_x. J_y, L_y, J_z, and L_z are similarly defined

JFNK: Resistive MHD - Preconditioner

- Directional splitting is employed to further approximate the preconditioner

$$
\begin{aligned}
P_h = &\left[I + \overline{\gamma} J_x L_x^{-1} \partial_x (L_x(\cdot))\right] \left[I + \overline{\gamma} J_y L_y^{-1} \partial_y (L_y(\cdot))\right] \left[I + \overline{\gamma} J_z L_z^{-1} \partial_z (L_z(\cdot))\right] \\
&\left[I - \overline{\gamma}\left(J_x L_x^{-1} \partial_x (L_x) + J_y L_y^{-1} \partial_y (L_y) + J_z L_z^{-1} \partial_z (L_z)\right)\right] \\
= &\, P_x\, P_y\, P_z\, P_{\text{corr}}.
\end{aligned}
$$

- Decoupling into 1D wave equations along characteristics

$$L_i(x) J_i(x) = \Lambda_i(x) L_i(x), \quad \Lambda_i = \mathrm{Diag}(\lambda^1, \ldots, \lambda^8)$$

$$L_i \left[I + \overline{\gamma} J_i L_i^{-1} \partial_i (L_i(\cdot))\right] \xi = L_i \beta \quad \Leftrightarrow \quad \zeta + \overline{\gamma}\, \Lambda_i\, \partial_i \zeta = \chi,$$

where $\zeta = L_i \xi$ and $\chi = L_i \beta$

- Only the fastest stiffness inducing waves need to be solved. Furthermore, accuracy may be sacrificed because this is done in the context of the preconditioner.
- Thus along each direction, we get a system of linear wave equations. For each wave family, we now get a sequence of tridiagonal linear systems which can be efficiently solved. In parallel we use the method proposed by Arbenz & Gander (1994)

PPPL PRINCETON PLASMA PHYSICS LABORATORY

JFNK: Resistive MHD - Preconditioner

- For spatially varying J(U) a correction solve is involved

$$\begin{aligned} P_{\text{corr}} &= I - \overline{\gamma}\left[J_x L_x^{-1}\partial_x\left(L_x\right) + J_y L_y^{-1}\partial_y\left(L_y\right) + J_z L_z^{-1}\partial_z\left(L_z\right)\right] \\ &= I - \overline{\gamma}\left[L_x^{-1}\Lambda_x\partial_x\left(L_x\right) + L_y^{-1}\Lambda_y\partial_y\left(L_y\right) + L_z^{-1}\Lambda_z\partial_z\left(L_z\right)\right] \end{aligned}$$

- Since this has no spatial couplings, the resulting local block systems may be solved easily by precomputing the 8x8 block matrices P_{corr} at each location coupled with a LU factorization

JFNK: Resistive MHD - Preconditioner

- Diffusion Preconditioner P_d: This solves the subsystem $\partial_t \mathbf{U} - \nabla \cdot \mathbf{F}_v = 0$

$$P_d = J_v(\mathbf{U}) = I - \overline{\gamma}\frac{\partial}{\partial \mathbf{U}}\left(\nabla \cdot \mathbf{F}_v\right)$$

$$= \begin{bmatrix} I & 0 & 0 & 0 \\ 0 & I - \overline{\gamma} D_{\rho \mathbf{v}} & 0 & 0 \\ 0 & 0 & I - \overline{\gamma} D_{\mathbf{B}} & 0 \\ -\overline{\gamma} L_\rho & -\overline{\gamma} L_{\rho \mathbf{v}} & -\overline{\gamma} L_{\mathbf{B}} & I - \overline{\gamma} D_e \end{bmatrix}$$

- To solve $P_d\, y = b$ for $y = [\, y_\rho, y_{\rho \mathbf{v}}, y_{\mathbf{B}}, y_e]^T$
    1. Update $y_\rho = b_\rho$
    2. Solve $(I - \overline{\gamma} D_{\rho \mathbf{v}})\, y_{\rho \mathbf{v}} = b_{\rho \mathbf{v}}$ for $y_{\rho \mathbf{v}}$
    3. Solve $(I - \overline{\gamma} D_{\mathbf{B}})\, y_{\mathbf{B}} = b_{\mathbf{B}}$ for $y_{\mathbf{B}}$
    4. Update $\tilde{b}_e = b_e + \overline{\gamma}\,(L_\rho\, y_\rho + L_{\rho \mathbf{v}}\, y_{\rho \mathbf{v}} + L_{\mathbf{B}}\, y_{\mathbf{B}})$
    5. Solve $(I - \overline{\gamma} D_e)\, y_e = \tilde{b}_e$ for y_e.

Steps 2,3 and 5 are solved using a geometric multigrid approach. Step 4 may be approximated with finite differences instead of constructing and multiplying by individual submatrices

$$L_\rho\, y_\rho + L_{\rho \mathbf{v}}\, y_{\rho \mathbf{v}} + L_{\mathbf{B}}\, y_{\mathbf{B}} = \frac{1}{\sigma}\left[\nabla \cdot \mathbf{F}_v(U + \sigma W) - \nabla \cdot \mathbf{F}_v(\mathbf{U})\right]_e + O(\sigma),$$

where $W = [y_\rho, y_{\rho \mathbf{v}}, y_{\mathbf{B}}, 0]^T$

PPPL PRINCETON PLASMA PHYSICS LABORATORY

Verification Test: Linear Wave Propagation

- Ideal MHD test: Linear waves propagated to t=50
- Slow wave propagating obliquely to the mesh

Mesh	C	CPU[N]	CPU[BT]	CPU[FW]	Krylov[N]	Krylov[BT]	Krylov[FW]
64^2	50	14.75	52.31	17.31	620	50	50
	100	26.27	78.29	15.59	1226	111	50
	500	31.10	593.19	285.00	1531	1283	5146
128^2	50	56.17	227.00	64.12	661	50	50
	100	100.46	364.89	64.87	1254	120	50
	500	422.11	1941.88	599.23	4729	927	2482
256^2	50	307.12	873.00	278.93	618	50	56
	100	661.75	1333.38	274.23	1409	113	50
	500	2951.58	7692.26	1880.34	6209	966	1701
512^2	50	1285.05	3719.43	991.43	608	50	50
	100	2765.79	6278.70	1000.86	1265	133	50
	500	14791.33	35547.76	1009.03	6444	1055	56

Summary

- Verification tests of implicit (w/o preconditioning)
  - *Linear wave propagation*
  - *GEM reconnection*
  - *Pellet model problem*
  - *In all tests, the implicit code agreed with the explicit one*

- Preconditioners are developed recognizing that ideal MHD is a hyperbolic system
  - *Operator + Directional splitting*
  - *Along each direction, the local wave structure was exploited to solve for the fastest waves*
  - *Diffusion part of the equations preconditioned using multi-grid*

- For the chosen tests (wave propagation, KH) our preconditioned JFNK approach works well as the problem size gets larger and the time step is ~(10-100) larger than the explicit CFL constrained time step

- **Future Work**: Under the auspices of SciDAC-2, we will combine block-structured hierarchical adaptive mesh refinement (w/ Chombo) technique with fully implicit time stepping (JFNK)

PPPL PRINCETON PLASMA PHYSICS LABORATORY

Introduction to Newton-Krylov: Preconditoners

- Krylov methods can lead to slow convergence. This is especially true for MHD where the Jacobian is ill-conditioned. Preconditioners help alleviate the problem of slow convergence and are formulated as follows

$$
\begin{aligned}
(J(U^k)P^{-1})(P\delta U^k) &= -\mathcal{F}(U^{n+1,k}) \quad (Right), \\
(P^{-1}J(U^k))\delta U^k &= -P^{-1}\mathcal{F}(U^{n+1,k}) \quad (Left), \\
(P_L^{-1}J(U^k)P_R^{-1})(P_R\delta U^k) &= -P_L^{-1}\mathcal{F}(U^{n+1,k}) \quad (Both).
\end{aligned}
$$

- The basic idea of preconditioners is that the matrix JP^{-1} or $P^{-1}J$ is close to the identity matrix, i.e., P is a good approximation of J. Furthermore, to make preconditioning effective, P^{-1} should be computationally inexpensive to evaluate

- Two broad classes of preconditioners
  1. ***Algebraic***: *These are of the "black-box" type. Obtained from relatively inexpensive techniques such as incomplete LU, multi-grid etc. These require storage for the preconditioner.*
  2. ***Physics-based***: *These may be derived from semi-implicit methods, and pay close attention to the underlying physics in the problem. Furthermore, these can still operate in the "Jacobian-Free" mode.*