Princeton Plasma Physics Laboratory

PPPL- 4377

PPPL- 4377

Operator-Based Preconditioning of Stiff Hyperbolic Systems

Daniel R. Reynolds, Ravi Samtaney, And Carol S. Woodward

February, 2009

PPPL
PRINCETON PLASMA
PHYSICS LABORATORY

Prepared for the U.S. Department of Energy under Contract DE-AC02-76CH03073.

Princeton Plasma Physics Laboratory
Report Disclaimers

Full Legal Disclaimer

Trademark Disclaimer

PPPL Report Availability

Princeton Plasma Physics Laboratory:

http://www.pppl.gov/techreports.cfm

Office of Scientific and Technical Information (OSTI):

http://www.osti.gov/bridge

Related Links:

U.S. Department of Energy

Office of Scientific and Technical Information

Fusion Links

OPERATOR-BASED PRECONDITIONING OF STIFF HYPERBOLIC SYSTEMS*

DANIEL R. REYNOLDS†, RAVI SAMTANEY‡, AND CAROL S. WOODWARD§

Abstract. We introduce an operator-based scheme for preconditioning stiff components encountered in implicit methods for hyperbolic systems of partial differential equations posed on regular grids. The method is based on a directional splitting of the implicit operator, followed by a characteristic decomposition of the resulting directional parts. This approach allows for solution to any number of characteristic components, from the entire system to only the fastest, stiffness-inducing waves. We apply the preconditioning method to stiff hyperbolic systems arising in magnetohydrodynamics and gas dynamics. We then present numerical results showing that this preconditioning scheme works well on problems where the underlying stiffness results from the interaction of fast transient waves with slowly-evolving dynamics, scales well to large problem sizes and numbers of processors, and allows for additional customization based on the specific problems under study.

Key words. Implicit Methods, Preconditioning, Hyperbolic Systems

AMS subject classifications. 65F10, 65Y20, 35L60

1. Introduction.

The numerical solution of hyperbolic systems pervades the modeling of processes ranging from astrophysics and gas dynamics to magnetically-confined fusion. A key feature for many of these processes is a significant stiffness of the system as compared with the physical time scales of interest, e.g. plasma refueling in tokamaks, magnetic reconnection in the earth's magnetotail and core collapse supernova [17, 21, 22, 32, 36, 38, 39]. As an example, in core collapse supernova, one solves a hyperbolic hydrodynamic system including a shock moving from the supernova core outward. Stability of an explicit scheme applied to this problem requires a time step governed by the CFL condition of motion in the core ($\sim 10^{-7}$ s), but the slower dynamics of the shock are of interest ($\sim 10^{-4}$ s). Hence, one would like to run the simulation at the larger time step governed by accuracy considerations of modeling the shock rather than the stability constraint at the core [38]. Additionally, while there have been rather significant advances in more diffusive simulations of Navier-Stokes and resistive magnetohydrodynamic (MHD) flows through the development of modern multigrid methods, such approaches fail at high Reynolds and Lundquist numbers. In many of these and similar problems, such stiffness is induced through only a small number of the hyperbolic waves. Moreover, these stiff components may prohibit scaling of explicit simulations to the very high resolutions required in studying these processes. For these types of problems fully implicit solution techniques have recently been gaining favor in the community [10, 26, 28]. Of primary importance in the use of fully implicit methods for such problems is their ability to perform such simulations more efficiently and/or more scalably than their explicit counterparts.

Even for non-stiff systems, implicit methods may prove beneficial as computational simulations increase in scale, since explicit methods can succumb to poor parallel weak scaling. As detailed in [26], parallel execution time (E) for a CFL-constrained explicit calculation for a hyperbolic system is proportional to $TS^{1+\alpha/d}P^{\alpha/d}$, where

*This work was supported by the U.S. DOE under UCSD contract DE-FC02-06ER25785 (DRR), PPPL and Princeton university contract DE-AC020-76-CH03073, and under the auspices of the U.S. DOE by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

†Mathematics, Southern Methodist University, Dallas, TX 75275-0156 (reynolds@smu.edu).

‡Princeton Plasma Physics Lab, MS 26, PO Box 451, Princeton, NJ 08543, (samtaney@pppl.gov)

§Lawrence Livermore Nat'l Lab, Box 808, L-561, Livermore, CA 94551, (cswoodward@llnl.gov)

P is the number of processors, T is the simulation time interval, S is the problem size per processor, d gives the spatial resolution, and α is determined by the PDE under consideration (hyperbolic gives $\alpha = 1$, parabolic gives $\alpha = 2$). Thus even for 3D hyperbolic problems (best case scenario), explicit method execution times scale with processor count as $E \propto P^{1/3}$; 2D parabolic problems scale dramatically worse, with $E \propto P$. As a result of this CFL stability dependence, explicit methods have difficulty demonstrating ideal weak scaling ($E \propto 1$). While this may not seem severe for moderate problem sizes, target applications on modern petascale computational architectures encounter $P \approx 10^4 - 10^5$, at which level even explicit methods may not be utilized efficiently.

The application of implicit approaches usually requires the solution of a nonlinear system within each time step, typically solved with Newton's method. For large-scale applications simulated on modern petascale machines, these nonlinear systems are solved with a Newton-Krylov method [26]. As Krylov methods can stall, preconditioning is generally required for efficient solution of the linear systems.

Recent years have seen increased activity in the development and use of implicit solution approaches to hyperbolic systems as part of fluid dynamics applications (see [20, 28] and references therein). Generally, the emphasis of these works was to obtain steady state solutions for aerodynamics applications, altough more recently the emphasis has shifted to unsteady problems. An excellent example of the latter is the work of Mavriplis and coauthors [30, 41] who investigated nonlinear multigrid methods as solvers and preconditioners with a Newton or Newton-Krylov technique using unstructured meshes. Another recent example in the aerodynamic literature for implicit solutions to unsteady problems [5] combined nonlinear multigrid with a Newton-Krylov approach; using line Jacobi and diagonally dominant ADI approaches for preconditioning the linear system. Similarly, in the context of MHD, there has been significant recent work on the development of preconditioning approaches for fully implicit Newton-Krylov simulations. The notion of "physics-based" preconditioners has been championed by Chacón and co-workers [10, 8]. Their approach relies on an approximate Schur factorization of the linear Newton systems to "parabolize" the hyperbolic portions of the MHD system, making it amenable to multigrid techniques. The resulting preconditioners have been employed in 2D reduced MHD [10], and 3D resistive MHD [8]. Our work, described next, is distinguished from other works, in that our preconditioner directly attacks the sources of stiffness in the system through an appropriate decomposition of the hyperbolic operator. In addition, our preconditioner does not rely on a diffusive component (although one could be handled in our approach through an operator split preconditioner applied to the unsplit implicit system).

As evident in the literature, there are a variety of approaches for implicit solutions of hyperbolic systems arising in gas dynamics and MHD, with the dominant methods being nonlinear multgrid, Newton-Krylov or a combination of both. In this paper, we present a systematic approach based on a preconditioned Newton-Krylov method to implicitly solve general stiff hyperbolic systems such as those encountered in applications from fluid dynamics, fusion, and astrophysics. The idea behind our preconditioning approach is as follows: since the stiffness of the system results from a small number of fast waves, we derive an approximate decomposition of the system into its component waves and precondition only the stiffness-inducing parts. We base this approximate decomposition on an $O(\Delta t^2)$-accurate splitting of the system into its directional components, followed by a characteristic projection to decouple the

component waves. The result of this decoupling is a set of tridiagonal systems, that we solve in parallel using a divide-and-conquer approach similar to [1]. We note that within this strategy each characteristic equation is solved independently, enabling solution of only those components inducing stiffness to the fully implicit system, while leaving the slower components alone. Upon solution of these decoupled equations, we project the preconditioned solutions back into their original conserved variables, resulting in the approximate solution of the original linear Jacobian system. This preconditioning scheme is valid for any method of lines approach to hyperbolic systems that employs a linear single or multistep method for time integration, including those discussed in our earlier work using CVODE (high order, adaptive, fixed-leading-coefficient BDF) and KINSOL (fixed-step θ method) [23]. In this paper we focus on the fixed time-step Crank-Nicolson approach, in order to better elucidate the effect of preconditioning outside the adaptive time-stepping context. We note that extensions to the general hyperbolic system (e.g. reaction or diffusion terms), may be preconditioned independently from the stiff waves using an operator-split formulation.

This paper is organized as follows. In the next section we describe an implicit preconditioned Newton-Krylov approach to solving a general system of hyperbolic equations. In Section 3 we show how the implicit approach is applied to the systems of gas dynamics and magnetohydrodynamics. We present numerical results for the preconditioning strategy on a suite of test problems designed to investigate accuracy and scalability in Section 4 and give concluding remarks in Section 5.

2. Preconditioned Newton-Krylov Method for Hyperbolic PDEs. In this section, we describe the fully implicit formulation and Newton-Krylov solution approach for general hyperbolic conservation laws.

2.1. Discretization Approach. Consider the general hyperbolic sytem:

$$\partial_t U + \nabla \cdot \mathbf{F}(U) = 0, \tag{2.1}$$

where $U \equiv U(\mathbf{x}, t) : R^3 \times R \rightarrow R^n$ is a vector of n conserved quantities, and $\mathbf{F}(U) \equiv \{F(U), G(U), H(U)\}^T : R^n \rightarrow R^{3n}$ is a vector of fluxes.

In numerically approximating solutions of the system (2.1), we follow a method of lines approach for splitting the time and space dimensions. Under this methodology, the spatial components are discretized following a spatial semi-discretization involving a diagonal mass matrix (i.e. finite-differences or finite-volumes) on a regular spatial grid. Time discretization of (2.1) is performed using a standard linear multistep method that defines a nonlinear residual based on the parameters α_i, β_i and q,

$$f(U^n) = U^n + \Delta t \beta_0 \nabla \cdot \mathbf{F}(U^n) - \sum_{i=1}^{q} \left[\alpha_i U^{n-i} + \Delta t \beta_i \nabla \cdot \mathbf{F}(U^{n-i}) \right]. \tag{2.2}$$

In this work we demonstrate results using a second-order Crank-Nicolson method in which $f(U) = U^n - U^{n-1} + \Delta t \theta \left(\nabla \cdot \mathbf{F}(U^n) + \nabla \cdot \mathbf{F}(U^{n-1}) \right)$, $\theta = 0.5$. Fully implicit evolution of the equation (2.1) over a time interval $[t^{n-1}, t^n]$ is then accomplished through solution to the nonlinear root-finding problem $f(U^n) = 0$. To this end, we employ an *inexact Newton-Krylov* algorithm, as is standard for scalable parallel solution to coupled systems of partial differential equations [28].

Since all operations in the Newton-Krylov context require only linear complexity operations, the key component required for scalability of fully implicit simulations using this technology is an optimal preconditioning strategy for the inner Krylov

linear solver [25, 28]. In Newton-Krylov algorithms, at each Newton iteration a Krylov iterative method is used to solve Jacobian systems of the form

$$J(U)V = -f(U), \qquad J(U) \equiv I + \gamma \frac{\partial}{\partial U}(\nabla \cdot \mathbf{F}(U)), \qquad \gamma = \Delta t \beta_0. \tag{2.3}$$

The number of iterations required for convergence of a Krylov method depends on the eigenstructure of J, where systems with clustered eigenvalues typically result in faster convergence than those with evenly distributed eigenvalues [18, 19, 40]. Unfortunately, for a fixed Δt, as the spatial resolution is refined the distribution of these eigenvalues spreads, resulting in increased numbers of Krylov iterations and hence non-scalability of the overall solution algorithm. The role of a preconditioning operator P is to transform the original Jacobian system (2.3) to either

$$JP^{-1}PV = -f \ \text{(right prec.)}, \qquad \text{or} \qquad P^{-1}JV = -P^{-1}f \ \text{(left prec.)}.$$

The Krylov iteration is then used to solve one of

$$(JP^{-1})W = -f, \qquad \text{or} \qquad (P^{-1}J)V = X,$$

where $X = -P^{-1}f$ is computed prior to the Krylov solve or $V = P^{-1}W$ is computed after the Krylov solve. Scalable convergence of the method then depends on the spectrum of the preconditioned operator (JP^{-1} or $P^{-1}J$), as opposed to the original Jacobian operator J. Hence, an optimal preconditioning strategy will satisfy the two competing criteria:

1. $P \approx J$, to help cluster the spectrum of the preconditioned operator.
2. Application of P^{-1} should be much more efficient than solution to the original system, optimally with linear complexity as the problem is refined and with no dependence on an increasing number of processors in a parallel simulation.

In the next section, we develop an operator-based preconditioning strategy for the system of hyperbolic conservation laws (2.1).

2.2. Preconditioner Formulation. For linear multistep time integration approaches, we first rewrite the nonlinear problem (2.2) in the form

$$f(U) = U + \gamma \left[\partial_x F(U) + \partial_y G(U) + \partial_z H(U)\right] + g = 0, \tag{2.4}$$

where the terms $F(U)$, $G(U)$ and $H(U)$ denote the x, y and z directional hyperbolic fluxes, and the term g incorporates previous time-level information into the discretized problem. This nonlinear problem has Jacobian

$$J(U) = I + \gamma \left[J_F(U)\, \partial_x(\cdot) + J_G(U)\, \partial_y(\cdot) + J_H(U)\, \partial_z(\cdot)\right], \tag{2.5}$$

with, e.g., $J_F(U) = \frac{\partial}{\partial U} F(U)$. We use the notation $(\cdot)$ to denote the location at which the action of the linear operator takes place, e.g.

$$\left[I + \gamma\, J_F(U) \partial_x(\cdot)\right] V = V + \gamma\, J_F(U) \partial_x V.$$

Omitting the explicit dependence on U from the notation, and introducing nonsingular matrices L_F, L_G and L_H, we may re-write the Jacobian system (2.5) as

$$\begin{aligned}
J &= I + \gamma \left[J_F L_F^{-1} L_F \partial_x(\cdot) + J_G L_G^{-1} L_G \partial_y(\cdot) + J_H L_H^{-1} L_H \partial_z(\cdot) \right] \\
&= I + \gamma \left[J_F L_F^{-1} \partial_x \left(L_F(\cdot) \right) - J_F L_F^{-1} \partial_x \left(L_F \right)(\cdot) \right. \\
&\qquad + J_G L_G^{-1} \partial_y \left(L_G(\cdot) \right) - J_G L_G^{-1} \partial_y \left(L_G \right)(\cdot) \\
&\qquad \left. + J_H L_H^{-1} \partial_z \left(L_H(\cdot) \right) - J_H L_H^{-1} \partial_z \left(L_H \right)(\cdot) \right] \\
&= I + \gamma \left[J_F L_F^{-1} \partial_x \left(L_F(\cdot) \right) + J_G L_G^{-1} \partial_y \left(L_G(\cdot) \right) + J_H L_H^{-1} \partial_z \left(L_H(\cdot) \right) \right. \\
&\qquad \left. - J_F L_F^{-1} \partial_x \left(L_F \right)(\cdot) - J_G L_G^{-1} \partial_y \left(L_G \right)(\cdot) - J_H L_H^{-1} \partial_z \left(L_H \right)(\cdot) \right].
\end{aligned}$$

The preconditioning scheme in this approach is based on the assumption that the majority of the stiffness found in the Jacobian is a result of a small number of very fast hyperbolic waves. To develop an approach for separately treating only these fast waves, we consider the preconditioning matrix, P, constructed using a directional and operator-based splitting of J,

$$\begin{aligned}
P &= \left[I + \gamma J_F L_F^{-1} \partial_x \left(L_F(\cdot) \right) \right] \left[I + \gamma J_G L_G^{-1} \partial_y \left(L_G(\cdot) \right) \right] \left[I + \gamma J_H L_H^{-1} \partial_z \left(L_H(\cdot) \right) \right] \\
&\quad \left[I - \gamma J_F L_F^{-1} (\partial_x L_F) - \gamma J_G L_G^{-1} (\partial_y L_G) - \gamma J_H L_H^{-1} (\partial_z L_H) \right] \\
&= J + O(\gamma^2).
\end{aligned} \tag{2.6}$$

We denote these components as $P = P_F P_G P_H P_{local}$. Through constructing the operator P as a product in this manner, the preconditioner solve consists of 3 simpler, 1-dimensional implicit advection problems, along with one additional correction for spatial variations in the directional Jacobians J_F, J_G and J_H. Hence, $Pu = b$ may be solved via the steps (i) $P_F\, \chi = b$, (ii) $P_G\, w = \chi$, (iii) $P_H\, v = w$, and (iv) $P_{local}\, u = v$. We note that the splitting (2.6) is not unique, and that in fact these operations can be applied in any order. We discuss our technique for efficient solution of each of the above systems in the following two sections.

2.3. Directional Preconditioner Solves. We first consider solution of the three preconditioning systems P_F, P_G and P_H from (2.6) of the form, e.g. (x-direction)

$$P_F \chi = b \quad \Leftrightarrow \quad \chi + \gamma\, J_F L_F^{-1} \partial_x \left(L_F \chi \right) = b. \tag{2.7}$$

To this point L_F, L_G, and L_H are still unspecified. We take these to be $n \times n$ matrices whose rows are the left eigenvectors of the respective Jacobians, giving the identities,

$$L_F J_F = \Lambda_F L_F, \qquad \Lambda_F = \mathrm{diag}(\lambda^1, \ldots, \lambda^n), \qquad J_F R_F = R_F \Lambda_F,$$

where $R_F \equiv L_F^{-1}$ are the right eigenvectors ($n \times n$ column matrix), and λ^k are the eigenvalues of J_F. Through pre-multiplication of (2.7) by L_F, we have

$$L_F \chi + \gamma\, L_F J_F R_F \partial_x \left(L_F \chi \right) = L_F b \qquad \Leftrightarrow \qquad L_F \chi + \gamma\, \Lambda_F \partial_x \left(L_F \chi \right) = L_F b.$$

Defining the vector of characteristic variables $w = L_F \chi$, we decouple the equations,

$$w + \gamma \Lambda_F \partial_x w = L_F b \qquad \Leftrightarrow \qquad w^k + \gamma \lambda^k \partial_x w^k = \beta^k,\ k = 1, \ldots, n,$$

where w^k denotes the k-th element of the characteristic vector w, and $\beta = L_F b$.

Spatial discretization of each of the characteristic variables w^k in the same manner as the original PDE (2.1), results in a tightly-banded linear system of equations (tridiagonal, pentadiagonal, etc., depending on the method), to solve for the values w_j^k. For example the tridiagonal version due to a $O(\Delta x^2)$ finite-difference discretization is

$$w_j^k + \frac{\gamma\lambda_j^k}{2\Delta x}\left(w_{j+1}^k - w_{j-1}^k\right) = \beta_j^k. \tag{2.8}$$

In the ensuing results, we use a second order centered finite-volume approximation, with resulting systems for each w^k that are tridiagonal. Moreover, the above approach results not only in tridiagonal systems for each characteristic variable w^k, but the systems are in fact *block tridiagonal*, where each block corresponds to only one spatial $\{x, y, z\}$ row that is decoupled from all other rows through the domain in the same direction. Thus solution of these linear systems can be very efficient, as the computations on each row may be performed independently of one another.

Furthermore, since our initial assumption was that the stiffness of the overall system resulted from a few very fast waves, we need not construct and solve the above systems for each characteristic variable w^k. In cases where the wave speeds can be estimated, we may set a pre-defined cutoff to the number of waves included in the preconditioner. This reduction allows for significant savings in preconditioner computation. For those waves that we do not precondition, we approximate them as having wave speed equal to zero, i.e. solving with the approximation $\hat{\Lambda}_F = diag(\lambda^1, \ldots, \lambda^q, 0, \ldots, 0)$. Omission of the $(n-q)$ slowest waves in this fashion amounts to a further approximation of the preconditioner to the original discretized PDE system. Writing $\hat{P}_F$ as the x-directional preconditioner based on q waves, we may consider $\|\chi - \hat{\chi}\|_p$, where χ solves $P_F\chi = b$ and $\hat{\chi}$ solves $\hat{P}_F\hat{\chi} = b$, i.e.

$$\chi + \gamma J_F R_F \partial_x(L_F\chi) = b, \qquad \hat{\chi} + \gamma \hat{J}_F R_F \partial_x(L_F\hat{\chi}) = b,$$

where $\hat{J}_F = R_F\hat{\Lambda}_F L_F$. Left-multiplying by L_F and proceeding as before, we obtain

$$\begin{aligned} & w + \gamma\Lambda_F\partial_x w = L_F b, \qquad \hat{w} + \gamma\hat{\Lambda}_F\partial_x\hat{w} = L_F b, \\ \Leftrightarrow & \\ & w^k + \gamma\lambda^k\partial_x w^k = (L_F b)^k, \quad k = 1, \ldots, n \\ & \hat{w}^k + \gamma\lambda^k\partial_x\hat{w}^k = (L_F b)^k, \quad k = 1, \ldots, q \\ & \hat{w}^k = (L_F b)^k, \quad k = q+1, \ldots, n. \end{aligned}$$

Measuring the error between w and $\hat{w}$ in the l^p-norm ($1 \le p < \infty$), we have

$$\begin{aligned} \|w - \hat{w}\|_p^p &= \sum_{k=1}^{n} \left\|w^k - \hat{w}^k\right\|_p^p = \sum_{k=q+1}^{n} \left\|w^k - (L_F b)^k\right\|_p^p \\ &\le \sum_{k=q+1}^{n} \left\|\left(I + \gamma\lambda^k\partial_x(\cdot)\right)^{-1} - I\right\|_p^p \left\|(L_F b)^k\right\|_p^p \\ &\le \sum_{k=q+1}^{n} \left(\frac{\left\|\gamma\lambda^k\partial_x(\cdot)\right\|_p}{1 - \left\|\gamma\lambda^k\partial_x(\cdot)\right\|_p}\right)^p \left\|(L_F b)^k\right\|_p^p, \end{aligned}$$

where we have assumed for the last inequality that $\left\|\gamma\lambda^k\partial_x(\cdot)\right\|_p < 1$, $k = q+1,\ldots,n$ [16]. Lastly since $\chi = R_F w$, we obtain the error bound

$$\|\chi - \hat{\chi}\|_p \leq \|R_F\|_p \left[\sum_{k=q+1}^{n} \left(\frac{\left\|\gamma\lambda^k\partial_x(\cdot)\right\|_p}{1-\left\|\gamma\lambda^k\partial_x(\cdot)\right\|_p}\right)^p \left\|(L_F b)^k\right\|_p^p\right]^{1/p}.$$

Since the eigenvector matrices L_F and R_F may be renormalized as desired, and the eigenvalues are ordered so that $\lambda_i \geq \lambda_j$, for $i < j$, the dominant error from preconditioning only the q fastest waves is approximately

$$\frac{|\gamma\lambda^{q+1}/\Delta x|}{1-|\gamma\lambda^{q+1}/\Delta x|}.$$

Hence omission of waves with small eigenvalues compared to the dynamical time scale (i.e. $\gamma\lambda \ll 1$) will not significantly affect preconditioner accuracy.

2.4. Local Non-Constant Coefficient Correction Solve. The remaining component of the split preconditioner (2.6) comprises the local system $P_{local}u = v$,

$$\begin{aligned} & \left[I + \gamma J_F R_F \partial_x L_F + \gamma J_G R_G \partial_y L_G + \gamma J_H R_H \partial_z L_H\right] u = v \\ \Leftrightarrow \quad & \left[I + \gamma R_F \Lambda_F \partial_x L_F + \gamma R_G \Lambda_G \partial_y L_G + \gamma R_H \Lambda_H \partial_z L_H\right] u = v. \end{aligned}$$

We note that for spatially homogeneous Jacobians, $\partial_x L_F = \partial_y L_G = \partial_z L_H = 0$, in which case this system reduces to $u = v$. In keeping with the previous discretization approaches, we approximate this system in the same way as above, e.g.

$$\gamma R_F \Lambda_F \partial_x L_F \;\approx\; \tfrac{\gamma}{2\Delta x} R_{F,i}\Lambda_{F,i}\left(L_{F,i+1} - L_{F,i-1}\right).$$

These solves are spatially decoupled (with respect to u), resulting in a block-diagonal matrix whose solution requires only $n \times n$ dense linear solves at each spatial location.

3. Application to Compressible Gas Dynamics and Magnetohydrodynamics. The preconditioning method developed in the previous section is applicable to any system of multi-dimensional hyperbolic equations. In this section we apply it to the compressible gas dynamics and magnetohydrodynamics (MHD) systems of equations. We include a simple 1D example to illustrate the effect of the preconditioning method on the spectrum of the preconditioned Jacobian.

3.1. Equations of Compressible Gas Dynamics and MHD. Single fluid magnetohydrodyanmics is a mathematical description of a plasma which may be derived from kinetic equations assuming quasineutrality and no distinction between ions and electrons. The equations of gas dynamics form a subset of the MHD equations and may be simply obtained by setting the magnetic field, $\mathbf{B}$, to zero. The MHD equations couple the equations of compressible hydrodynamics with the low-frequency Maxwell's equations, and may be written in conservation form using rational units,

$$\partial_t U + \nabla\cdot\mathbf{F}(U) = S(U, \nabla\cdot\mathbf{B}) + S(U) \tag{3.1}$$

where the solution vector $U \equiv U(\mathbf{x},t)$ is given by $U = \{\rho, \rho\mathbf{u}, \mathbf{B}, e\}^T$, and the hyperbolic flux vector $\mathbf{F}(U)$ is given by

$$\mathbf{F}(U) = \left\{\rho\mathbf{u}, \rho\mathbf{u}\mathbf{u} + \left(p + \tfrac{\mathbf{B}\cdot\mathbf{B}}{2}\right)\bar{\bar{\mathbf{I}}} - \mathbf{B}\mathbf{B}, \mathbf{u}\mathbf{B} - \mathbf{B}\mathbf{u}, \left(e + p + \tfrac{\mathbf{B}\cdot\mathbf{B}}{2}\right)\mathbf{u} - \mathbf{B}(\mathbf{B}\cdot\mathbf{u})\right\}^T.$$

In these equations ρ is the density, $\mathbf{u}$ is the velocity, $\mathbf{B}$ is the magnetic field, p and T are the pressure and temperature, and e is the total energy per unit volume of the plasma. The system is closed by the equation of state, $e = \frac{p}{\Gamma-1} + \frac{\rho}{2}\mathbf{u}\cdot\mathbf{u} + \frac{1}{2}\mathbf{B}\cdot\mathbf{B}$, with the ratio of specific heats denoted by Γ and taken to be $5/3$ throughout this work. The left hand side of the above equations comprises the ideal MHD portion of the single-fluid MHD equations, and for completeness we have included a source term $S(U)$ above which allows diffusive terms (usually viscous, resistive and heat conduction terms; see Reynolds et al. [37] for these). As mentioned above, setting $\mathbf{B} = 0$ results in the compressible Navier-Stokes equations if viscosity and heat conductivity are included, or the compressible Euler equations describing the motion of a ideal compressible gas if diffusion terms are absent. The source term is also a placeholder for reaction terms which may occur, for example, in gas combustion systems.

Following the original work of Godunov [15], and subsequent work by Falle et al. [13] and Powell et al. [35] we have included a non-conservative source term $S(U, \nabla\cdot\mathbf{B}) = -\nabla\cdot\mathbf{B}\{0, \mathbf{B}, \mathbf{u}, \mathbf{B}\cdot\mathbf{u}\}^T$ to symmetrize the ideal MHD system. Inclusion of this source term leads to a non-singular Jacobian, J_F, for the ideal MHD system, and as long as $\nabla\cdot\mathbf{B}$ remains negligible, inclusion of this term will not affect conservation or accuracy of the PDE system. Finally, for the MHD system, a consequence of Faraday's law is that an initially divergence-free magnetic field must lead to a divergence-free magnetic field for all times, which corresponds to the lack of observations of magnetic monopoles in nature. This *solenoidal property* is expressed as $\nabla\cdot\mathbf{B} = 0$. The effect of implicit solvers and preconditioning on this property is discussed in Appendix A.

For both MHD and compressible gas dynamics, nonlinearities typically occur on the left hand side of equations (3.1), i.e, in the *hyperbolic* part of the system of equations. Examples of physical systems where the stiffness results from the large separation of scales or wave-speeds present in the hyperbolic portion of the equations are: (a) tokamak fusion MHD in which the dynamical time scales of interest are ten-to-hundred times slower than the fastest time scales present, and (b) low Mach number combustion in which the advective time scales of interest are nearly two orders of magnitude smaller than the acoustic time scales (altough the fastest time scales in combustion may be due to chemical reactions which are not part of the hyperbolic part of the equations and not considered here). Due to this stiffness, numerical simulations of such systems can benefit from the implicit approach presented in this work.

3.2. Preconditioner for the Gas Dynamics/Ideal MHD System. The development of the preconditioner for the gas dynamics or the ideal MHD system proceeds in almost exactly the same fashion as described in Section 2.2. For ideal MHD, we base our work on the 8-wave MHD system introduced by Powell et al. [35]. The row-matrix (now an 8×8 matrix) of left eigenvectors for this system is derived as $L_F = Q^{-1}\bar{L}_F Q$, where Q corresponds to the Jacobian of the conserved variable to primitive variable transformation, $(\rho, \rho\mathbf{u}, \mathbf{B}, e) \to (\rho, \mathbf{u}, \mathbf{B}, p)$, and $\bar{L}_F$ corresponds to the matrix of left eigenvectors of $\bar{J}_F$. This 8-wave formulation is equivalent to the ideal MHD system (3.1), so long as the solenoidal constraint $\nabla\cdot\mathbf{B} = 0$ is satisfied exactly. Hence, for such constraint-preserving states we have the identities

$$J_F = Q^{-1}\bar{J}_F Q, \quad L_F = Q^{-1}\bar{L}_F Q, \quad R_F = Q^{-1}\bar{R}_F Q,$$
$$\text{where } R_F \equiv L_F^{-1}, \quad L_F J_F = \Lambda_F L_F, \quad J_F R_F = R_F \Lambda_F.$$

The eight eigenvalues for the ideal MHD system are: $\Lambda_F = \text{diag}(u_x \pm c_f, u_x \pm c_a, u_x \pm c_s, u_x, u_x)$, where u_x is the x-velocity, and c_f, c_a, c_s are, respectively, the fast magnetosonic, Alfvén and slow magnetosonic speeds. Typically in tokamak fusion MHD,

$c_s \ll c_a < c_f$. Of the two identical eigenvalues u_x, one corresponds to the entropy wave while the other corresponds to advection of $\nabla \cdot \mathbf{B}$ in the 8-wave formulation.

In three-dimensional gas dynamics, the left and right eigenvectors are 5×5 matrices and can be easily generalized from a one-dimensional system presented in Chapter 3 of Reference [29]. The eigenvalues of inviscid gas dyanmics are: $\Lambda_F = \text{diag}(u_x \pm c, u_x, u_x, u_x)$, where c is the sound speed given by $c^2 = \gamma p/\rho$.

3.3. An Illustrative Example. Before applying the preconditioner to multi-dimensional nonlinear examples, we illustrate its benefit on a simple example. We consider the 1D linearized ideal MHD system,

$$\partial_t U + J_F^0 \, \partial_x U = 0, \tag{3.2}$$

where $J_F^0 \equiv \frac{d\mathbf{F}(U)}{dU}|_{U=U^0}$, under the following background state: $\{\rho_0 = 1.0, \rho\mathbf{u}_0 = 0, B_x = 0.1\cos(\alpha), B_y = 0.1\sin(\alpha), B_z = 1, p_0 = 0.01\}$, with $\alpha = 60^o$. This corresponds to a low-β tokamak plasma with B_z playing the role of the toroidal magnetic field, and B_x and B_y corresponding to the poloidal magnetic field. The wave speeds for this state are: $c_s = 0.006, c_a = 0.05$, and $c_f = 1.013$ (i.e. $c_f \propto 10c_a \propto 100c_s$, conditions similar to those encountered in tokamak fusion plasmas). We discretize (3.2) uniformly over the domain $x \in [0, L]$ with N finite volume cells, and mesh spacing $\Delta x = L/N$. We further assume periodic boundary conditions, a fourth order finite difference discretization of the spatial derivative and a θ-scheme time discretization. The spatial derivative is approximated as $(\partial_x U)_i \approx a(U_{i+1} - U_{i-1}) + b(U_{i+2} - U_{i-2})$, where in the fourth-order method $a = (1.5\Delta x)^{-1}$ and $b = (-12\Delta x)^{-1}$. We then solve the linear system $JU^{n+1} = g(U^n)$ at every time step, where
$g_i(U) = U_i - (1-\theta)\Delta t \, J_F \, [a(U_{i+1} - U_{i-1}) + b(U_{i+2} - U_{i-2})]$,

$$J = \begin{bmatrix} I & K & M & 0 & \cdots & 0 & -M & -K \\ -K & I & K & M & 0 & \cdots & 0 & -M \\ -M & -K & I & K & M & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 0 & -M & -K & I & K & M \\ M & 0 & \cdots & 0 & -M & -K & I & K \\ K & M & 0 & \cdots & 0 & -M & -K & I \end{bmatrix}, \tag{3.3}$$

$K = a\,\theta\Delta t \, J_F$, and $M = b\,\theta\Delta t \, J_F$. The preconditioner is defined by setting all but the $q-$stiffest eigenvalues to zero; we define $\tilde{\Lambda} = diag\{\lambda_1, \cdots \lambda_q, 0, \cdots, 0\}$ and then solve only for the corresponding characteristic variables. This results in a preconditioner matrix that is identical to (3.3), albeit with the sub-blocks K and M replaced with the approximate blocks $\tilde{K} = a\theta\Delta t R\tilde{\Lambda}L$ and $\tilde{M} = b\theta\Delta t R\tilde{\Lambda}L$, respectively.

We choose the time step Δt to be 500 times the explicit time step constrained by the usual CFL condition and use $\theta = 0.5$. The domain length and discretization size are $L = 2$ and $N = 512$, respectively. For this example, the real part of the eigenvalues of J are all unity, and its condition number is approximately 343. In Figure 3.1 we plot the eigenvalues of the preconditioned operator, JP^{-1}, where P uses $q = 0$ (no preconditioning), $q = 2$ (fast magnetosonic), $q = 4$ (Alfvén and fast magnetosonic), and $q = 8$ (all waves). For these values of q, the condition number decreases from 343 to 16 to 2 and finally to 1, leading to the observation that *for a linear system of hyperbolic conservation laws, the preconditioner is exactly equal to the Jacobian.*

4. Computational Results. Unless otherwise noted, all tests were performed using identical solver parameters: a relative nonlinear residual tolerance $\epsilon = 10^{-7}$,

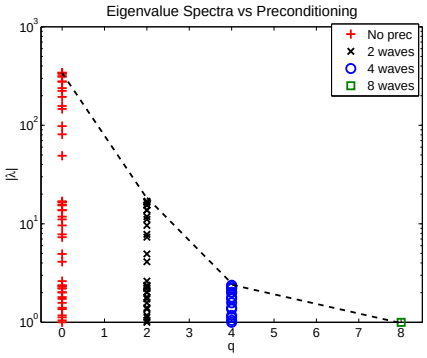

FIG. 3.1. *Eigenvalue spectrum as preconditioning is increased. All eigenvalues are of the form* $\lambda = 1 \pm i\lambda_I$*, we therefore plot* $|\lambda|$ *to indicate the shrinking spectrum as* q *increases.*

a maximum of 20 Newton iterations, at most 200 GMRES iterations per Newton step, an inexact Newton tolerance of $0.01\|f(U)\|_{WRMS}$, and a time-centered implicit discretization (i.e. $q = 1$, $\beta_0 = \beta_1 = 0.5$ from equation (2.2)). All preconditioned tests used the full 8-wave formulation with the splitting correction P_{local} (labeled FW Prec), and are compared against an implicit but un-preconditioned solver (labeled No Prec). Of these, all non-scaling studies were performed in serial, on a dual 3.0 GHz linux workstation, while all parallel scaling tests were performed on the Thunder Linux cluster at LLNL, consisting of 1024 1.4GHz Itanium-2 nodes (4 processors/node). The computational tests are performed using an implicit resistive MHD code described in [37], in which we have added the preconditioning approach. We examine this approach on a suite of stiff test problems from MHD and gas dynamics. We first examine a MHD linear wave propagation problem on which we expect the preconditioner solve to be nearly exact. This problem is then rotated to exercise the directionally-split nature of the preconditioner. The third test is a two dimensional Rayleigh-Taylor instability in compressible gas dynamics. The fourth problem is the Kelvin Helmholtz instability, a stringent nonlinear resistive MHD test that has been used to assess the efficacy of preconditioners for implicit MHD solvers [9]. We further provide parallel weak scaling results, as well as some investigation of various approximations to this preconditioning approach that may offer increased computational efficiency.

4.1. Linear Wave Propagation Tests. We begin with the "best case scenario" for this preconditioner, consisting of an ideal MHD linear wave advection test problem. It involves application of a small amplitude perturbation to an initially constant equilibrium, where the perturbation consists of an eigen-function corresponding to the slow magnetosonic wave [37]. Hence implicit integration of this problem consists of advection of this stiff, slow wave across the domain.

We set the computational domain to $[0,2] \times [0,2]$, with periodic boundary conditions in both the x and y directions, and initialize the equilibrium state $\tilde{U} = (\rho, \rho\mathbf{u}, \mathbf{B}, e)^T$, where $\rho = 1$, $\mathbf{u} = \mathbf{0}$, $\mathbf{B} = (\cos\alpha\cos\theta, \sin\alpha\sin\theta, 0)$ and $e = 0.1$. Here, $\theta = \tan^{-1}\frac{k_y}{k_x}$, $\alpha = -44.5^o$, and the ratio $\frac{k_y}{k_x}$ gives the direction of wave propagation. We project these to characteristic variables via $W = L\tilde{U}$, where L is left eigenvector matrix of the linearized MHD system, and then perturb the 6th-fastest characteristic, $w^6 = w^6 + 10^{-5}\cos(\pi k_x x + \pi k_y y)$. The initial condition is then set as $U^0 = RW$, where R is the right eigenvector matrix. All runs are taken to a final time of $T_f = 10$.

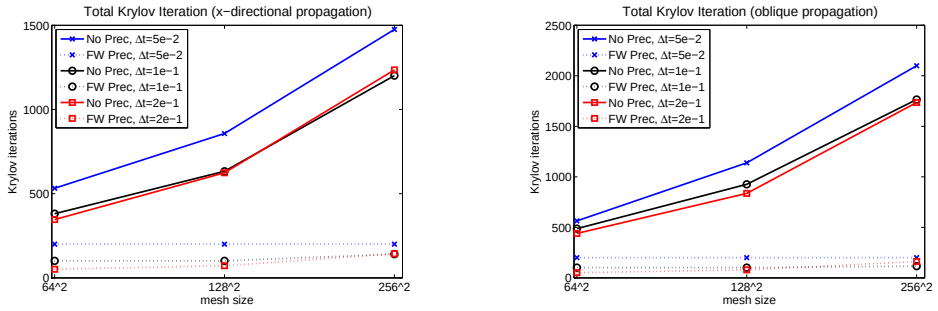

FIG. 4.1. *Krylov iterations for the linear wave tests: x-directional (left) and oblique (right).*

Mesh	Δt	NP Time (X)	FW Time (X)	NP Time (O)	FW Time (O)
64^2	0.05	13.426	13.100	13.864	15.284
64^2	0.1	8.104	7.920	9.097	7.564
64^2	0.2	5.820	3.991	7.310	3.859
128^2	0.05	70.393	64.763	85.321	63.295
128^2	0.1	45.559	32.447	62.747	32.605
128^2	0.2	43.914	19.495	57.916	18.926
256^2	0.05	568.336	338.796	727.269	322.778
256^2	0.1	419.905	189.650	609.195	174.714
256^2	0.2	449.975	129.825	713.768	136.875

TABLE 4.1
Runtimes for linear wave tests.

We note that even though the simulation code is fully nonlinear, the nonlinear effects are second order in the perturbation amplitude, and thus simulations still operate within the linearized MHD regime.

We examine two scenarios: in the first the wave propagates parallel to the x-axis (i.e. $k_x = 1$, $k_y = 0$), hence the directional splitting used in the preconditioner should not affect its accuracy. In the second the propagation is at a 45^o angle oblique to the x-axis (i.e. $k_x = k_y = 1$), exercising the directionally-split nature of the approach.

Results for both tests are shown in Figure 4.1. The total number of Krylov iterations is plotted for various time step sizes (different curves) and spatial discretizations (horizontal axis). For the x-directional test, the preconditioner is nearly exact, and hence the Krylov iterations remain nearly constant as the mesh is refined, as compared with the non-preconditioned tests that increase rapidly. For the oblique test, the directional splitting does not appear to significantly affect the preconditioner accuracy, again resulting in nearly constant Krylov iterations with mesh refinement. Table 4.1 shows the CPU timings, which suggest that for small problem sizes the reduction in linear iterations does not outweigh the computational cost of preconditioning; though once the mesh is refined the preconditioning benefit becomes increasingly apparent.

4.2. Rayleigh-Taylor Test. The Rayleigh Taylor instability (RTI) occurs whenever fluids of different density are subjected to acceleration in a direction opposite that of the density gradient [11]. RTI is encountered in a variety of contexts, such as combustion, inertial-confinement fusion, supernovae explosions and geophysics. Simulations of the compressible RTI are quite common (e.g. see Table 1 of [12] for a list of compressible RTI investigations and other references therein). Furthermore, in a variety of such simulations the Mach number is typically low ($\mathcal{O}(0.1)$ or less). This

begs for an implicit treatment of the fast acoustic time scales, as they evolve much more quickly than the advective time scales of physical interest. For this test, we augment the compressible gas dynamics equations with gravitational acceleration and viscous stress terms in the momentum equations, and the work done due to gravity and viscosity in the energy equation. Initially, the fluids are at rest under hydrostatic equilibrium. The density field is then perturbed acccording to the equations

$$\rho(x,y) = \frac{1}{2}(\rho_1+\rho_2) + \frac{1}{2}(\rho_2-\rho_1)\tanh[\psi(x,y)],$$
$$\psi(x,y) = y - \frac{1}{2}\sum_{k=1}^{N} \exp[-\sigma(k-2)^2]\left[a_1\cos(kx) + a_2\sin(kx)\right].$$

We set $\rho_1 = 1$ and $\rho_2 = 2$ (i.e. Atwood ratio is one-third), where a_1, a_2 are random amplitudes in the interval $[-0.5, 0.5]$ and $\sigma = 0.01$. The domain of investigation is $[-\pi/2, \pi/2] \times [-6, 6]$, with boundary conditions as outflow in y and periodic in x. We set the Froude number to 10 and Reynolds number to 10^5. Snapshots of the density field at $t = 0, 400$ are shown in Figure 4.2 for a 256^2 mesh simulation computed with a time step of $\Delta t = 0.25$. In this simulation the peak Mach number varied from 0 at $t = 0$ to approximately 0.011 at $t = 400$. In Figure 4.3 we present

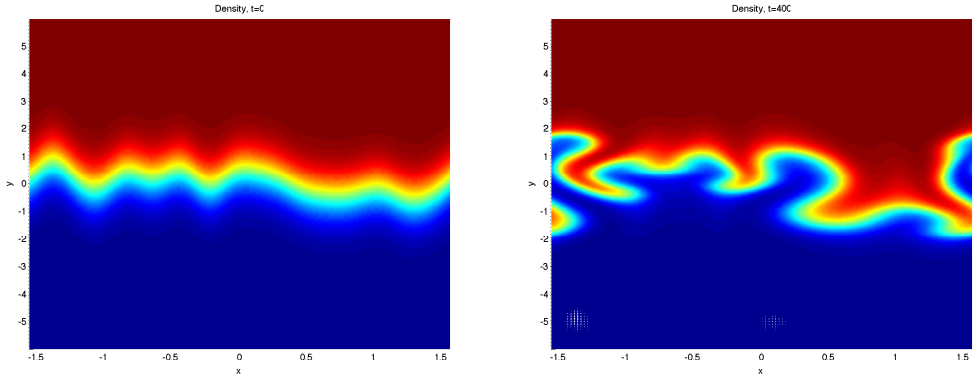

FIG. 4.2. *Snapshots of the ρ in the Rayleigh-Taylor test at $t = 0$ (left) and $t = 400$ (right).*

solver results, in which the preconditioned approach takes considerably fewer linear iterations for all time step sizes and spatial discretizations used. The CPU data in Table 4.2 demonstrates that for very small time steps and coarse meshes, the cost of preconditioning outweighs the runtime benefit resulting from the reduced iterations. For larger time steps or better-refined meshes, the preconditioned simulation again significantly outperforms the un-preconditioned solver.

4.3. Kelvin-Helmholtz Tests. We now investigate a more strenuous MHD problem, the Kelvin-Helmholtz test [11, 27]. This instability exhibits a large growth rate, and the dynamics quickly become nonlinear with tightly-coupled, highly inhomogeneous fields. We therefore consider this to be a "worst-case" test problem (among stiff MHD tests) for this preconditioning approach. The test should require multiple Newton iterations per time step, each requiring re-computation of the characteristic decomposition for projection onto individual waves. In addition, the test will significantly tax the splitting errors induced through the preconditioner formulation.

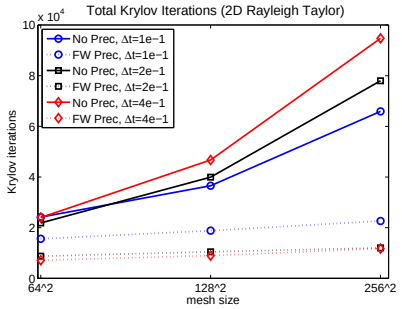

FIG. 4.3. *Krylov iterations for the 2D Rayleigh Taylor tests.*

Mesh	Δt	NP Time	FW Time
64^2	0.1	145.41	269.35
64^2	0.2	119.33	134.73
64^2	0.4	139.82	106.79
128^2	0.1	1129.5	1365.5
128^2	0.2	1157.8	678.36
128^2	0.4	1502.7	581.49
256^2	0.1	9791.4	7594.5
256^2	0.2	13862	3773.3
256^2	0.4	18951	3525.9

TABLE 4.2

Runtimes for 2D Rayleigh Taylor tests. Comparable explicit simulation times were 225.39, 3168.8 and 25404.0 seconds for the 64^2, 128^2 *and* 256^2 *tests, respectively.*

We set the computational domain to $\left[-\frac{5}{4},\frac{5}{4}\right]\times\left[-\frac{1}{2},\frac{1}{2}\right]\times\left[-\frac{5}{4},\frac{5}{4}\right]$, with periodic boundary conditions in the x and z directions, and homogeneous Neumann boundary conditions in the y direction. We initialize the constant fields $\rho = 1$, $\mathbf{B} = (0.1, 0, 10)$, $p = 0.25$, $\gamma = \frac{5}{3}$ and $u_y = u_z = B_y = 0$. We then set $u_x = \frac{1}{2}\tanh(100y) + \frac{1}{10}\cos(0.8\pi x) + \frac{1}{10}\sin(3\pi y) + \frac{1}{10}\cos(0.8\pi z)$, and initialize the total energy to $e = \frac{p}{\gamma-1} + \frac{1}{2}|\mathbf{B}|^2 + \frac{\rho}{2}|\mathbf{u}|^2$. This problem employs the resistive MHD equations, with resistivity, viscosity and heat conduction coefficients set to 10^{-4}, and all runs are taken to a final time of $T_f = 2$. As previous results on this problem suggest that the instability growth rate is independent of the size of the resistivity, such small parameters are natural since the instability is predominantly driven by nonlinear (hyperbolic) effects [6, 24, 27]. Moreover, for these parameters, $T_f = 2$ is well within the nonlinear evolution regime for this problem. The 2D tests ignore the z contribution to u_x, while the 3D tests replace GMRES with a BiCGStab linear solver using a maximum of 500 iterations.

Snapshots of the x and z components of the (initially homogeneous) magnetic field at $t = 2$ are shown in Figure 4.4 for a 256^2 mesh simulation computed with a time step of $\Delta t = 0.0025$. Throughout this simulation, the number of nonlinear iterations ranged from 1 to 3, with the associated preconditioned Krylov iteration counts beginning at 6 and ending at 13 per time step. Solver results for these tests are shown in Figure 4.5 and Table 4.3. The 3D non-preconditioned solver could not converge within the allowed iterations at $\Delta t = 0.01$ for larger mesh sizes, and could not converge at $\Delta t = 0.02$ for any but the coarsest mesh. For all time step sizes and all spatial discretizations used, the preconditioner results in significantly fewer linear iterations, with the disparity growing as the mesh is refined. However for small

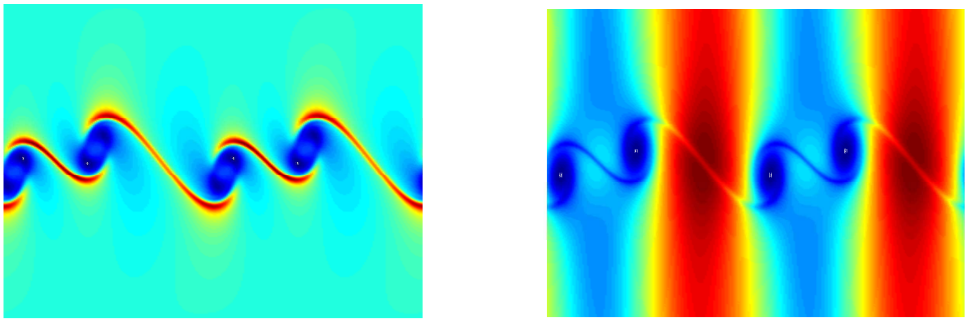

FIG. 4.4. *Snapshots of B_x (left) and B_z (right) in the 2D Kelvin Helmholtz test at $t = 2$.*

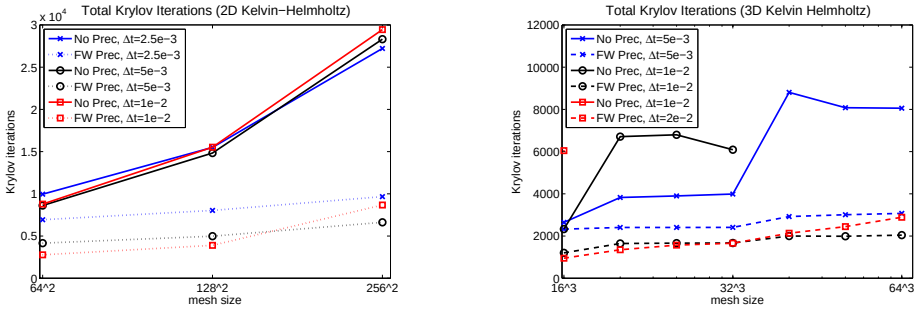

FIG. 4.5. *Krylov iterations for the 2D (left) and 3D (right) Kelvin Helmholtz tests. The 3D non-preconditioned solver could not converge within the allowed iterations for many of the $\Delta t = 0.01$ runs, or for nearly all of the $\Delta t = 0.02$ runs, so those portions of the curves are omitted.*

time steps and coarse spatial meshes we again see that the cost of preconditioning outweighs the runtime benefit due to the reduced iteration count. At larger time step sizes and for more refined meshes the preconditioned simulation is the clear winner.

4.4. Weak Scaling Tests. We also consider the scalability of this approach as both the problem size and number of processors are increased (i.e. *weak scaling*). Here, all problem specifications match those in earlier tests, except that the final time is reduced to $T_f = 0.5$. We plot total wall-clock time and total linear iterations required as we increase the computational mesh proportionately with the number of processors. Perfect weak scaling of these quantities would be represented by a horizontal line.

Results for the obliquely-propagating linear wave test problem and both 2D and 3D Kelvin Helmholtz test problems are presented in Figure 4.6. On the linear wave tests we note that the requisite linear iterations for convergence scales almost perfectly for all time step factors. While the linear iterations scale ideally, we note that CPU scaling depends on both a constant number of linear iterations *and* the scalability of the inner parallel periodic tridiagonal solver. The plot therefore investigates the scalability of this inner linear solver, showing modest increases in CPU time as the problem size is increased, with good scaling up to $p \approx 1000$ for these 2D tests.

For the more strenuous Kelvin Helmholtz tests we see how the preconditioner accuracy comes into play for moderate to large problem sizes. Here the 2D tests exhibit a slow growth in linear iterations up through $p \approx 256$, while the 3D tests show

Mesh	Δt	NP Time	FW Time
64^2	0.0025	267	359
64^2	0.005	220	204
64^2	0.01	228	117
128^2	0.0025	1720	1720
128^2	0.005	1740	980
128^2	0.01	2170	731
256^2	0.0025	16200	9950
256^2	0.005	19800	6170
256^2	0.01	26000	8050

Mesh	Δt	NP Time	FW Time
16^3	0.005	84.2	141.1
16^3	0.01	64.7	72.6
16^3	0.02	141.0	52.4
32^3	0.005	1044	1236
32^3	0.01	1368	790.5
32^3	0.02	—-	729.6
64^3	0.005	16393	12510
64^3	0.01	—-	7893
64^3	0.02	—-	10037

TABLE 4.3
Runtimes for Kelvin Helmholtz tests: left is 2D, right is 3D.

good scalability in linear iterations for all but the largest Δt. Such behavior could be due to either an increase in splitting inaccuracy or an increase in the second-order (viscous and resistive) terms that are left un-preconditioned with this formulation. These increases in linear iterations are borne out in the CPU scalings, where we notice a steady increase in required CPU time. Even with those increases, however, we point out that in 3D the required CPU time only goes up by about a factor of 6 in scaling up from 1 to 1000 processors.

4.5. Preconditioner Optimizations. In the preconditioner formulation from Section 2, a number of optimizations are readily apparent. We examine three such changes here: preconditioning only a subset of the MHD waves, omitting the local correction solve P_{local}, and freezing the eigen-decomposition used in the preconditioner to the initial condition. Each may provide increased efficiency over the full approach used in the previous results. For problems whose stiffness results from only the fastest MHD waves, preconditioning the slow waves may not contribute to the accuracy of the overall approach and may therefore be omitted. For problems that are close to spatially-homogeneous, the correction matrix $P_{\text{local}} \approx I$, and hence may be omitted. Lastly, for problems that do not deviate far from their initial conditions, the eigen-decompositions $J_* = R_* \Lambda_* L_*$ may remain nearly unchanged throughout a simulation, and therefore the initial decomposition could be reused.

We investigate such optimizations on both the oblique linear wave test and the 2D Kelvin Helmholtz test problems in Tables 4.4 and 4.5. Here, the preconditioners include the standard 8 wave approach above (Prec = 8), the 8 wave approach without the correction solve (Prec = 8 NC), the 8 wave approach without correction that is frozen on the initial condition (Prec = 8 NC-f), the non-corrected 4 wave formulation (Prec = 4 NC), and the non-corrected 2 wave formulation (Prec = 2 NC). We tested each of these approaches on a variety of spatial discretizations and time step factors ($C = \Delta t / \Delta t_{\text{CFL}}$). All simulations were performed for 50 time steps, and we present values of both the total linear iterations (Krylov) and the total run time (CPU). As demonstrated in these results, such optimizations can indeed provide benefits over the full preconditioner formulation; however it is difficult to *a priori* determine the best formulation for a given problem, as the optimal approach appears to be problem dependent. However, in most circumstances it appears that the splitting correction solve P_{local} may both slow convergence of the Krylov iteration and contribute to unnecessary overhead in the preconditioner evaluation.

5. Conclusions. We have introduced a preconditioner designed to alleviate stiffness induced by fast hyperbolic effects for problems posed on structured grids. Through splitting the implicit operator into its directional components along each

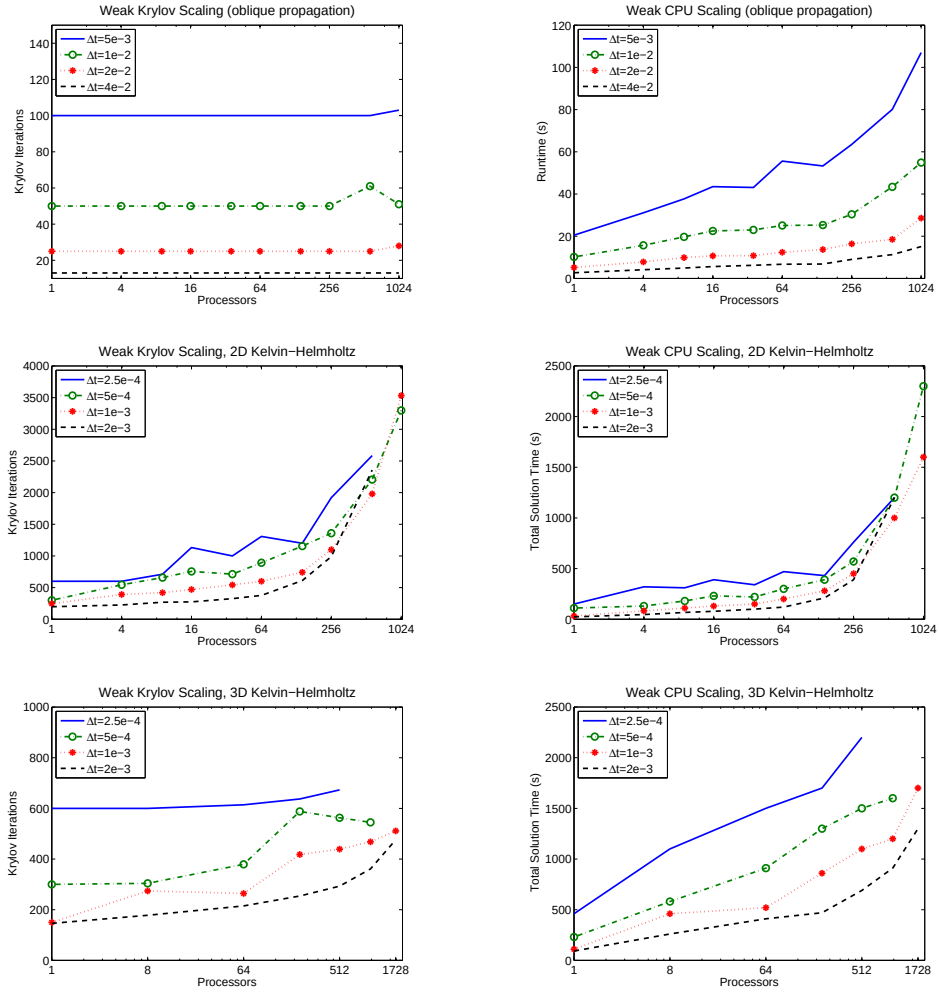

FIG. 4.6. *Weak scaling results for the oblique wave (top row), 2D Kelvin Helmholtz (middle row) and 3D Kelvin Helmholtz (bottom row) tests: linear iterations on left, CPU time on right.*

axial direction and then using the characteristic decomposition of the linearized operator appropriately, we approximate the solution to a multivariable 3D hyperbolic operator with a sequence of single-valued tridiagonal solves. Moreover, through using the characteristic decomposition we may choose to precondition any number of components, namely only those contributing the greatest stiffness to the problem.

Such an approach is not limited to uniform grid calculations. Block-structured AMR methods discretize PDEs with a hierarchy of regular grids to adaptively refine around regions of interest [3, 4, 39]. For such problems, *Fast Adaptive Composite* linear solvers are often used, which employ fast uniform-grid algorithms on each level in an iterative fashion to achieve the solution on the hierarchical composite mesh [31, 33, 34]. This preconditioning approach could easily be utilized in such an method.

As a result of the splittings on which this preconditioner is based, its accuracy may be limited by the time step size, the strength of propagation in directions oblique

Mesh	Prec	$C = 50$ Krylov	$C = 50$ CPU	$C = 100$ Krylov	$C = 100$ CPU	$C = 250$ Krylov	$C = 250$ CPU
128×128	8	1.0	0.6	1.0	0.6	1.0	0.6
128×128	8 NC	1.0	0.4	1.0	0.4	1.0	0.4
128×128	8 NC-f	1.0	0.4	1.0	0.4	1.0	0.4
128×128	4 NC	1.0	0.4	2.7	0.7	1.2	0.4
128×128	2 NC	1.0	0.5	2.7	0.6	1.2	0.4
256×256	8	1.0	2.8	1.0	2.8	4.0	5.1
256×256	8 NC	1.0	2.1	1.0	2.2	3.8	3.7
256×256	8 NC-f	1.0	2.1	1.0	2.1	3.8	3.7
256×256	4 NC	1.1	2.0	1.0	2.0	3.6	3.5
256×256	2 NC	1.1	2.1	1.0	2.1	3.6	3.7
512×512	8	1.0	13.1	1.1	12.7	1.8	15.0
512×512	8 NC	1.0	10.1	1.0	9.7	7.8	27.9
512×512	8 NC-f	1.0	10.1	1.0	9.7	7.8	28.0
512×512	4 NC	1.4	11.4	3.2	14.2	7.8	25.5
512×512	2 NC	1.4	11.7	3.2	16.4	7.8	30.2

TABLE 4.4

Comparisons of average linear iterations and CPU time per time step for various 'optimizations' on the oblique linear advection test.

Mesh	Prec	$C = 25$ Krylov	$C = 25$ CPU	$C = 50$ Krylov	$C = 50$ CPU	$C = 100$ Krylov	$C = 100$ CPU
128×64	8	15.8	1.9	27.1	2.8	95.7	9.4
128×64	8 NC	14.7	1.3	27.3	2.0	65.9	4.8
128×64	8 NC-f	13.9	1.1	21.0	1.6	42.6	2.8
128×64	4 NC	14.7	1.1	27.2	1.8	66.2	4.2
128×64	2 NC	14.8	1.0	26.3	1.6	65.1	3.7
256×128	8	13.6	8.8	23.9	11.2	72.3	30.5
256×128	8 NC	12.4	5.9	22.6	7.3	56.2	17.0
256×128	8 NC-f	11.8	5.3	17.9	5.9	29.9	8.7
256×128	4 NC	12.4	4.3	22.5	6.5	56.8	14.7
256×128	2 NC	11.4	3.4	21.1	5.5	54.5	13.0
512×256	8	10.7	29.1	21.2	49.4	58.8	122.2
512×256	8 NC	9.0	18.9	18.8	33.5	48.1	84.6
512×256	8 NC-f	10.0	19.3	14.4	25.8	26.7	44.8
512×256	4 NC	9.0	16.0	18.1	27.2	49.9	69.1
512×256	2 NC	9.4	14.9	17.3	23.9	45.8	55.4

TABLE 4.5

Comparisons of average linear iterations and CPU time per time step for various 'optimizations' on the 2D Kelvin-Helmholtz test problem.

to the coordinate axes, the spatial inhomogeneity of the fluxes that give rise to the local correction matrix P_{local}, and the number of waves that are treated with the preconditioner. However even with such approximations, we have demonstrated its effectiveness on a variety of problems that have been designed to exercise each of these error-inducing terms. All of these tests have one similarity in common – they are numerically stiff, having dynamical time scales that occur orders of magnitude more slowly than the fastest wave effects. In fact these are precisely the type of problems that require implicit methods; otherwise standard explicit-time approaches prove more efficient. We note that in a previous paper, we have shown that non-preconditioned implicit methods can result in tremendous efficiency improvements over explicit methods on stiff MHD problems [37]; through this preconditioning approach we have further improved the efficiency of such implicit simulations.

We have also investigated the weak scaling performance of the preconditioned implicit solver on a number of tests. In each case, the preconditioned solver scaled well as the problem sizes (and processor counts) were increased, even demonstrating nearly-ideal weak scaling in linear iterations for moderate processor counts ($p \leq 256$ in 2D, and $p \leq 1728$ in 3D). In those tests where the weak CPU scaling deteriorated,

the number of linear iterations required for convergence did not increase dramatically, indicating that improvements could still be made in the parallel implementation of the preconditioning approach (the parallel tridiagonal solver), extending its effectiveness to increasingly large processor counts. We intend to investigate this issue, as well as approaches for reducing the splitting error, in future work.

Acknowledgements. The authors wish to acknowledge the many insightful discussions with Luis Chacon, Michael Pernice, Steve Jardin, David Keyes and Phillip Colella during the course of this work.

REFERENCES

[1] P. Arbenz and W. Gander, *A survey of direct parallel algorithms for banded linear systems*, Tech. Report 221, ETH Zurich, November 1994.
[2] D.S. Balsara and D.S. Spicer, *A staggered mesh algorithm using high order Godunov fluxes to ensure solenoidal magnetic fields in magnetohydrodynamics simulations*, J. Comput. Phys., 149 (1999), pp. 270–292.
[3] M.J. Berger and P. Colella, *Local adaptive mesh refinement for shock hydrodynamics*, J. Comput. Phys., 82 (1989), pp. 64–84.
[4] M.J. Berger and J. Oliger, *Adaptive mesh refinement for hyperbolic partial differential equations*, J. Comput. Phys., 53 (1984), pp. 484–512.
[5] H. Bijl, *Iterative methods for unsteady flow computations using implicit Runge-Kutta integration schemes*, AIAA Paper 2006-1278, (2006).
[6] D. Biskamp, *Magnetic reconnection*, Phys. Rep., 237 (1994), pp. 179–247.
[7] L. Chacón, *A non-staggered, conservative, $\nabla \cdot B = 0$, finite-volume scheme for 3D implicit extended MHD in curvilinear geometries*, Comput. Phys. Comm., 163 (2004).
[8] ———, *An optimal, parallel, fully implicit Newton-Krylov solver for three-dimensional visco-resistive magnetohydrodynamics*, Phys. Plasmas, (2008).
[9] L. Chacón and D.A. Knoll, *A fully implicit 3D extended MHD algorithm*, Rome, Italy, June 2006, 33rd EPS Conference on Plasma Physics.
[10] L. Chacón, D.A. Knoll, and J.M. Finn, *An implicit, nonlinear reduced resistive MHD solver*, J. Comput. Phys., 178 (2002), pp. 15–36.
[11] S. Chandrasekhar, *Hydrodynamic and hydromagnetic stability*, Oxford Press, 1961.
[12] B. Le Creurer and S. Gauthier, *A return toward equilibrium in a 2d rayleigh-taylor instability for compressible fluid with a multidomain adaptive chebyshev method*, Theor. Comput. Fluid Dyn., 22 (2008), pp. 125–144.
[13] S. A. E. G. Falle, S. S. Komissarov, and P. Joarder, *A multidimensional upwind scheme for magnetohydrodynamics*, MNRAS, 297 (1998), pp. 265–277.
[14] T.A. Gardiner and J.M. Stone, *An unsplit Godunov method for ideal mhd via constrained transport*, J. Compup. Phys., 205 (2005), pp. 509–539.
[15] S. K. Godunov, *The symmetric form of magnetohydrodynamics equation.*, Numer. Method Mech. Contin. Media, 26 (1972).
[16] G.H. Golub and C.F. Van Loan, *Matrix Computations*, Johns Hopkins Press, 1996.
[17] T.I. Gombosi, D.L. De Zeeuw, K.G. Powell, A.J. Ridley, I.V. Sokolov, Q.F. Stout, and G. Toth, *Space Plasma Simulation*, vol. 615, Springer Verlag, 2003, ch. Adaptive mesh refinement for global magnetohydrodynamic simulation, pp. 247–274.
[18] A. Greenbaum, *Iterative Methods for Solving Linear Systems*, SIAM, Philadelphia, 1997.
[19] Anne Greenbaum, Vlastimil Pták, and Zdenuek Strakous, *Any nonincreasing convergence curve is possible for gmres*, SIAM J. Matrix Anal. Appl., 17 (1996), pp. 465–469.
[20] W. Gropp, D. Keyes, L.C. McInnes, and M.D. Tidriri, *Globalized Newton-Krylov-Schwarz algorithms and software for parallel implicit CFD*, Internat. J. High Perf. Comput. Appl., 14 (2000), pp. 102–136.
[21] D.S. Harned and W. Kerner, *Semi-implicit method for three-dimensional compressible magnetohydrodynamic simulation*, J. Comput. Phys., 60 (1985), pp. 62–75.
[22] ———, *Semi-implicit method for long time scale magnetohydrodynamic computations in three dimensions*, J. Comput. Phys., 65 (1986), pp. 57–70.
[23] A.C. Hindmarsh, P.N. Brown, K.E. Grant, S.L. Lee, R. Serban, D.E. Shumaker, and C.S. Woodward, *SUNDIALS, SUite of Nonlinear and DIfferential/ALgebraic equation Solvers*, ACM Trans. Math. Software, 31 (2005), pp. 363–396.

[24] T.W. Jones, J.B. Gaalaas, D. Ryu, and A. Frank, *The MHD Kelvin Helmholtz instability II. the roles of weak and oblique fields in planar flows*, Ap.J., 482 (1997), pp. 230–244.
[25] C.T. Kelley, *Iterative Methods for Linear and Nonlinear Equations*, SIAM, 1995.
[26] D.E. Keyes, D.R. Reynolds, and C.S. Woodward, *Implicit solvers for large-scale nonlinear problems*, J. Phys.: Conf. Ser., 46 (2006), pp. 433–442.
[27] D.A. Knoll and J.U. Brackbill, *The Kelvin-Helmholtz instability, differential rotation, and three-dimensional, localized, magnetic reconnection*, Phys. Plasmas, 9 (2002).
[28] D.A. Knoll and D.E. Keyes, *Jacobian-free Newton-Krylov methods: a survey of approaches and applications*, J. Compup. Phys., 193 (2004), pp. 357–397.
[29] C. B. Laney, *Computational Gasdynamics*, Cambridge Univeristy Press, Cambridge, UK, 1998.
[30] D.J. Mavriplis, *3-dimensional unstructured multigrid for the Euler equations*, AIAA J., 30 (1992), pp. 1753–1761.
[31] S.F. McCormick, *Multilevel Adaptive Methods for Partial Differential Equations*, Frontiers in Applied Mathematics, SIAM, Philadelphia, 1989.
[32] W. Park, E.V. Belova, G.Y. Fu, and X.Z. Tang, *Plasma simulation studies using multilevel physics models*, Phys. Plasmas, 6 (1999), pp. 1796–1803.
[33] M. Pernice and R.D. Hornung, *Newton-krylov-fac methods for problems discretized on locally-refined grids*, Comput. Vis. Sci., 8 (2005).
[34] M. Pernice and B. Philip, *Solution of equilibrium radiation diffusion problems using implicit adaptive mesh refinement*, SIAM J. Sci. Comput., 27 (2006), pp. 1709–1726.
[35] K.G. Powell, P.L. Roe, T.J. Linde, T.I. Gombosi, and D.L. DeZeeuw, *A solution-adaptive upwind scheme for ideal magnetohydrodynamics*, J. Comput. Phys., 154 (1999).
[36] J. Raeder, *Space Plasma Simulation*, vol. 615 of Lecture Notes in Physics, Springer Verlag, Berlin, 2003, ch. Global Geospace Modeling: Tutorial and Review, pp. 212–246.
[37] D.R. Reynolds, R. Samtaney, and C.S. Woodward, *A fully implicit numerical method for single-fluid resistive magnetohydrodynamics*, J. Comput. Phys., 219 (2006), pp. 144–162.
[38] D.R. Reynolds, F.D. Swesty, and C.S. Woodward, *A Newton-Krylov solver for implicit solution of hydrodynamics in core collapse supernovae*, J. Phys.: Conf. Ser., 125 (2008).
[39] R. Samtaney, P. Colella, T.J. Ligocki, D.F. Martin, and S.C. Jardin, *An adaptive mesh semi-implicit conservative unsplit method for resistive MHD*, J. Phys. Conf. Ser., 16 (2005).
[40] L.N. Trefethen and D. Bau III, *Numerical Linear Algebra*, SIAM, 1997.
[41] L. Wang and D.J. Mavriplis, *Implicit solution of the unsteady Euler equations for high-order accurate discontinuous Galerkin discretizations*, J. Comput. Phys., 225 (2007), p. 1994.

Appendix A. Solenoidal Preservation in Implicit MHD Solutions.

In this section we show that the use of fully implicit methods based on matrix-free inexact Newton-Krylov algorithms retains the solenoidal condition on the magnetic field for MHD calculations. To demonstrate this preservation, we first summarize the structure of these algorithms before moving on to the theoretical result.

In solving the nonlinear root-finding problem $f(U^n) = 0$, given in (2.2), inexact Newton methods generate a sequence of iterates $\{U_i\}$ that converge to the time-evolved solution U^n as follows [25]. Given an initial guess U_0 (typically chosen as some convex combination of previous states, $U_0 = \sum \alpha_k U^{n-k}, \sum \alpha_k = 1$), and nonlinear and linear tolerances ε and δ, the sequence $\{U_i\}$ is generated through the steps:

(i) Solve $J(U_i)V_i = -f(U_i)$ for V_i such that $\|J(U_i)V_i + f(U_i)\| < \delta$.
(ii) Set $U_{i+1} = U_i + V_i$, and if $\|f(U_{i+1})\| < \varepsilon$, stop.

In solving the systems (i), matrix-free Krylov methods [16] generate another sequence of iterates $\{S_k\}$ converging to V_i, where each iterate is chosen from the subspace

$$K_k(\tilde{J}, f) = \text{span}\{f, \tilde{J}f, \tilde{J}^2 f, \ldots, \tilde{J}^k f)\}, \tag{A.1}$$

and where $\tilde{J}$ approximates the Jacobian via a finite-difference,

$$\tilde{J}(U)V = [f(U + \sigma V) - f(U)]/\sigma = J(U)V + O(\sigma). \tag{A.2}$$

For solution algorithms of this type, we have the following result.

Theorem A.1. *If the spatial semi-discretization of* $(\nabla \cdot F(U))$ *is commutative, i.e.* $\partial^2_{xy} = \partial^2_{yx}$, *then the above algorithm applied to* (2.2) *exactly preserves an initially solenoidal magnetic field, for* <u>any</u> *nonlinear and linear tolerances* ε *and* δ.

Proof. We first define the space of all constraint-preserving discrete states,

$$\mathcal{V} = \{U = (\rho, \rho\mathbf{u}, \mathbf{B}, e)^T \in I\!R^{8N} \mid \nabla \cdot \mathbf{B} = 0\}.$$

Here N is the spatial discretization size. Due to the homogeneous linear constraint $\nabla \cdot \mathbf{B} = 0$, $\mathcal{V}$ is a vector space, since for any $U, V \in \mathcal{V}$ and $\alpha, \beta \in I\!R$, $\nabla \cdot (\alpha U + \beta V) = \alpha \nabla \cdot U + \beta \nabla \cdot V = 0$, hence $(\alpha U + \beta V) \in \mathcal{V}$ and by the vector space structure of $I\!R^{8N}$, the vector space structure of $\mathcal{V}$ follows.

Let $F_B(U)$ correspond to the $\mathbf{B}$ flux components, $F_B(U) = \nabla \times \mathbf{E}$, where from Ohm's law the electric field $\mathbf{E} = -\mathbf{u} \times \mathbf{B} + \eta \nabla \times \mathbf{B}$, so $\nabla \cdot (\nabla \cdot F_B(U)) = \nabla \cdot (\nabla \times \mathbf{E})$. While this identically equals zero on the continuum level, at the discrete level

$$\begin{aligned}\nabla \cdot (\nabla \times \mathbf{E}) &= \partial_x(\partial_y E_z - \partial_z E_y) + \partial_y(\partial_z E_x - \partial_x E_z) + \partial_z(\partial_x E_y - \partial_y E_x) \\ &= (\partial^2_{xy} - \partial^2_{yx})E_z + (\partial^2_{yz} - \partial^2_{zy})E_x + (\partial^2_{zx} - \partial^2_{xz})E_y,\end{aligned}$$

which only equals zero if the spatial discretization is commutative, which holds by assumption. Hence, given any element $U \in \mathcal{V}$, $(\nabla \cdot F(U)) \in \mathcal{V}$.

Therefore if $U \in \mathcal{V}$, $f(U) \in \mathcal{V}$ since $f(U)$ is formed as a linear combination of elements in $\mathcal{V}$. Moreover, the Krylov subspace $K_k(\tilde{J}, f) \subset \mathcal{V}$, due to the finite-difference approximation $\tilde{J}$ that recursively forms $K_k(\tilde{J}, f)$ out of elements in $\mathcal{V}$. Additionally, the updated solution is formed as the linear combination $U^n = U_0 + \sum V_i$, where $V_i \in K_k(\tilde{J}, f)$. Lastly, since the initial guess $U_0 = \sum \alpha_k U^{n-k} \in \mathcal{V}$ is the starting point for the iteration, and we begin the simulation with an initially solenoidal magnetic field, the updated solution $U^n \in \mathcal{V}$ as well. □

The salient features of this argument require that (a) the spatial semi-discretization is commutative, and (b) the Krylov algorithm is both matrix-free and not preconditioned. The first of these requirements is not met when upwind spatial discretizations based on Riemann or approximate Riemann solvers are used [2, 14]. For preconditioned simulations, the Krylov subspace is formed with the preconditioned operator, i.e. $K_k(P\tilde{J}, f)$, and hence a preconditioner may push the solution off of the constraint manifold. While we do not show that the preconditioned system satisfies the constraints, we have observed minimal deviation from the solenoidal property in our tests to date, as shown in Figure A.1. We note that a proof demonstrating similar behavior regarding the global conservation of mass, momentum and energy (exact conservation without preconditioning for any ε and δ) may be found in [7].

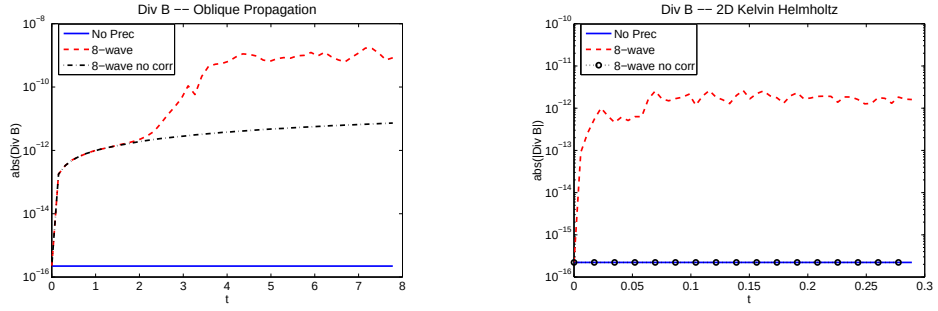

FIG. A.1. $\nabla \cdot \mathbf{B}$ *results due to preconditioning. Oblique wave propagation on a* 256^2 *mesh with* $\Delta t \approx 0.15$ *(left); 2D Kelvin Helmholtz on a* 512^2 *mesh with* $\Delta t \approx 0.006$ *(right). While preconditioning increases* $|\nabla \cdot \mathbf{B}|$ *error (worst with* P_{local}*), it remains well below simulation accuracy.*

The Princeton Plasma Physics Laboratory is operated
by Princeton University under contract
with the U.S. Department of Energy.

Information Services
Princeton Plasma Physics Laboratory
P.O. Box 451
Princeton, NJ 08543

Phone: 609-243-2750
Fax: 609-243-2751
e-mail: pppl_info@pppl.gov
Internet Address: http://www.pppl.gov