Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational Physics 219 (2006) 144–162
JOURNAL OF COMPUTATIONAL PHYSICS
www.elsevier.com/locate/jcp

A fully implicit numerical method for single-fluid resistive magnetohydrodynamics

Daniel R. Reynolds [a,*], Ravi Samtaney [b], Carol S. Woodward [c]

[a] *Department of Mathematics, University of California at San Diego, 9500 Gilman Drive, Dept 0112, La Jolla, San Diego, CA 92093-0112, United States*
[b] *Princeton Plasma Physics Lab, MS 26, P.O. Box 451, Princeton, NJ 08543, United States*
[c] *Lawrence Livermore National Lab, P.O. Box 808, L-551, Livermore, CA 94551, United States*

Received 13 May 2005; received in revised form 4 March 2006; accepted 18 March 2006
Available online 11 May 2006

Abstract

We present a nonlinearly implicit, conservative numerical method for integration of the single-fluid resistive MHD equations. The method uses a high-order spatial discretization that preserves the solenoidal property of the magnetic field. The fully coupled PDE system is solved implicitly in time, providing for increased interaction between physical processes as well as additional stability over explicit-time methods. A high-order adaptive time integration is employed, which in many cases enables time steps ranging from one to two orders of magnitude larger than those constrained by the explicit CFL condition. We apply the solution method to illustrative examples relevant to stiff magnetic fusion processes which challenge the efficiency of explicit methods. We provide computational evidence showing that for such problems the method is comparably accurate with explicit-time simulations, while providing a significant runtime improvement due to its increased temporal stability.
© 2006 Elsevier Inc. All rights reserved.

Keywords: Newton–Krylov; Implicit couplings; Resistive magnetohydrodynamics

1. Introduction

1.1. Motivation

The design of next-generation magnetic fusion devices requires increased understanding of nonlinear macroscopic stability, reconnection processes and refueling approaches for burning plasmas. Due to the high cost of conducting physical experiments of these processes in magnetic fusion devices, researchers are increasingly turning to computational simulation as a tool for such scientific investigation. However, it is well known that

* Corresponding author. Tel.: +1 858 534 5862; fax: +1 858 534 5273.
E-mail addresses: drreynolds@ucsd.edu (D.R. Reynolds), samtaney@pppl.gov (R. Samtaney), cswoodward@llnl.gov (C.S. Woodward).

0021-9991/$ - see front matter © 2006 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2006.03.022

the numerical modeling of magnetic-confinement fusion systems is one of the most challenging problems in contemporary computational physics. This is a result of many factors, including the complexity of models that accurately represent burning plasmas, as well as the resolution of the large range of spatio-temporal scales at which significant physical processes occur. A key result of temporal stiffness is that traditional explicit methods used for solution to such models may experience prohibitively small time step restrictions compared to the dynamical scales of macroscopic stability and plasma fueling.

In this work, we propose a fully implicit numerical approach for solving the single-fluid, resistive magnetohydrodynamic equations which constitute one of the relevant models describing burning plasma processes at the device scale. We root our implicit numerical methods in a Backwards Differentiation Formula–Newton–Krylov solution framework. There are many attractive qualities of such numerical techniques for these problems, including their resolution of nonlinear couplings between the disparately evolving internal physical processes and their increased temporal stability compared to traditional numerical solution methods for these problems.

A true description of plasma motion must rely on kinetic equations for each plasma species. As this approach is too costly for simulation of full magnetic fusion devices, a fluid description of the plasma is often used. This description is obtained by taking velocity moments of the kinetic equations describing a plasma under certain closure assumptions and the assumptions of large collisionality (see [1] for details). Magnetohydrodynamics, or MHD, is the term given to a single fluid description of a plasma in which a single velocity and pressure describe both the electrons and ions. This approximation is distinguished from *two-fluid MHD* in which electrons and ions retain separate pressures and velocities. The simplest MHD model is that of *ideal MHD*, which ignores the diffusion terms arising from collisions, assuming that these effects are negligible compared with other terms. When these diffusion terms are retained, the mathematical model is referred to as *single-fluid resistive MHD*, which is the primary focus of this paper. While single-fluid resistive MHD may be considered to be one of the simplest models used to describe plasma dynamics, it is nonetheless rich in mathematical structure and has been successfully employed to simulate physics at the device-scale [2,3]. We note that there have been a number of recent developments in the literature that are based on related physical models incorporating simplifications and/or incorporation of additional physical processes. An oft-used approximation of the MHD system in the presence of a strong magnetic field is to constrain the plasma compressibility in the direction perpendicular to the field. This asymptotic expansion results in simplified sets of modeling equations, and is generally referred to as *reduced MHD*. Additional processes that have been modeled are two-fluid effects including the Hall term and electron pressure gradients, under the umbrella of *extended MHD* [4].

In particular, we are interested in resistive MHD modeling of tokamaks, magnetic fusion devices in a toroidal confinement configuration having a strong background toroidal magnetic field. In the MHD modeling of tokamaks (and other confinement configurations), the numerical difficulties stem from: (1) a wide range of space scales, (2) a wide range of time scales, and (3) a large anisotropy induced by the background magnetic field. The presence of a large background field and toroidal geometry separates the effective speeds of the MHD waves into three branches, each with characteristic wave speeds that differ from one another by approximately an order of magnitude. Thus, if the characteristic transit times of the fast magnetosonic wave, the shear Alfvén wave, and the slow magnetosonic wave are denoted τ_F, τ_A and τ_S, these satisfy $\tau_F \ll \tau_A \ll \tau_S$. We are specifically interested in problems where the physical processes under study occur on the τ_A time-scale, or slower. Thus, for explicit methods whose time step is restricted by the CFL condition to be at the τ_F time-scale, the calculation will require an excessive number of time steps for resolution of the relevant physical processes. Our motivation to develop a Newton–Krylov technique for implicit integration of the resistive MHD equations stems from the desire to follow the dynamics relevant to these physical problems of interest and not the τ_F time scale restriction. Thus, this work is mainly concerned with addressing the second issue mentioned above, i.e., the wide range of time scales. We also note that Newton–Krylov techniques have been demonstrably successful in the implicit solution of similarly stiff physical systems, such as the Navier–Stokes equations, radiation hydrodynamics and a variety of other applications [5–7].

This paper is organized as follows. In the next subsection we briefly review previous work on different time approaches to MHD modeling. Section 2 introduces the single-fluid resistive MHD equations, and our numerical methods are described in Section 3. We then present results on a suite of test problems designed to verify

the numerical method and to demonstrate the usefulness of our approach on physical problems of reconnection and plasma fueling. Concluding remarks follow in Section 5.

1.2. Survey of other work

In this section we present a brief survey of related work. In general, time advancement for the numerical solution of MHD equations falls under three categories: (a) explicit, (b) semi-implicit, and (c) fully implicit methods. Explicit methods have been favored among the community that deals with MHD flows in the presence of shocks (e.g. the space-weather community [8], and others [9,10]). In these physical regimes, nonlinear phenomena at time scales comparable to the fast time scale need to be studied. In such applications, there is no separation of scales between the interesting physical processes and the fast time scale τ_F, so explicit methods prove to be efficient and adequate for such simulations.

For problems in which there is a significant separation of physical scales, however, alternate methods must be chosen. For such problems in which stability limits the utility of explicit-time methods, one traditional approach has been semi-implicit methods. These methods have been developed to treat the (usually linear) diffusion terms implicitly while explicitly treating the (nonlinear) hyperbolic portion of the resistive MHD equations (for example, see [11]). These kinds of semi-implicit methods are useful to integrate over the time stepping constraints imposed by the diffusion terms when resolving near-singular current layers or when some form of fast heat conduction is present. Other semi-implicit methods have treated not only the diffusion terms implicitly, but also the fast compressive magnetosonic waves. The CFL stability condition in such implementations is generally restricted to those imposed by the shear Alfvén waves. Examples of this type of method can be found in the work of Park et al. [2] and Harned and Kerner [12]. The semi-implicit method proposed by Harned and Schnack [13] went one step further and included implicit treatment for the shear Alfvén waves. Another semi-implicit approach to the resistive MHD equations was used by Tokman and Bellan [14] in which they linearized the MHD system and solved the resulting linearly implicit system using exponential propagation iterative techniques. This solution method allowed for increased accuracy and solver efficiency over other semi-implicit techniques.

We now turn our attention to the class of implicit time stepping methods generally referred to as fully implicit methods. We distinguish the fully implicit methods from the so-called "linearly implicit" methods. An excellent example of the latter work is that of Jardin and Breslau [15] wherein the right hand side of the equations written in semi-discrete form is expanded in a Taylor series about the previous time step retaining only terms which are first order in the time step. Thus, these linearly implicit methods are akin to a single Newton iteration of the fully implicit Newton–Krylov methods. In fully implicit methods, on the other hand, each time step requires solution of a nonlinear problem. A nonlinear implicit difference scheme for resistive MHD, developed by Jones et al. [16], used an approximate LU decomposition and a Gauss–Seidel iterative technique for solution of the linear systems resulting from the implicit operator. This implementation was somewhat unique because the hyperbolic terms were treated by an upwind flux-vector splitting scheme, whereas in previous work these terms have been computed using standard central difference methods. The examples in their paper were of somewhat modest complexity: a 1D unsteady Riemann problem that was solved explicitly, two steady-state 2D examples, and one example in 1D that was solved implicitly. Their 1D unsteady example exhibited a speed up of approximately three compared with the explicit computation.

The recent work by Chacón et al. [17] on the development of an implicit nonlinear resistive MHD solver is perhaps most closely related to our effort. They employed a matrix-free, preconditioned Newton–Krylov approach to solve the *reduced* MHD equations in 2D. They chose a second order Rannacher time stepping approach [18,19], a variant of Crank–Nicholson time stepping which damps high-frequency errors. The speed-up afforded by their implicit method over an explicit one ranged from approximately four (for a 64^2 mesh) to eight (for a 256^2 mesh). Chacón and Knoll [20] have also developed an implicit solver for Hall MHD in 2D, although slightly simplified through use of the reduced MHD approximation. In the Hall MHD case, the stiffness results from the Whistler waves, which obey a quadratic dispersion relation, and hence very small explicit time steps. Compared with their implicit results for single fluid resistive MHD the Hall MHD simulations yielded a larger speedup (ranging from 4 to 30) of the implicit solver compared with the explicit one.

In this work, we develop a fully implicit numerical method for single-fluid resistive MHD, which can be extended for simulating large magnetic fusion devices such as ITER. Similarly to the approach by Chacón et al., we employ a matrix-free, Newton–Krylov technique. Additional contributions of this work are that we use the more general compressible MHD formulation, we allow variable spatial accuracy from second to fourth order, our time integration allows for adaptive step and adaptive order BDF methods, and our verification includes 3D examples computed in parallel. One concern with the use of fully implicit techniques is that the linear system solves that are required in these methods can be very costly for large problems. Thus, we include large-scale 3D results demonstrating that these systems can be scaled appropriately for problems of physical interest. We note that in this paper we do not include Hall or two-fluid MHD effects.

The time-evolution approach taken in this work relates closely to the work of Rognlien et al. [21], in that we make use of the implicit ODE solver CVODE for MHD simulations. However, their work studies a demonstrably different regime of confined plasmas, that of edge-physics, through solving the Braginskii form of the MHD equations by splitting the system into two separate systems comprised of a (3-D) edge-plasma transport problem and a (2-D) plasma profile evolution. It is due to the differences in problem regime, as well as our fully coupled approach, that this work contributes to the field of implicit solutions for compressible, resistive MHD simulations.

2. Governing equations

The single-fluid resistive MHD equations couple the equations of hydrodynamics and resistive Maxwell's equations and may be written below in conservation form:

$$\frac{\partial U}{\partial t} + \nabla \cdot F(U) = \nabla \cdot F_{\mathrm{d}}(U), \tag{2.1}$$

where the solution vector $U \equiv U(\mathbf{x}, t)$ is,

$$U = \{\rho, \rho\mathbf{u}, \mathbf{B}, e\}^T,$$

and the hyperbolic flux vector $F(U)$, and the diffusive fluxes $F_{\mathrm{d}}(U)$ are given by

$$F(U) = \left\{ \rho\mathbf{u},\ \rho\mathbf{u}\mathbf{u} + \left(p + \frac{1}{2}\mathbf{B}\cdot\mathbf{B}\right)\bar{\bar{\mathbf{I}}} - \mathbf{B}\mathbf{B},\ \mathbf{u}\mathbf{B} - \mathbf{B}\mathbf{u},\ \left(e + p + \frac{1}{2}\mathbf{B}\cdot\mathbf{B}\right)\mathbf{u} - \mathbf{B}(\mathbf{B}\cdot\mathbf{u}) \right\}^T,$$

$$F_{\mathrm{d}}(U) = \left\{ 0,\ Re^{-1}\bar{\bar{\tau}},\ S^{-1}(\eta\nabla\mathbf{B} - \eta(\nabla\mathbf{B})^T),\ Re^{-1}\bar{\bar{\tau}}\cdot\mathbf{u} + \frac{\gamma}{\gamma-1}\frac{\kappa}{RePr}\nabla T + \frac{\eta}{S}\left(\frac{1}{2}\nabla(\mathbf{B}\cdot\mathbf{B}) - \mathbf{B}(\nabla\mathbf{B})^T\right) \right\}^T.$$

In the above equations, ρ is the density, $\mathbf{u}$ is the velocity, $\mathbf{B}$ is the magnetic field, p and T are the pressure and temperature, respectively, and e is the total energy per unit volume of the plasma. The plasma properties are the resistivity η, the thermal conductivity κ, and the viscosity μ, which have been normalized, respectively, by a reference resistivity η_{R}, a reference conductivity κ_{R}, and a reference viscosity μ_{R}. The ratio of specific heats is denoted by γ and taken to be 5/3 throughout this work. The non-dimensional parameters in the above equations are the Reynolds number, defined as $Re \equiv \rho_0 U_0 L/\mu_{\mathrm{R}}$, the Lundquist number, defined as $S \equiv \mu_0 U_0 L/\eta_{\mathrm{R}}$, and the Prandtl number, denoted by Pr, which is the ratio of momentum to thermal diffusivity. The non-dimensionalization is carried out using a characteristic tokamak length scale, L, and the Alfvén speed $U_0 = B_0/\sqrt{\mu_0\rho_0}$, where B_0, ρ_0, and μ_0 are the characteristic strength of the magnetic field, a reference density and the permeability of free space, respectively. The equations are closed by the following equation of state

$$e = \frac{p}{\gamma-1} + \frac{\rho}{2}\mathbf{u}\cdot\mathbf{u} + \frac{1}{2}\mathbf{B}\cdot\mathbf{B},$$

and the stress tensor is related to the strain as

$$\bar{\bar{\tau}} = \mu(\nabla\mathbf{u} + (\nabla\mathbf{u})^T) - \frac{2}{3}\mu\nabla\cdot\mathbf{u}\bar{\bar{\mathbf{I}}}.$$

Finally, 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 $\nabla \cdot \mathbf{B} = 0$.

3. Methods

In this section, we discuss the spatial and temporal discretization techniques used in our implicit solution approach. In addition, we present the nonlinear and linear solvers used, as well as our techniques for adaptive temporal error control.

3.1. High-order spatial discretization

The domain is discretized using finite volumes that are indexed with an n-tuple ((i,j) in 2D and (i,j,k) in 3D), and the conserved quantities are stored at cell centers. Each finite volume is bound by faces indexed as $(i \pm \frac{1}{2}, j, k)$ and so on. We discretize the divergence of the fluxes in (2.1) as

$$\left(\frac{\partial f}{\partial x}\right)_{i,j,k} = \frac{\tilde{f}_{i+\frac{1}{2},j,k} - \tilde{f}_{i-\frac{1}{2},j,k}}{\Delta x}, \tag{3.1}$$

where f may represent either the hyperbolic or the diffusive fluxes, and Δx is the mesh spacing in the x-direction (assumed uniform). The quantity $\tilde{f}_{i+\frac{1}{2},j,k}$ is referred to as the numerical flux through the face $\{i+\frac{1}{2}, j, k\}$, and is computed as a linear combination of the fluxes at cell centers as

$$\tilde{f}_{i+\frac{1}{2},j,k} = \sum_{v=-m}^{n} a_v f_{i+v,j,k}. \tag{3.2}$$

Our numerical framework provides the flexibility of using any of a group of different spatial discretization schemes. For a second-order central difference implementation, $m = 0$, $n = 1$ and $a_0 = a_1 = \frac{1}{2}$; for a fourth-order central difference approximation, $m = 1$, $n = 2$, and $a_{-1} = a_2 = -\frac{1}{12}$, $a_0 = a_1 = \frac{7}{12}$; and for tuned second-order central differences, $a_{-1} = a_2 = -0.197$, $a_0 = a_1 = 0.697$ [22]. These central difference approximations are free of dissipation errors, except perhaps near domain boundaries. They do, however, suffer from dispersion errors. Consequently, physical phenomena that are not well resolved can suffer from ringing. The dispersion errors can be minimized by using schemes such as the tuned-second order scheme, mentioned above, which has lower dispersion error than the central difference schemes. The numerical approximation to the divergence $\nabla \cdot \mathbf{B}$ is written as

$$\nabla \cdot \mathbf{B} = \frac{\widetilde{B}^x_{i+\frac{1}{2},j,k} - \widetilde{B}^x_{i-\frac{1}{2},j,k}}{\Delta x} + \frac{\widetilde{B}^y_{i,j+\frac{1}{2},k} - \widetilde{B}^y_{i,j-\frac{1}{2},k}}{\Delta y} + \frac{\widetilde{B}^z_{i,j,k+\frac{1}{2}} - \widetilde{B}^z_{i,j,k-\frac{1}{2}}}{\Delta z} + \mathcal{O}(\Delta x^p) + \mathcal{O}(\Delta y^p) + \mathcal{O}(\Delta z^p), \tag{3.3}$$

where B^α is the α-component of the magnetic field, the terms $\widetilde{B}^\alpha$ are evaluated as shown in Eq. (3.2), and p is the order of the spatial derivatives. If the numerical approximation of $\nabla \cdot \mathbf{B}$ is ensured to be zero at $t = 0$ then it can be easily shown that the numerical fluxes, as computed above, ensure that the solenoidal property of the magnetic field in the discrete sense is automatically satisfied, which for explicit schemes, guarantees satisfaction of the solenoidal magnetic field in time.

3.2. High-order adaptive time discretization

As previously discussed, we employ high-order implicit methods for time integration of the system (2.1), in order to accurately evolve the system along the dynamical time scales of interest.

3.2.1. Implicit BDF method

For our time integration method, we use the parallel ODE solver CVODE [23,24], developed at Lawrence Livermore National Laboratory and based on the VODPK package [25]. CVODE employs the fixed-leading-

coefficient variant of the Backward Differentiation Formula (BDF) method [26,27] and allows for variation in the order of the time discretization, as well as in the time step size.

The methods in CVODE are Predictor–Corrector in nature, so each time step begins with the calculation of an explicit predictor giving a first approximation to the solution at the new time level, $U^{n(0)}$. An implicit corrector is then employed to solve for the time step solution, U^n. Writing the original system (2.1) as

$$\frac{\partial U}{\partial t} = g(U), \tag{3.4}$$

these methods are based on equations of the form

$$U^n - \Delta t_n \beta_{n,0} g(U^n) - \sum_{i=1}^{q} [\alpha_{n,i} U^{n-i} + \Delta t_n \beta_{n,i} g(U^{n-i})] = 0,$$

where $U^j = U(t_j)$ are the recent time-discretized states, the value of q defines the BDF order (between 1 and 5), and the coefficients $\alpha_{n,i}$ and $\beta_{n,i}$ are uniquely determined by q and the recent history of step sizes $\Delta t_n = t_n - t_{n-1}$ (see [27,28]). At each time step, the multidimensional nonlinear system

$$G(U^n) \equiv U^n - \Delta t_n \beta_{n,0} g(U^n) - \sum_{i=1}^{q} [\alpha_{n,i} U^{n-i} + \Delta t_n \beta_{n,i} g(U^{n-i})] = 0 \tag{3.5}$$

must be solved to determine the time-updated solution U^n. For solution of this system, we employ an inexact Newton–Krylov method [29], described in Section 3.2.2.

Within the CVODE time integration and solvers, we scale the unknowns in order to better handle the multivariable coupled systems. Thus, we include an absolute tolerance (ATOL) for each unknown and a relative tolerance (RTOL) applied to all unknowns. These tolerances are then used to form a weight that is applied to each solution component during the time step from t_{n-1} to t_n. This weight is given as

$$w_i = \text{RTOL} \cdot |U_i| + \text{ATOL}_i, \tag{3.6}$$

and then the weighted root mean square norm

$$\|\mathrm{U}\|_{\text{WRMS}} = \left[N^{-1} \sum_{i=1}^{N} (U_i / w_i)^2 \right]^{1/2}, \tag{3.7}$$

where N is the total number of unknowns in the system (in our case, eight × the number of spatial grid points), is applied on all error-like vectors within the solution process. The weight w_i represents a tolerance in the component U_i, so a vector whose WRMS norm is 1 is regarded as "small".

3.2.2. Newton–Krylov nonlinear solver

The nonlinear systems (3.5) are solved using an inexact Newton–Krylov solver. Newton's method provides a standard, efficient solution technique for systems of nonlinear equations, where at each iteration m the state $U^{n(m)}$ is updated as the solution to a local linear model of the residual $G(U^{n(m)})$:

Iterate over m:

(i) Solve (approximately): $M(U^{n(m)})\delta U^{(m)} = -G(U^{n(m)})$, where $M(U) \approx I - \gamma J(U)$, $J(U) = \partial G(U)/\partial U$, and $\gamma = \Delta t_n \beta_{n,0}$,
(ii) Update: $U^{n(m+1)} = U^{n(m)} + \delta U^{(m)}$.

It is widely known that nonlinear solution methods benefit tremendously from accurate initial guesses, $U^{n(0)}$. In this work, the explicit predictor is used as an initial guess to the implicit nonlinear system. Thus, the initial guess uses a collection of available time-level states U^{n-i} already stored for determining the system (3.5). The effect of this predictor on the implicit solver is discussed in Section 4.2.

To solve the linear systems in step (i) above, we rely on a scaled, non-restarted GMRES iterative solver [30]. Such linear solvers prove quite efficient for large-scale problems since they do not require storage of the matrix M, only its action in matrix-vector products. Approximations to these products are applied based on the

observation that the Jacobian action can be approximated by finite differences of the nonlinear function around the current state U,

$$MV = (I - \gamma J)V \approx V - \gamma[G(U + \sigma V) - G(U)]/\sigma,$$

where the increment σ is given by $1/\|V\|_{\text{WRMS}}$ so that the product σV has norm 1.

3.2.3. Local temporal error control

A critical part of the time evolution method is its adaptive control of local error. We include brief comments on how this strategy works here, and we refer the reader to [23] for more details. At every time step, the local truncation error (LTE) is estimated and required to satisfy tolerance conditions such that

$$\|\text{LTE}\|_{\text{WRMS}} \leqslant 1. \tag{3.8}$$

In CVODE, if this test passes, the step is considered successful. If it fails, the step is rejected, a reduced step size $\Delta t'$ is attempted, and the error test is repeated. If it fails three times, the order q is reduced (if $q > 1$) or the step is restarted from scratch (if $q = 1$). In addition to adjusting the step size to meet the local error test, the solver periodically adapts the BDF method order with the goal of maximizing the step size Δt. The integration begins at order 1 and varies the order dynamically to pick the order q for which a polynomial of that order best fits the discrete data involved in the multistep method.

Within each of these time steps, Newton iteration is used to solve the nonlinear system (3.5). The stopping test for the Newton iteration is related to the local temporal error test and attempts to keep the nonlinear system errors from interfering with the local error control. We set the Newton convergence (stopping) test as

$$R\|\delta U^{(m)}\|_{\text{WRMS}} < 0.1\epsilon,$$

where R is a current estimate of the convergence rate, $\delta U^{(m)}$ is the Newton correction for iteration m, and ϵ is an estimate of the local solution error [23]. If at any iteration, $\|\delta U^{(m)}\|_{\text{WRMS}}/\|\delta U^{(m-1)}\|_{\text{WRMS}} > 2$ with $m > 1$, we consider the method divergent and reduce the time step by a factor of 4.

Within the Newton iteration, the linear system solution errors must also be controlled. We measure these errors with the linear residual vector and attempt to ensure that the linear iteration errors do not interfere with the nonlinear and local integration error controls by requiring that the norm of the linear residual be less than 5% of 0.1ϵ, i.e., 0.005ϵ.

The various algorithmic features of the solver described above are documented in [25,26] and summarized in [23,24].

4. Computational results

In this section, we present numerical results using the high-order space (fourth-order central difference) and time (implicit BDF) solution approach from Section 3. We compare this method with a fourth-order-accurate explicit-time Runge–Kutta method based on the same spatial discretization. In these comparisons, we seek to demonstrate the correctness of the computed results and the benefits of the implicit-time approach for both solution accuracy and computational speed when dealing with stiff resistive MHD systems. We note that all of the following results are in dimensionless form, using the non-dimensionalization discussed in Section 2.

These results are examined on a suite of three problems, each designed to test a different aspect of the method. In Section 4.1, we consider a 2D linear wave propagation test having analytical solution, which we use to demonstrate the high accuracy of our spatial and temporal schemes. The second example, in Section 4.2, is a stiff 2D reconnection problem that has been the subject of intense study in the literature [31]. With this example, we verify the correctness of solutions to the single-fluid resistive MHD model, as well as examine the efficiency of the implicit approach. In Section 4.3, we demonstrate the performance of our methods on a large-scale, 3D pellet injection problem.

Unless otherwise noted, all of the following implicit-time computational results use the solver parameters detailed in Table 1.

Table 1
Default parameters used in the implicit-time and nonlinear solution methods

Parameter	Value
Absolute tolerance on solution	10^{-11}
Relative tolerance on solution (RTOL)	10^{-7}
Maximum BDF method order	5
Maximum Newton iterations/time step	3
Maximum GMRES iterations/Newton iteration	5
Nonlinear tolerance coefficient	0.1
Linear tolerance coefficient	0.05

4.1. 2D Linear wave propagation tests

The first test is a 2D linear wave propagation problem. Here, we test the method using the purely hyperbolic portion of the MHD system (2.1), usually referred to as the equations of *ideal MHD*, by taking $F_d(U) \equiv 0$. In this problem, the domain is the square $[0,2]\times[0,2]$, having periodic boundary conditions on all sides. The initial conditions are set up as follows. Writing the ideal MHD equations in quasilinear form suitable for wave propagation along a direction $\xi = k_x x + k_y y$,

$$\frac{\partial U}{\partial t} + A_\xi \frac{\partial U}{\partial \xi} = 0,$$

where $U = \{\rho, u, v, B_x, B_y, p\}^T$, we begin with a constant equilibrium state, denoted as U_0, and project this state on to the characteristic space by taking the inner product of U_0 with the left eigenvectors of the matrix $A_\xi(U_0)$. Thus the k-th equilibrium characteristic variable is written as $W_{0,k}(U_0) \equiv (l_k(U_0), U_0)$, where l_k is the k-th left eigenvector of $A_\xi(U_0)$. The l-th wave is then initialized by perturbing only the l-th characteristic variable, i.e., $W_k(x,y,0) = W_{0,k}(U_0) + \delta_{kl}\epsilon\cos(k_x x + k_y y)$, where δ_{kl} is the Kronecker delta function. The perturbed characteristic variables are then projected back to physical space by multiplying W_k with the right eigenvectors of $A_\xi(U_0)$. In order to simulate a stiff physical system, the amplitude of perturbation is 10^{-5}, the l-th wave is chosen to be a slow magneto-acoustic wave moving obliquely at 45 ° ($k_x = k_y$) to the mesh, with the magnetic field aligned at 89.5 ° to the direction of propagation. To mimic wave propagation in a low-beta plasma, the initial equilibrium beta is chosen to be $\beta \equiv 2p/|B|^2 = 0.02$. Under these initial and boundary conditions, the linearized time-dependent wave propagation problem has an analytic solution which can be compared with the computed nonlinear solution. (We hasten to add that the small perturbation amplitude ensures that the waves propagate almost linearly, with nonlinear effects of $\mathcal{O}(\epsilon^2)$.) Hence, we use it to analyze the accuracy of the spatial and temporal numerical methods.

In Fig. 1, we show the error in a typical pressure wave solution computed using the explicit method and the implicit method with varying tolerances for a number of spatial discretizations. These errors are computed at times $t = 5000\Delta t_{\text{exp}}$, through comparing the computed pressure waves with the analytical solutions. As expected, as the implicit tolerance is tightened the solution accuracy increases, down to the high-order spatial discretization resolution. We also note that choosing weaker tolerances significantly degrades the solution accuracy, as seen from the implicit curve at RTOL $= 10^{-6}$.

In Fig. 2, we compare the scaled runtimes required to propagate the simulation to a time of $t = 100$ for the explicit and implicit methods, under a variety of spatial meshes. For this and the following examples, we plot the *scaled runtime* as the total CPU time divided by the spatial mesh size, to provide a measure of the work performed due to the time evolution method only (thus, a perfect integration method would exhibit a horizontal line, i.e., constant work per mesh point). We note that for the explicit-time method, satisfaction of the CFL stability condition results in doubling this scaled time with each spatial refinement; whereas the implicit method has no stability restriction on the time step. The resulting plot shows increasing benefit with the implicit approach, which is expected due to the stiff behavior built into this problem.

In Fig. 3, we show the propagation of the pressure wave, using a 128×128 spatial mesh. Here, we plot the values of p along the diagonal cross-section of the 2D plane, i.e., along $y = x$. These values are plotted for both the explicit and implicit method (RTOL $= 1e-7$) at the initial time of $t = 0$ and further along in time at $t = 250$.

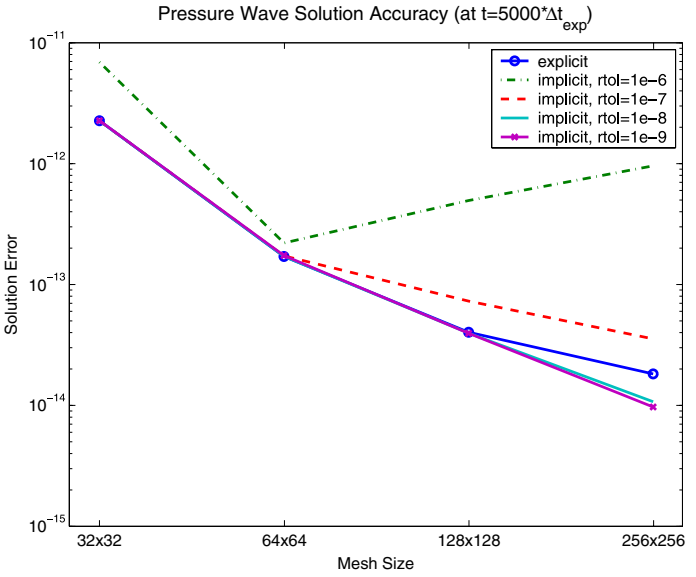

Fig. 1. For tight-enough solver tolerances, the implicit method captures the linear wave solutions at least as well as the explicit-time method.

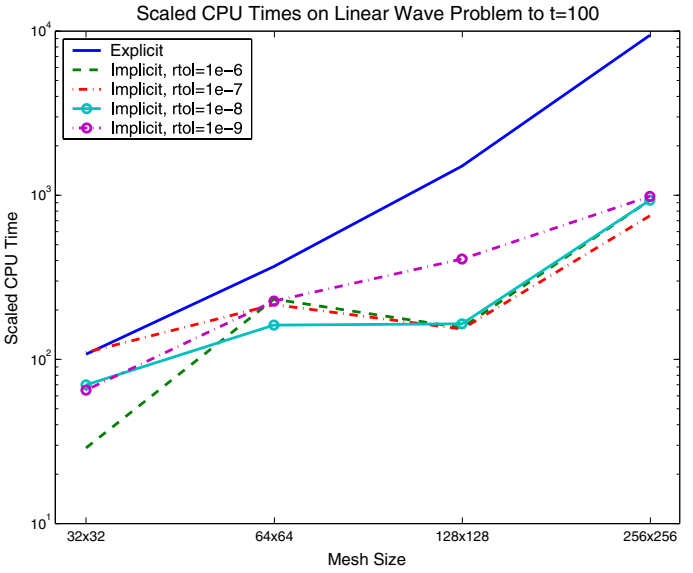

Fig. 2. Scaled CPU timings on the linear wave problem to $t = 100$. We note that the implicit timings are highly dependent on the tolerance RTOL.

As can be seen in the figure, both the explicit and implicit methods retain the original wave profile. It is of further note that due to the problem stiffness, the implicit method reaches the same solution as the explicit method in less than one-fifth the number of time steps, although each step requires slightly more computational work.

We note that tests corresponding to the fast magnetosonic wave and the Alfvén wave, with a variety of orientations to the magnetic field and the mesh, yielded similar accuracy and wave preservation comparisons with the explicit time method, and are not presented here in the interest of brevity.

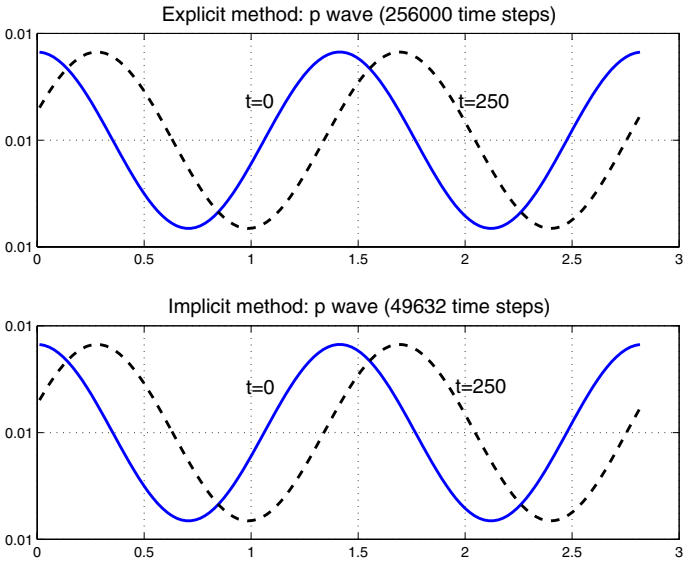

Fig. 3. The implicit method (bottom) reproduces the pressure wave propagation equally well as the explicit method (top), with significantly fewer time steps.

4.2. Magnetic reconnection in 2D

The term magnetic reconnection (MR) refers to the breaking and reconnecting of oppositely directed magnetic field lines in a plasma. In this process, magnetic field energy is converted to plasma kinetic and thermal energy, and occurs in many contexts, including the sawtooth-like oscillations observed in the operation of a tokamak and in solar coronal events. In MR two regions are generally distinguished: an outer "inviscid" region and an inner "resistive" region, whose width scales as $\eta^{1/2}$, where the actual reconnection process takes place. Of considerable interest in the reconnection problem is the speed at which the reconnection occurs (the *reconnection rate*) and its variation as a function of the magnetic resistivity η or Lundquist number S. For single-fluid resistive MHD, the reconnection rate for unforced reconnection should scale proportionally to the inverse square root of the Lundquist number, a dependence known as the *Sweet–Parker scaling* [32]. During reconnection, the process is characterized by a thin current layer whose thickness scales as $S^{-1/2}$, which when fully resolved results in explicit time steps proportional to the square of the local mesh spacing for an explicit method. Therefore, the computational time required for an explicit simulation to complete the reconnection process is expected to scale as $S^{3/2}$. As a result of the wide literature base on this problem, it is well-suited for testing the implicit solutions against known results. Moreover, due to the increasing stiffness of the problem, it is an ideal example for evaluating the efficiency improvements through using our implicit approach.

We have carried out simulations of MR in an idealized canonical 2D setting. The initial conditions consist of a perturbed Harris sheet configuration as described in Birn et al. in [31] (the *GEM magnetic reconnection challenge*). In this problem, the domain of simulation is a 2D box $[-12.8,12.8]\times[-6.4,6.4]$, where the characteristic length and velocity scales chosen are the ion inertial length and the Alfvén speed. It is clear that for single fluid MHD, the ion inertial length is not a meaningful quantity. Nonetheless, the simulation domain is chosen to be of the same size as in the GEM challenge. The boundary conditions are periodic in the x-direction with perfectly conducting wall boundary conditions in the y-direction. The non-dimensional parameters are chosen to be $S = 200$, $Re = 20$, and $Pr = 0.7$.

In Fig. 4, we plot snapshots of the current density $J = \nabla \times \mathbf{B}$, showing the time evolution of this 2D reconnection process. The initial state is given in the upper-left corner. As time proceeds (to the right), we see the formation of the thin reconnection layer in the center of the domain, indicative of resistive reconnection [32].

Two necessary features of numerical methods for MHD equations are retention of the solenoidal property of the magnetic field and satisfaction of conservation laws. As most often posed, the MHD equations imply,

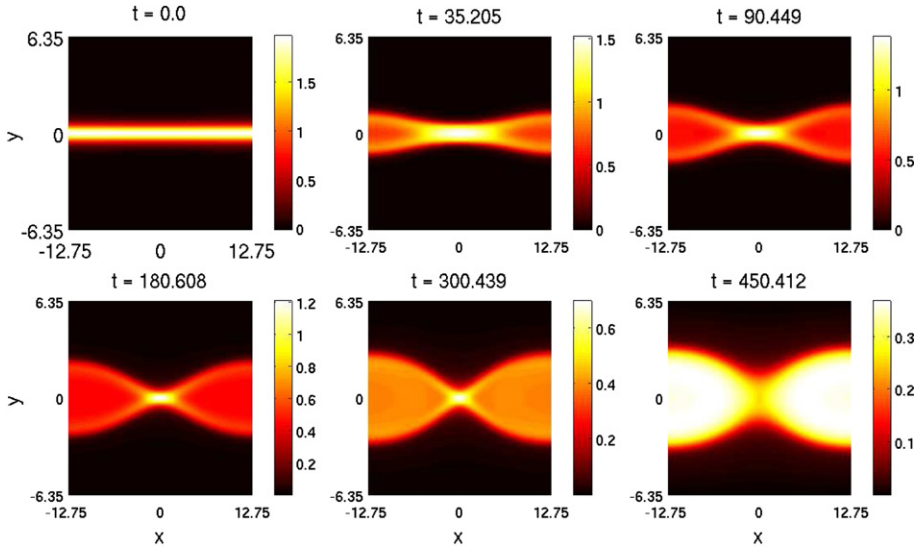

Fig. 4. Progress of the current field in time, exhibiting the magnetic reconnection. Time progresses from upper left to lower right. Note the progressive decrease of the maximum current value, as seen in the color-bar scales to the right.

but do not specifically enforce, the constraint that $\nabla \cdot \mathbf{B} = 0$, which is required for solution of Maxwell's equations. As a result, in numerical schemes where this constraint is not enforced, significant errors may be introduced into the magnetic field, resulting in a parallel Lorentz force and eventually contaminating the solution of all model variables. Similarly, conservation of energy is violated in numerical methods which do not solve the equation for total energy per unit volume, instead relying on an alternative non-conservative form of the energy equation using either the temperature or the pressure field. Through use of explicit methods in divergence form, conservation is guaranteed, and for suitably chosen fluxes, $\nabla \cdot \mathbf{B} = 0$ is also guaranteed through time. However, as implicit methods rely on a solver as opposed to explicit updates, such guarantees are no longer valid in general. In Fig. 5, we see that when given a solenoidal initial condition, our numerical methods retain the divergence-free magnetic field to numerical precision. Moreover, our numerical methods are conservative, as shown in Fig. 6, where we see precision-level variation in relative energy and mass levels over time. We note that these properties hold as a result of our Newton–Krylov methodology, and could be violated for other solution approaches.

We also use this example to compare simulation with theory, in order to check the correctness of the computed solutions. In Fig. 7, we plot the reconnection rate history for both the explicit and implicit time stepping schemes for a variety of Lundquist numbers. We note that although the implicit method uses significantly larger time steps, the computed histories of the reconnection rate are identical. More importantly, we also see that these rates satisfy the theoretical Sweet–Parker scaling in Fig. 8, where the maximum reconnection rates are plotted with respect to $S^{-1/2}$.

In addition to producing the correct results, we would like to see an efficiency benefit to using the implicit method for this stiff problem, since it is no longer constrained by the explicit time step stability limit. Fig. 9 shows the scaled run times required by the explicit and implicit methods to reach a physical time of $t = 50$, over a range of spatial discretizations. We see that when including the stiff diffusive effects of the full resistive MHD system (2.1), the implicit approach demonstrates significant runtime improvement over the explicit-time scheme. For these tests, the average explicit and implicit time steps were $\{2.9, 1.5, 0.75, 0.37\} \times 10^{-2}$ and $\{6.7, 5.7, 5.6, 5.9\} \times 10^{-2}$, from the coarsest to finest discretizations, respectively.

We also comment on the additional curve showing the scaled run time for the implicit method without use of the explicit-time predictor. This curve shows significantly hirer run times than the implicit method with the predictor. Without use of the predictor, the Newton method was started with an initial iterate equal to the previous time solution. We see clearly, that by using the predictor to generate an initial iterate based on a

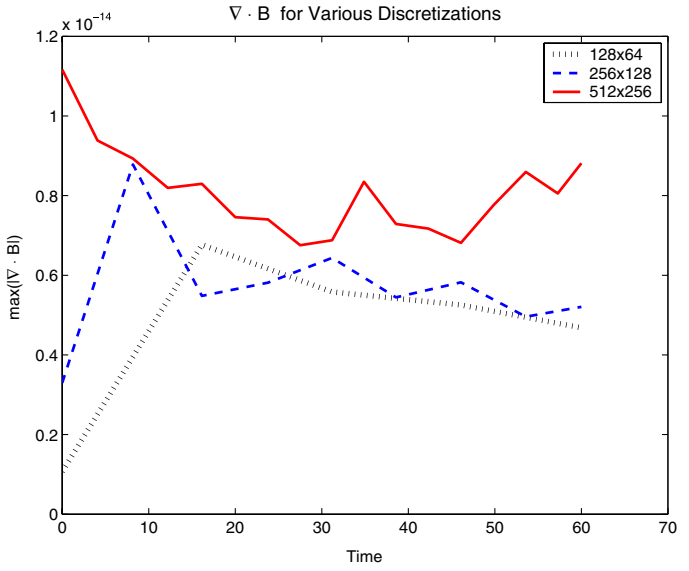

Fig. 5. The method retains the solenoidal magnetic field in time to computational roundoff error, thus not introducing unnecessary errors into the magnetic field.

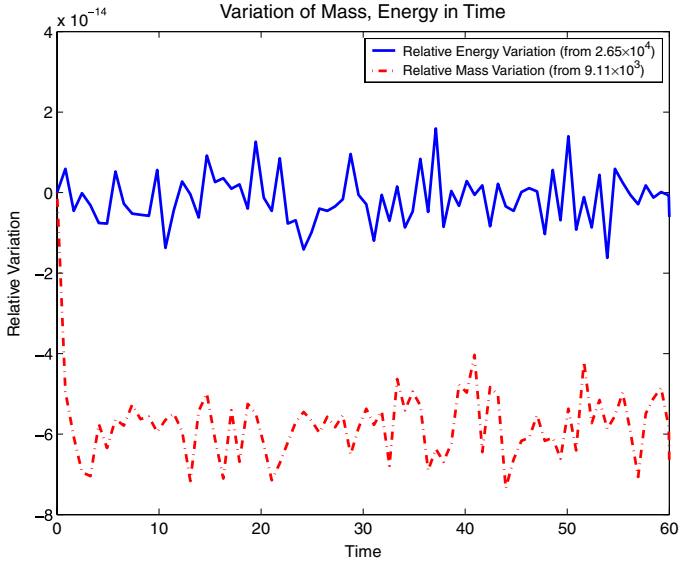

Fig. 6. We plot the relative variation from initial values of total energy and mass. As desired, the method conserves these to the level of computational roundoff error.

higher-order approximation, the implicit method will take fewer nonlinear iterations per time step and thus scale with a much better efficiency.

We note that although the above implicit time step sizes remain nearly constant, the corresponding run times grow due to increased solver load. This phenomenon may be understood through Table 2, which shows general trends of the nonlinear and linear convergence from a number of implicit simulations. The salient features in this table are the very slow increase in internal time steps, the decreasing difficulty of the nonlinear solve, and the increasing difficulty of the linear solve with respect to refining the spatial mesh. We note that the equivalent explicit method requires doubling the number of time steps with each spatial refinement, while

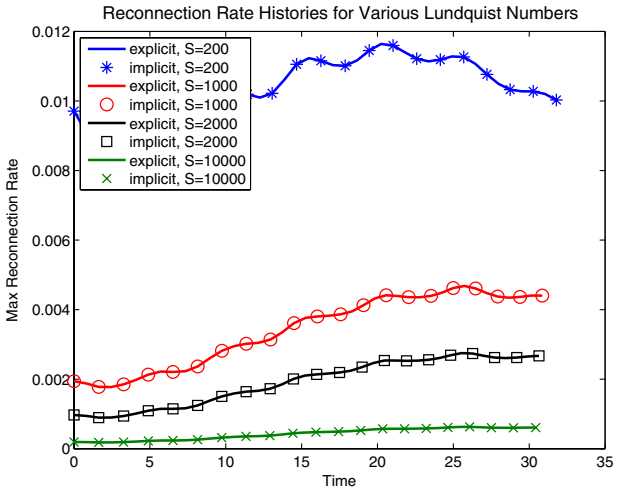

Fig. 7. The computed reconnection rates are identical between the explicit and implicit simulations, for a variety of Lundquist numbers (S).

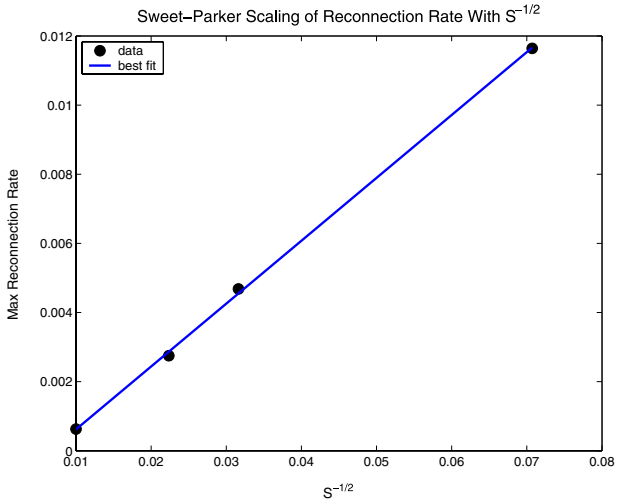

Fig. 8. The method exhibits the theoretical Sweet–Parker scaling between peak magnetic reconnection rate and the Lundquist number, S. The values are plotted using filled circles and scale linearly with the inverse square root of the Lundquist number.

the implicit method only increases these slightly as the mesh is refined. We attribute the nonlinear solver convergence to the fact that since more steps are taken, each step is smaller. The result of a smaller step is that the nonlinear system will be slightly easier and the predictor a little better. As the mesh is refined, however, the linear Jacobian systems are more difficult to solve due to a correspondence between the matrix conditioning and the mesh size. As the mesh size decreases, the conditioning of the matrix gets worse, and the linear problem more difficult. We expect future work in preconditioning to address this issue and remove this performance degradation with mesh size.

Lastly, we note that the difference in cost per time step between the explicit and implicit methods is fairly low for coarse meshes. The differences in cost per time step are partly due to the predictor, but they are also due to the fact that the linear solves for the implicit method on such a coarse mesh require few iterations and are thus fast. Also, the explicit method which uses a fourth order Runge-Kutta method requires four evaluations of the nonlinear function per step whereas, Table 2 indicates that the implicit method is taking about two evaluations per step for the coarse mesh.

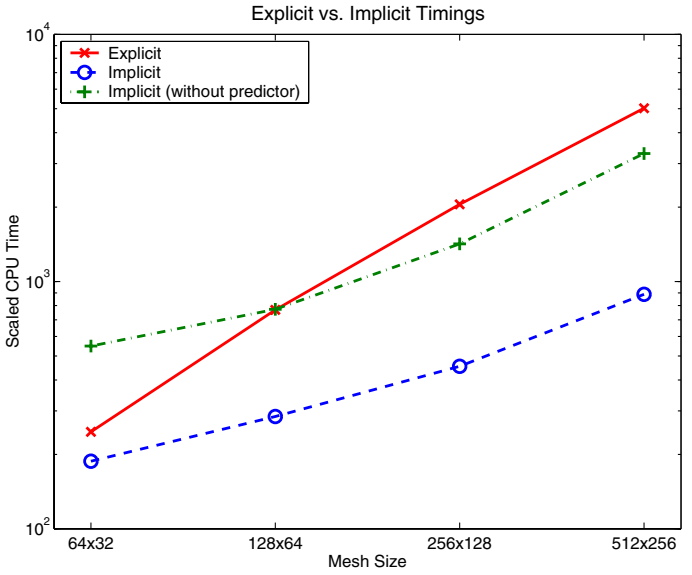

Fig. 9. Explicit and implicit scaled timings for 2D reconnection simulations up to a physical time of $t = 50$, for a variety of spatial resolutions. Note that the implicit method achieves speedups from 1.6× (64×32 mesh) to 5.6× (512×256 mesh), and that without the accurate predictor, these efficiency results are significantly degraded.

Table 2
Convergence rates attained with the implicit solver for the 2D reconnection problem: all simulations were run to a physical time of $t = 100$

Mesh	RTOL	Time steps	Newton/time step	GMRES/Newton
64×32	10^{-5}	1298	1.66	1.37
64×32	10^{-7}	1767	1.44	1.25
64×32	10^{-9}	2388	1.30	1.16
128×64	10^{-5}	1338	1.56	1.90
128×64	10^{-7}	1850	1.48	1.63
128×64	10^{-9}	2385	1.46	1.46
256×128	10^{-5}	1438	1.43	2.97
256×128	10^{-7}	1903	1.25	2.84
256×128	10^{-9}	2427	1.21	2.37
512×256	10^{-5}	2197	1.37	4.06
512×256	10^{-7}	3325	1.04	4.02
512×256	10^{-9}	2622	1.23	4.16

Equivalent explicit runs on the four meshes required 3404, 6647, 13287 and 26574 time steps, respectively.

4.3. Pellet injection

An experimentally proven method of refueling tokamaks is through pellet injection [33,34], which is currently seen as the most likely refueling technique for ITER, a next-generation fusion reactor currently under design. In this refueling process, small frozen deuterium pellets ($\mathcal{O}(10^3$–$10^4)$ times smaller than the reactor) are shot into the plasma at high velocity (from 300 to 1000 m/s). Simulations are required to understand the MHD processes that cause redistribution of the pellet mass in the tokamak. These pellets get rapidly heated by long mean-free-path electrons streaming along magnetic field lines, leading to ablation at the frozen pellet surface with a shield of neutral gas and an ionized high-density plasma cloud around it. This forms a local high-β plasmoid, which implies a localized region of high pressure that can trigger MHD instabilities. Fully resolving the high density cloud around the pellet leads to extremely small time steps in explicit simulations, as a result of the large toroidal guide field typically present in a tokamak. For such methods, simulation of the full pellet

ablation and mass redistribution processes thus requires $\mathcal{O}(10^6$–$10^7)$ time steps. However, in simulating such processes, it is not necessary to resolve the motion at time scales corresponding to fast magnetosonic speeds, but rather resolve the much slower motion of the ablated pellet cloud with a reasonable amount of accuracy and precision. Due to such problem stiffness, it is therefore desirable to have an implicit time stepping method, which may exceed the explicit step restriction and proceed at the time scale relevant to the pellet motion and ablation.

In this example, we do *not* simulate the full pellet injection process in a tokamak. Instead we use a model problem containing many of the time scale separations found in the original problem, albeit in a 3D Cartesian box domain instead of a toroidal domain. We choose this model problem in order to illustrate the implicit method in a 3D domain on a problem of interest with a large span of time scales, before extending these methods to the curvilinear meshes required for tokamak geometry. Following Samtaney et al. [35], our mathematical model consists of the single-fluid, resistive MHD equations with additional source terms to model the mass injected into the system by the pellet and a source term in the energy equations to model electron heating, which, in this example is instantaneous. In addition to these source terms, the model includes a variable for the pellet radius and an evolution equation for the ablation processes. Writing the pellet radius as r_p, the pellet ablation contributions to the density and energy as S_ρ and S_e, respectively, and the rate of pellet ablation as S_{r_p}, the model may be written

$$\frac{\partial U}{\partial t} + \nabla \cdot F(U) = \nabla \cdot F_\mathrm{d}(U) + S(U). \tag{4.1}$$

Here, the solution vector is now $U = \{\rho,\ \rho\mathbf{u},\ \mathbf{B},\ e,\ r_\mathrm{p}\}^T$, and the vectors $F(U)$, $F_\mathrm{d}(U)$ are the same as in (2.1), augmented with identically zero extra entries corresponding to the equation governing the pellet radius r_p. The source term $S(U)$ is given by

$$S(U) = \{S_\rho,\ 0,\ 0,\ S_\mathrm{e},\ S_{r_\mathrm{p}}\}^T.$$

The ablation rate of the pellet, originally derived by Parks and Turnbull [36] and modified for hydrogen pellets by Kuteev [37] is given as (in atoms/s)

$$\dot{N} = -4\pi r_\mathrm{p}^2 \frac{\mathrm{d}r_\mathrm{p}}{\mathrm{d}t} 2n_m = 1.12 \times 10^{16} n_\mathrm{e}^{0.333} T_\mathrm{e}^{1.64} r_\mathrm{p}^{1.33} M_\mathrm{i}^{-0.333}, \tag{4.2}$$

where n_e is the background plasma density in cm^{-3}, T_e is the background plasma electron temperature in eV, M_i is the atomic mass number in atomic units, and $n_\mathrm{m} = 2.63 \times 10^{22}/\mathrm{cm}^3$ is the molecular density of frozen hydrogen. The density source term arises from the ablation of the pellet and is written in terms of number density, (i.e., atoms per unit volume per unit time) as $S_\mathrm{n} = \dot{N}\delta(x - x_\mathrm{p})$, where the delta function is approximated as a Gaussian distribution centered over the pellet with a characteristic size equal to $10r_\mathrm{p}$. The density source term, S_ρ, is obtained by expressing S_n in the appropriate non-dimensional form. A useful approximation that eliminates the electron timescale from the problem is to consider the electron heat flux as being instantaneous compared to the other processes being computed. This leads to a source term in the energy equation, i.e., $S_\mathrm{e} = 3S_\mathrm{n}T(\psi)$, which corresponds to the localized increase in energy due to the heating of the ablated pellet mass, where $T(\psi)$ is the temperature of the flux surface. Isosurfaces of ψ are flux surfaces at $t = 0$ as discussed in the next paragraph. It is obvious that the pellet perturbs the magnetic field locally, and hence the use of $T(\psi)$ is because of the model ansatz that the perturbation is small.

The following simulations are performed on the 3D box domain, $[0,2]\times[-1,1]\times[-1,1]$, using periodic boundary conditions in the z-direction to simulate the toroidal tokamak direction, and perfectly conducting wall boundary conditions in the x and y directions to approximate the material walls of the tokamak. The initial condition is a Taylor state given by $\psi(x,y) = \psi_0 \sin(2\pi x)\cos(2\pi y)$. The initial magnetic field is given by $\mathbf{B}(x,y,z,0) = \nabla z \times \nabla\psi + g_0\nabla z$. The constants $\psi_0 = 2/\pi$ and $g_0 = 10$ result in a magnetic field in the x–y plane (corresponding to a poloidal plane in a tokamak) approximately one order of magnitude smaller than the z-component magnetic field (which corresponds to the toroidal component). The non-dimensional parameters for the pellet injection simulations were taken to be $S = 10^3$, $Re = 10^2$ and $Pr = 0.7$. The pellet radius is chosen to be 0.0125 times the box size which is an additional simplification to resolve the ablation cloud adequately in this illustrative example. We also note that due to the potential instabilities introduced by the high

density cloud surrounding the pellet, the following results use fluxes computed with the tuned second-order scheme discussed in Section 3.1.

In Fig. 10, we see the resulting pellet motion and ablation from these simulations. In this figure, we see the pellet source is strongest in the area of largest density. The time history of the total kinetic energy, shown in Fig. 11, shows a large increase in kinetic energy due to the ablated mass motion along the field lines. More-

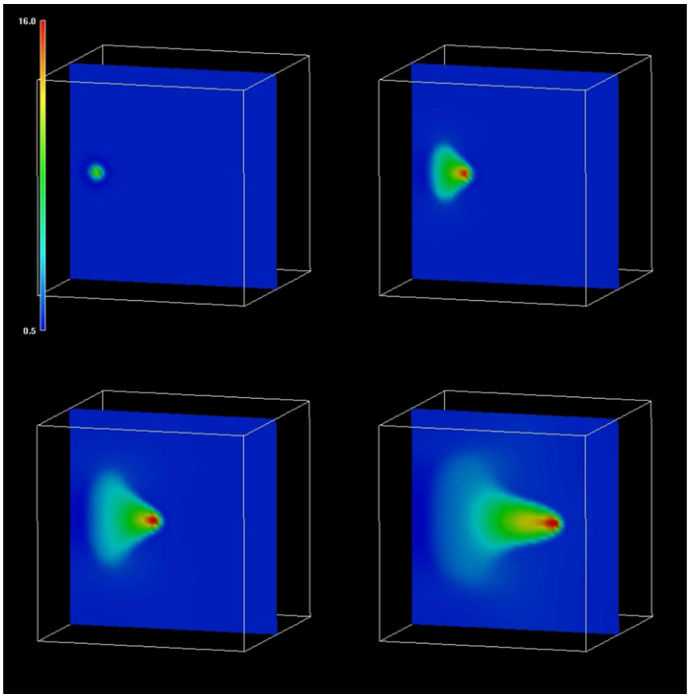

Fig. 10. Time sequence of density for pellet injection from top left to bottom right, at times $t = \{0.8, 9.8, 20.2, 40.0\}$.

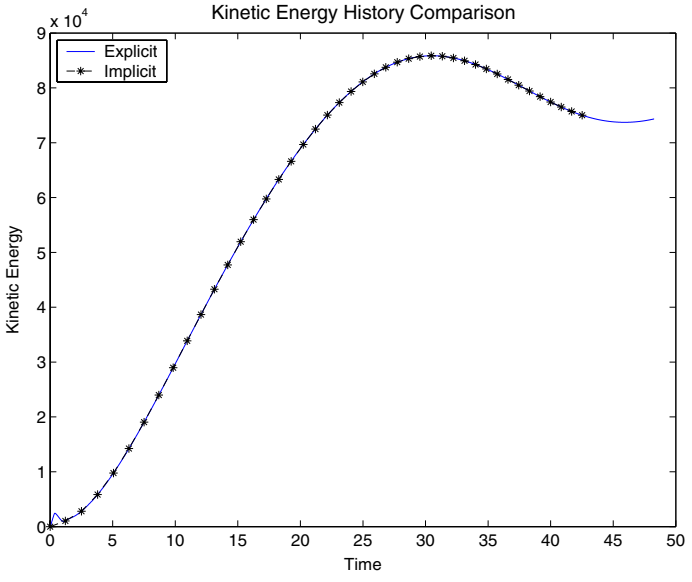

Fig. 11. Time history of the kinetic energy for explicit and implicit simulations.

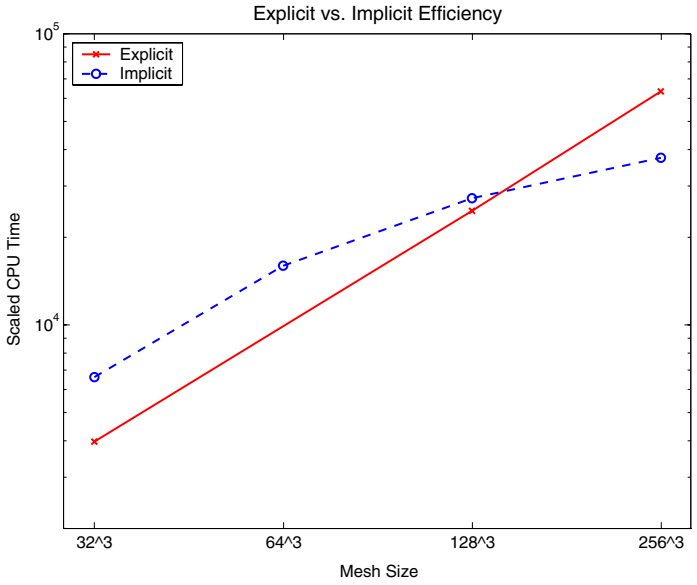

Fig. 12. 3D Pellet Injection timings: explicit and implicit method scaled CPU timings for simulations up to a physical time of $t = 3$, at a variety of spatial resolutions.

over, we see that the implicit method exactly matches the kinetic energies produced by the original explicit method.

As previously mentioned, we present this pellet injection example to test the efficiency of the fully implicit approach on a 3D problem that exhibits physical aspects of scientifically interesting problems. These results are shown in Fig. 12, in which we again plot the scaled CPU time per spatial mesh point at varying spatial resolutions. We see that although the implicit method proves significantly slower for low-resolution problems, it overtakes the explicit method as the mesh is refined toward more reasonable spatial accuracies. In each of the four examples, we used a uniform discretization of the domain in each direction and increased the number of processors to maintain a uniform data-per-processor ratio (i.e., 1, 8, 64, 256 processors for the four resolutions), so the timings shown include inter-processor communication. For these simulations, the average explicit time steps were $\{10.5, 5.3, 3.3, 1.7\} \times 10^{-4}$, while the average implicit time steps were $\{5.1, 4.8, 3.6, 3.5\} \times 10^{-4}$ for each of the four resolutions.

5. Conclusions

As shown in the above computational results, the fully implicit method from Section 3 results in an accurate and stable method. The fourth-order central difference spatial approximation provides a high level of accuracy for the problems from Section 4, which do not exhibit strong shocks or sharp solution interfaces. As a result of the highly accurate predicted solution and tuned implicit solver, the overall scheme reproduces the same level of solution accuracy as comparable explicit schemes, in significantly less time for these stiff MHD problems. Moreover, as demonstrated in the 3D pellet injection example from Section 4.3, this solution machinery is fully parallel and scales better with problem size than standard explicit-time approaches.

In addition to the efficiency benefits, this implicit approach retains the physicality of its solutions by satisfying the underlying conservation laws and retaining the solenoidal property of the magnetic field. Furthermore, the code correctly computes single fluid magnetic reconnection, for initial conditions specified by the GEM challenge problem, and satisfies the theoretical Sweet–Parker scaling of reconnection rate with respect to the Lundquist number.

We note that such implicit approaches are not ideal for all problems. For physical systems which exhibit strong shocks or gradients, special techniques must be used for computation of fluxes at cell faces; these techniques are often non-differentiable and as a result may pose convergence problems for implicit methods.

Moreover, as the implicit solver framework involves increased computational effort per time step over explicit methods, efficiency gains as shown in Section 4 will only be realized on sufficiently stiff problems, in which the relevant dynamic time scale is much larger than the time scale imposing the explicit stability restriction.

We further note that we did not use preconditioning techniques within the iterative linear solves in this work, though we still realized significant improvements over the explicit-time method. This improvement on the reconnection and pellet injection problems resulted from the time stepping approach, which can remain stable even while taking time steps larger than what the CFL condition would allow. Moreover, unlike implicit approaches that strive to take the largest time steps possible, placing all of the solution work on the nonlinear and linear solvers, this method guarantees a desired level of time integration accuracy while attempting to take the solution burden off of the nonlinear and linear methods by solving only as precisely as is needed to maintain the requested accuracy. We anticipate that incorporation of fast and effective preconditioning techniques will allow further efficiency improvements over explicit time methods as the meshes are refined, as evidenced in Table 2 by the increasing number of Krylov iterations per Newton iteration as the mesh size is decreased. The construction of such methods will be the subject of future work.

Acknowledgments

The authors acknowledge the insightful discussions with Luis Chacon, Steve Jardin, Alan Hindmarsh, and Phillip Colella during the course of this work. The parallel simulations were performed on the LLNL MCR linux cluster and at the National Energy Research Scientific Computing Center. This work was performed under the auspices of the US Department of Energy by the University of California, Lawrence Livermore National Laboratory under Contract W-7405-ENG-48 and was supported in part by the US Department of Energy under Contract DE-AC020-76-CH03073.

References

[1] R. Goldston, P. Rutherford, Introduction to Plasma Physics, Institute of Physics Publishing, Bristol and Philadelphia, 1997.
[2] W. Park, E.V. Belova, G.Y. Fu, X.Z. Tang, Plasma simulation studies using multilevel physics models, Phys. Plasmas 6 (1999) 1796–1803.
[3] C.R. Sovinec, A.H. Glasser, T.A. Gianakon, D.C. Barnes, R.A. Nebel, S.E. Kruger, D.D. Schnack, S.J. Plimpton, A. Tarditi, M. Chu, Nonlinear magnetohydrodynamics simulation using high-order finite elements, J. Comput. Phys. 195 (1) (2004) 355–386.
[4] R.D. Hazeltine, J. MeissPlasma Confinement, Frontiers in Physics, vol.86, Addison-Wesley, Redwood City, CA, 1992.
[5] W. Gropp, D. Keyes, L. McInnes, M. Tidriri, Globalized Newton–Krylov–Schwarz algorithms and software for parallel implicit CFD, Int. J. High Perform. Comput. Appl. 14 (2000) 102–136.
[6] P. Brown, D. Shumaker, C. Woodward, Fully implicit solution of large-scale non-equilibrium radiation diffusion with high order time integration, J. Comput. Phys. 204 (2) (2005) 760–783.
[7] D. Knoll, D. Keyes, Jacobian-free Newton–Krylov methods: a survey of approaches and applications, J. Comput. Phys. 193 (2004) 357–397.
[8] K. Powell, P. Roe, T. Linde, T. Gombosi, D. DeZeeuw, A solution-adaptive upwind scheme for ideal magnetohydrodynamics, J. Comput. Phys. 154 (1999) 284–300.
[9] T. Gardiner, J. Stone, An unsplit Godunov method for ideal MHD via constrained transport, J. Comput. Phys. 205 (2005) 509–539.
[10] R. Samtaney, Suppression of the Richtmyer–Meshkov instability in the presence of a magnetic field, Phys. Fluids 15 (8) (2003) L53–L56.
[11] R. Samtaney, P. Colella, T.J. Ligocki, D.F. Martin, S.C. Jardin, in: A. Mezzacappa et. al. (Ed.), SciDAC 2005, San Francsico, June 26–30, Journal of Physics: Conference Series, 2005.
[12] D.S. Harned, W. Kerner, Semi-implicit method for three-dimensional compressible magnetohydrodynamic simulation, J. Comput. Phys. 60 (1985) 62–75.
[13] D.S. Harned, W. Kerner, Semi-implicit method for long time scale magnetohydrodynamic computations in three dimensions, J. Comput. Phys. 65 (1986) 57–70.
[14] M. Tokman, P. Bellan, Three-dimensional model of the structure and evolution of coronal mass ejections, Astrophys. J. 567 (2002) 1202–1210.
[15] S.C. Jardin, J.A. Breslau, Implicit solution of the four-field extended magnetohydrodynamic equations using high-order high-continuity finite elements, Phys. Plasmas 12 (2005). Art. No. 056101.
[16] O.S. Jones, U. Shumlak, D.S. Eberhardt, An implicit scheme for nonideal magnetohydrodynamics, J. Comput. Phys. 130 (2) (1997) 231–242.
[17] L. Chacón, D. Knoll, J. Finn, An implicit, nonlinear reduced resistive MHD solver, J. Comput. Phys. 178 (2002) 15–36.
[18] M. Luskin, R. Rannacher, On the smoothing property of the Crank–Nicolson scheme, Appl. Anal. 14 (1982/1983) 117–135.

[19] R. Rannacher, Finite element solution of diffusion problems with irregular data, Numer. Math. 43 (1984) 309–327.
[20] L. Chacón, D. Knoll, A 2D high-β Hall MHD implicit nonlinear solver, J. Comput. Phys. 188 (2003) 573–592.
[21] T. Rognlien, X. Xu, A. Hindmarsh, Application of parallel implicit methods to edge-plasma numerical simulations, J. Comput. Phys. 175 (2002) 249–268.
[22] D. Hill, D. Pullin, Hybrid tuned center-difference-WENO method for large eddy simulations in the presence of strong shocks, J. Comput. Phys. 194 (2004) 435–450.
[23] A.C. Hindmarsh, P.N. Brown, K.E. Grant, S.L. Lee, R. Serban, D.E. Shumaker, C.S. Woodward, SUNDIALS: Suite of nonlinear and differential/algebraic equation solvers, ACM Trans. Math. Softw. 31 (3) (2005) 363–396.
[24] A. Hindmarsh, R. Serban, User documentation for CVODE v. 2.2.1, Tech. Rep., UCRL-SM-208108, LLNL, Dec 2004.
[25] G. Byrne, Pragmatic experiments with Krylov methods in the stiff ODE setting, in: J. Cash, I. Gladwell (Eds.), Computational Ordinary Differential Equations, Oxford University Press, Oxford, 1992, pp. 323–356.
[26] P. Brown, G. Byrne, A. Hindmarsh, VODE, a variable-coefficient ODE solver, SIAM J. Sci. Stat. Comput. 10 (1989) 1038–1051.
[27] K. Jackson, R. Sacks-Davis, An alternative implementation of variable step-size multistep formulas for stiff ODEs, ACM Trans. Math. Softw. 6 (1980) 295–318.
[28] G. Byrne, A. Hindmarsh, A polyalgorithm for the numerical solution of ordinary differential equations, ACM Trans. Math. Softw. 1 (1975) 71–96.
[29] P.N. Brown, Y. Saad, Hybrid Krylov methods for nonlinear systems of equations, SIAM J. Sci. Stat. Comput. 11 (1990) 450–481.
[30] P. Brown, A. Hindmarsh, Reduced storage matrix methods in stiff ODE systems, J. Appl. Math. Comp. 31 (1989) 49–91.
[31] J. Birn et al., Geospace Environmental Modeling (GEM) magnetic reconnection challenge, J. Geophys. Res. 106 (2001) 3715–3719.
[32] D. Biskamp, Magnetic Reconnection in Plasmas, 3rd ed., Cambridge Monographs on Plasma Physics, 2000.
[33] L. Baylor et al., Improved core fueling with high field pellet injection in the DIII-D tokamak, Phys. Plasmas 7 (2000) 1878–1885.
[34] L. Baylor, T.C. Jernigan, R.J. Colchin, G.L. Jackson, L.W. Owen, T.W. Petrie, Comparison of fueling efficiency from different fueling locations on DIII-D, J. Nucl. Mater. 313 (2003) 530–533.
[35] R. Samtaney, S.C. Jardin, P. Colella, D.F. Martin, 3D Adaptive mesh refinement simulations of pellet injection in tokamaks, Comput. Phys. Commun. 164 (2004) 220–228.
[36] P. Parks, R.J. Turnbull, Effect of transonic flow in ablation cloud on lifetime of a solid hydrogen pellet in a plasma, Phys. Fluids 21 (1978) 1735–1741.
[37] B.V. Kuteev, Hydrogen pellet ablation and acceleration by current in high temperature plasmas, Nucl. Fusion 35 (1995) 431–453.