JOURNAL OF COMPUTATIONAL PHYSICS **140,** 172–201 (1998)
ARTICLE NO. CP975841

Magnetohydrodynamics of Solar Coronal Plasmas in Cylindrical Geometry

Roberto Lionello,*,†,[1] Zoran Mikić,* and Dalton D. Schnack*

**Science Applications International Corporation, San Diego, California 92121-1578; †Department of Physics and Astronomy, University of California, Irvine, California 92697-4575*
E-mail: lionel@iris023.saic.com

Received November 18, 1996; revised August 16, 1997

We describe a three-dimensional algorithm for the advancement of the resistive MHD equations in cylindrical geometry with line-tied boundary conditions. This code has been developed to simulate the behavior of solar coronal plasmas. A finite-difference discretization is used for the radial and axial coordinates; a pseudospectral method is used for the azimuthal coordinate. The dependent variables are defined on finite-difference meshes that are staggered with respect to each other to facilitate the application of boundary conditions. The time-advance algorithm features a semi-implicit leapfrog scheme for the wave terms, a predictor–corrector treatment of advection, and an implicit advance of the resistive and viscous diffusion terms. The semi-implicit and implicit operators are inverted using a preconditioned conjugate gradient method. Special care is taken in maintaining the self-adjointness of the discretized operators, so that a fast inversion algorithm applicable to symmetric matrices can be used. By way of illustration, we describe the application of the code to the linear and nonlinear evolution of a kink instability in a twisted flux tube.

Key Words: partial differential equations; initial value and time-dependent initial-boundary value problems; numerical linear algebra; iterative methods for linear systems; astronomy and astrophysics; hydrodynamic and hydromagnetic problems; fluid mechanics; magnetohydrodynamics and electrohydrodynamics.

1. INTRODUCTION

The solar corona abounds with interesting phenomena of controversial physical interpretation. Although it is not understood why the corona is so hot (around 10^6 K) and what causes flares to occur, it is believed that magnetic reconnection plays a crucial role in determining

[1] Formerly at Dipartimento di Astronomia e Scienza dello Spazio, Università di Firenze, I-50125 Firenze, Italy.

0021-9991/98 $25.00

its structure and evolution. Loops of tenuous magnetized plasma are observed frequently in the corona. The ends of these coronal loops are anchored in the dense photosphere below, a situation that has been referred to as *line tying*. The slow motions in the photosphere drive the footpoints of the loops, which evolve through series of equilibria. Rapid evolution may occur when an unstable equilibrium is reached, possibly leading to the development of current sheets, magnetic reconnection, and the release of magnetic energy that may heat the corona. If such "disruptions" are sufficiently impulsive, they may be identified with solar flares. Recent observational and theoretical results on flares and coronal heating can be found in [1–4].

A complete description of these processes requires a three-dimensional model that includes the slow, long-wavelength evolution prior to disruption, as well as the rapid short-wavelength evolution in the nonlinear phase. The resistive magnetohydrodynamic (MHD) model is appropriate to describe much of the physics associated with these phenomena (except in places where the gradient scale-length is smaller than the gyroradius and a kinetic treatment must be adopted). For simplicity we will restrict our attention to geometries that are best described in a cylindrical coordinate system. When modeling coronal loops, we will therefore neglect the important effect of loop curvature [5], studying instead straight flux tubes as an approximation to large-aspect-ratio coronal loops.

Although the linear stability properties of cylindrical flux tubes have been studied analytically [6–9], a description of the nonlinear evolution requires a computational approach. Several cylindrical MHD codes, with axially periodic boundary conditions, have been used to model laboratory plasmas [10–14]. However, we cannot use such codes for our studies because we need to impose line-tied boundary conditions to properly model coronal loops, and a new algorithm must be developed for this purpose.

The goal of the present paper is to describe a fast, accurate, and reliable algorithm for the advancement of the full resistive and viscous MHD equations in cylindrical geometry which allows for the specification of driving photospheric motions at the magnetic footpoints. The code is an improved version of the algorithm employed in [15]. Quantities are evaluated on grids: the azimuthal variation (θ) is represented using Fourier series, with pseudospectral calculation of derivatives; the r and z coordinates are discretized on staggered meshes, which allows us to define a curl operator whose divergence vanishes identically. A leapfrog scheme is used for the time advancement of the wave terms. We employ a semi-implicit operator in the momentum equation, following the method described in [13], while treating advection with a predictor–corrector scheme. The semi-implicit scheme allows us to set the time step through considerations of accuracy rather than stability of the algorithm and leads to a substantial saving of CPU time, compared to a fully explicit algorithm.

The resistive and viscous diffusion terms are advanced implicitly. The resulting implicit equations and the semi-implicit operator are inverted using a preconditioned conjugate gradient method [16,17]. We have attempted to preserve many of the analytical properties of the MHD equations in the discretized equations. In particular, we have taken special care to preserve the self-adjointness of spatial difference operators. Since the proper differencing of a self-adjoint operator results in a symmetric matrix, we are therefore able to use the efficient methods that exist for inverting symmetric matrices. As an illustration of the properties of our algorithm, we describe its application to the linear and nonlinear evolution of a kink instability in a twisted flux tube.

The paper is organized as follows: in Section 2 we describe the MHD equations and the spatial and temporal approximations employed to advance them in time; Section 3 contains

a description of the implementation of a conjugate gradient algorithm for the inversion of the implicit spatial operators; in Section 4 we describe an application of the code, and in Section 5 we summarize our conclusions.

2. COMPUTATIONAL MODEL

Coronal loops consist of a hot, tenuous plasma embedded in a strong magnetic field. A most important feature of the loops is that their ends are firmly anchored in the dense photosphere. A sketch of a coronal loop is shown in Fig. 1.

The resistive MHD model is appropriate for our study of solar coronal plasmas. The MHD equations are written in cylindrical coordinates, neglecting for simplicity the curvature effect. Hence, loops in our analysis are "straightened out" as in Fig. 2. This is clearly an approximation and important effects are neglected in principle. This description is expected to be appropriate when the aspect ratio (i.e., the ratio between the radial and the axial length scales) is large. We write the MHD equations in a convenient nondimensional form as

$$\frac{\partial \mathbf{A}}{\partial t} = \mathbf{v} \times \mathbf{B} - \eta \boldsymbol{\nabla} \times \boldsymbol{\nabla} \times \mathbf{A}, \tag{1}$$

$$\frac{\partial \rho}{\partial t} = -\boldsymbol{\nabla} \cdot (\rho \mathbf{v}), \tag{2}$$

$$\frac{\partial p}{\partial t} = -\boldsymbol{\nabla} \cdot (p \mathbf{v}) - (\gamma - 1) p \boldsymbol{\nabla} \cdot \mathbf{v}, \tag{3}$$

$$\frac{\partial \mathbf{v}}{\partial t} = -\mathbf{v} \cdot \boldsymbol{\nabla} \mathbf{v} + \frac{\mathbf{J} \times \mathbf{B}}{\rho} - \frac{\boldsymbol{\nabla} p}{\rho} + \frac{\boldsymbol{\nabla} \cdot \nu \rho_0 \boldsymbol{\nabla} v}{\rho_0}, \tag{4}$$

where $\mathbf{A}$ is the vector potential of the magnetic field $\mathbf{B} = \boldsymbol{\nabla} \times \mathbf{A}$, $\mathbf{J} = \boldsymbol{\nabla} \times \mathbf{B}$ is the current density, $\mathbf{v}$ is the velocity, p the pressure, ρ the mass density, η the resistivity, and ν the viscosity.

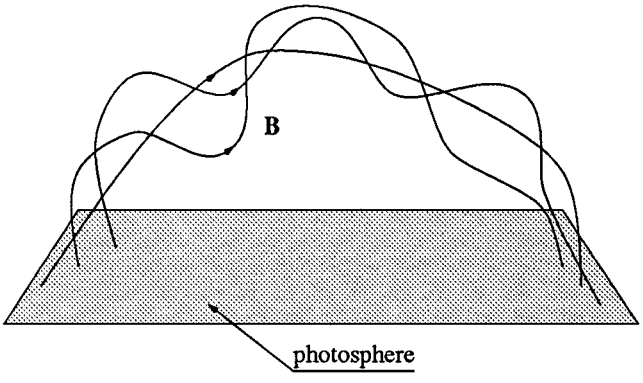

FIG. 1. A schematic representation of the magnetic field of a loop in the solar corona. Note that all the field lines are anchored in the photosphere at both ends. In our code we neglect the curvature and the loop appears "straightened out" (see Fig. 2).

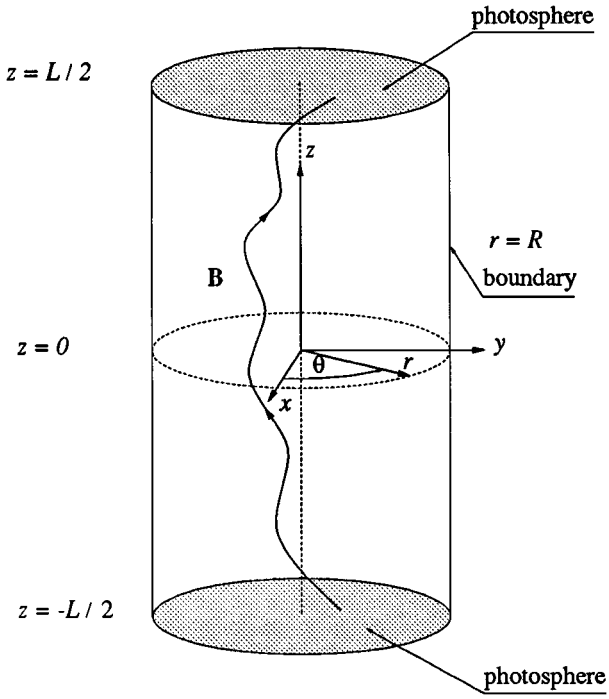

FIG. 2. Coordinates and boundaries for modeling coronal loops. Field lines are anchored in the photosphere at $z = \pm L/2$. The loops in the code are "straightened out," since we ignore the curvature observed in real loops (see Fig. 1).

The induction equation (1) allows us to select between ideal ($\eta = 0$) or resistive MHD ($\eta \neq 0$). An ideal run is possible only in particular conditions, for example, to study the linear phase of an instability. In general, the grid resolution dictates the minimum values of η and ν that may be used. For example, for a $64 \times 32 \times 64$ grid, we have found on a particular problem that they must be at least $\sim 10^{-3}$, and sometimes $\sim 10^{-2}$, for the solution to be physically valid.

When we discuss the zero-beta model, in which we assume that $p = 0$, we specify the density to be uniform and fixed in time, so that the mass continuity equation (2) is not solved. Similarly, we do not advance the energy equation (3) in the zero-beta model. Note that we neglect the influence of viscous and resistive heating, since we use an adiabatic energy equation. We plan to add the viscous and resistive heating terms, as well as thermal conduction, in future versions of the code.

The viscosity in the momentum equation (4) is mainly used to damp short-wavelength modes in the calculation. In this term we have used $\rho_0 = 1/(2\pi) \int \rho d\theta$, instead of ρ, to allow the matrix inversion to proceed mode by mode.

The equations describe the long-wavelength and long time-scale evolution of the corona, including magneto–acoustic waves, ideal and resistive instabilities, and resistive and viscous damping. However, particle acceleration and X-ray emission require kinetic models that are not part of the code.

In a coronal loop, the dense photosphere anchors the footpoints of the magnetic field lines, so that they are dragged by applied surface flows **V**. This footpoint shearing is modeled by

applying the condition at the boundaries $z = \pm L/2$,

$$\frac{\partial \mathbf{A}_t}{\partial t} = (\mathbf{V} \times \mathbf{B})_t, \tag{5}$$

$$\mathbf{v} = \mathbf{V}, \tag{6}$$

where the subscript t indicates the tangential component of vectors (the normal component of $\mathbf{A}$ is advanced as in Eq. (1)). Hence boundary conditions are specified only on the tangential electric field and normal magnetic field. Equations (5)–(6) are also valid at the radial boundary at $r = R$, where a conducting wall is present and $\mathbf{V} = \mathbf{0}$. The wall is placed far enough from the plasma not to affect the physics being studied.

In order to translate from nondimensional to physical quantities we have to specify three normalization variables for length, magnetic field, and density. For example, when modeling coronal loops, we can set $L_\circ = 10^8$ cm , $B_\circ = 10$ G, and $\rho_\circ = 10^{-15}$ g cm^{-3}. Then fields and scalars in Eqs. (1)–(4) can be measured in terms of

$$\begin{aligned}
A_\circ &\equiv B_\circ L_\circ && \text{G cm},\\
V_\circ &\equiv B_\circ (4\pi\rho_\circ)^{-1/2} && \text{cm s}^{-1},\\
J_\circ &\equiv c B_\circ (4\pi L_\circ)^{-1} && \text{statamp cm}^{-2},\\
P_\circ &\equiv B_\circ^2 (4\pi)^{-1} && \text{dyne cm}^{-2},\\
t_\circ &\equiv L_\circ B_\circ^{-1} (4\pi\rho_\circ)^{1/2} && \text{s},\\
\eta_\circ &\equiv (4\pi)^{1/2} c^{-2} \rho_\circ^{-1/2} B_\circ L_\circ && \text{s},\\
\nu_\circ &\equiv B_\circ L_\circ (4\pi\rho_\circ)^{-1/2} && \text{cm}^2\ \text{s}^{-1}.
\end{aligned}$$

In terms of this normalization, we have: the Alfvén velocity $v_\mathrm{A} = V_\circ \sim 9 \times 10^9$ cm s^{-1}, the Alfvén time $\tau_\mathrm{A} = L_\circ / v_\mathrm{A} \sim 1$ s, the mass scale $M = \rho_\circ L_\circ^3 = 1 \times 10^9$ g, etc.

2.1. *Spatial Approximation*

We use cylindrical coordinates (r, θ, z), with $0 \le r \le R$, $0 \le \theta \le 2\pi$, $-L/2 \le z \le L/2$, to model large aspect-ratio coronal loops. A sketch of the coordinate system is presented in Fig. 2. The θ coordinate is periodic, so we introduce a discrete mesh $\theta_j = 2\pi(j-1)/M$, $j = 1, 2, \ldots, M$, and write any field f as a finite Fourier series,

$$f(r, \theta_j, z) = \sum_{m=-M/2+1}^{M/2+1} f_m(r, z) e^{im\theta_j}. \tag{7}$$

It is well known that the discrete Fourier series converges rapidly if the solution is smooth [18]. Furthermore, time advancement in Fourier space is facilitated because the poloidal (m) modes for linear operators decouple. Hence, implicit terms, which are present in the induction and momentum equations and must be inverted, will be represented as distinct small submatrices (one for each Fourier mode), instead of a single large matrix.

The complex coefficients f_m are given by

$$f_m(r, z) = \frac{1}{M} \sum_{j=1}^{M} f(r, \theta_j, z) e^{-im\theta_j}. \tag{8}$$

The reality of f requires that $f_m = f^*_{-m}$, where f^*_m is the complex conjugate of f_m. Moreover, f_0 and $f_{m/2+1}$ have zero imaginary parts. We apply (8) to the MHD equations (1)–(4), obtaining a set of M nonlinear partial differential equations in the variables (r, z, t) describing the evolution of the Fourier components of $\mathbf{A}$, $\mathbf{v}$, ρ, and p.

We evaluate the nonlinear terms in Eqs. (1)–(4) with a fully dealiased pseudospectral algorithm, as described in [20]. The pseudospectral method consists of computing operations either in Fourier space or in real space, according to where it is more convenient. Thus multiplication is performed in real space to avoid convolution, and derivatives in θ are evaluated in Fourier space. We use a fast Fourier transform to transform between the two representations. However, multiplication generates aliasing errors, due to quadratic and higher nonlinearities. Hence, we truncate the θ-spectrum (dealiasing) and retain only two-thirds of available Fourier space.

In order to simplify an implicit treatment, we assume that η and ν do not depend on θ. This choice makes the implicit viscous and resistive operators linear in θ, and, consequently, poloidal modes decouple in Fourier space.

We choose two staggered meshes for each nonperiodic direction, r and z. Beside being second-order accurate in calculating derivatives (when uniform meshes are specified), in this method boundary conditions are specified naturally: for the magnetic field only the normal component is specified, while the tangential one is computed, and for the electric field the tangential component is specified, while the normal component is computed. Moreover, the algorithm has the property that the longitudinal and transverse parts of vectors are effectively decoupled, so that initially vanishing longitudinal and transverse components will vanish all the time. A consequence of this is that $\boldsymbol{\nabla} \cdot \mathbf{B} = 0$.

Current sheets may form during the nonlinear phase of instabilities in our simulations. We therefore allow the mesh points in the radial direction to have nonuniform spacing in order to have locally enhanced resolution in the proximity of the center of the loop. The axial mesh is normally (but not necessarily) uniformly spaced. Radial mesh points on the integer mesh are indicated with $(r_i, i = 1, I-1)$, where $r_1 = 0$ and $r_{I-1} = R$. On the half-integer mesh we write $(r_{h;i}, i = 1, I)$. The relationship between the two set of mesh points is $r_{h;i} = (r_i + r_{i-1})/2$. We define also the finite increments $(dr_{h;i} = r_i - r_{i-1}, i = 2, I-1)$ and $(dr_i = r_{h;i+1} - r_{h;i}, i = 1, I-1)$. In the axial direction we define $(z_j, j = 1, J-1)$, with $z_1 = -L/2$ and $z_{J-1} = L/2$, with analogous definitions for $z_{h;j}$, dz_j, and $dz_{h;j}$. Figures 3, 4, 5, and 6 show how the dependent variables are defined on each mesh.

Derivatives are defined on integer or half-integer meshes according to Table 1. With these definitions the gradient, divergence, and curl operator can be implemented so that the

TABLE 1
Differential Operators and Their Corresponding Finite Difference Representations

Operator	Integer mesh	Half-integer mesh
$\frac{1}{r}\frac{\partial}{\partial\theta}$	$i\frac{m}{r_i}C_{i,j}$	$i\frac{m}{r_{h;i}}D_{i,j}$
$\frac{\partial}{\partial r}$	$\frac{E_{i,j} - E_{i-1,j}}{dr_{h;i}}$	$\frac{F_{i+1,j} - F_{i,j}}{dr_i}$
$\frac{\partial}{\partial z}$	$\frac{G_{i,j} - G_{i,j-1}}{dz_{h;j}}$	$\frac{H_{i,j+1} - H_{i,j}}{dz_j}$

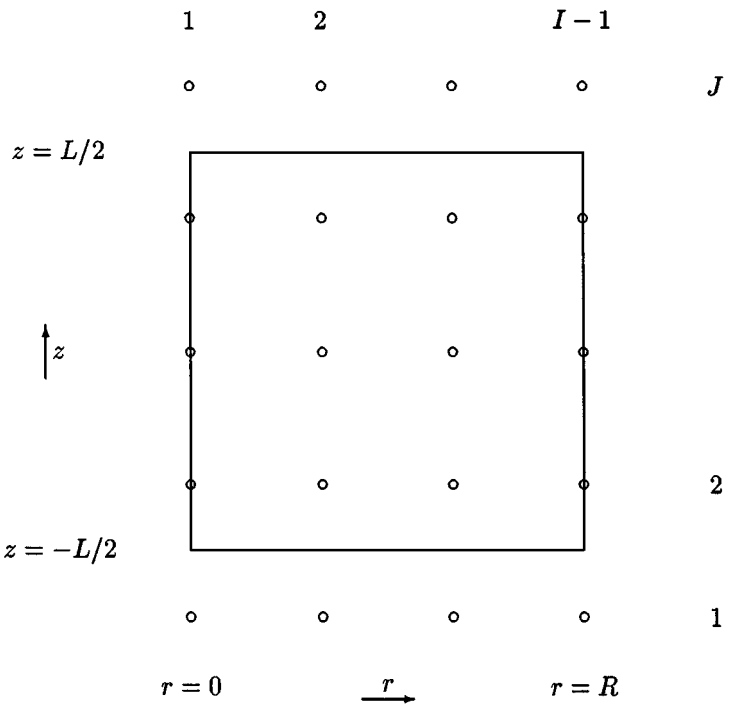

FIG. 3. Mesh for v_r, A_r, B_z, and J_r. The square represents the physical domain in r and z. Mesh points are indicated with ○.

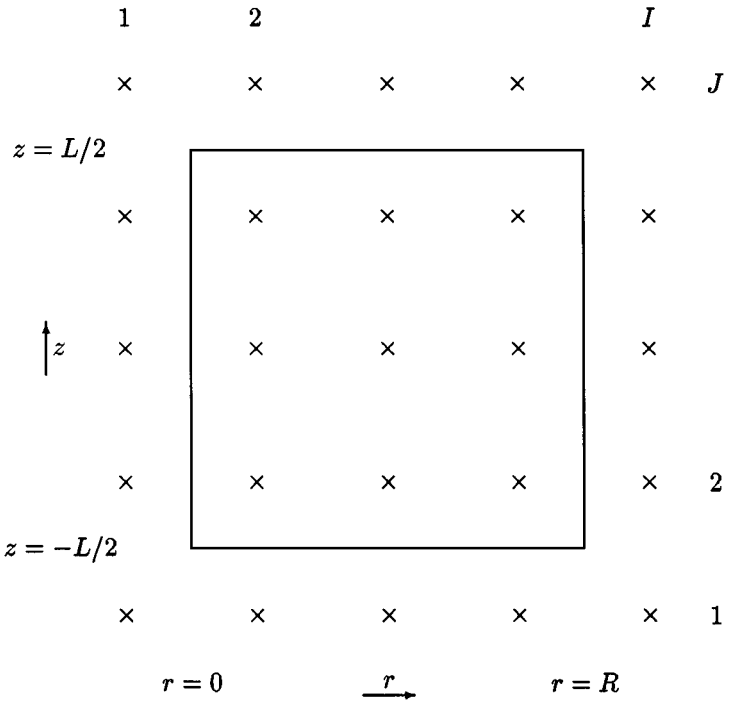

FIG. 4. Mesh for v_θ, A_θ, J_θ, ρ, and p. The square represents the physical domain in r and z. Mesh points are indicated with ×.

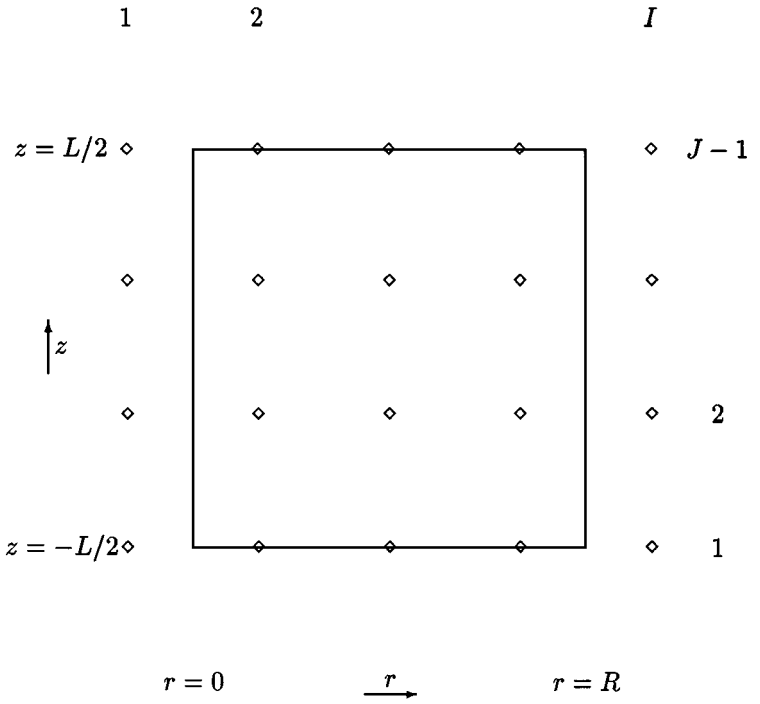

FIG. 5. Mesh for v_z, A_z, B_r, and J_z. The square represents the physical domain in r and z. Mesh points are indicated with $\diamond$.

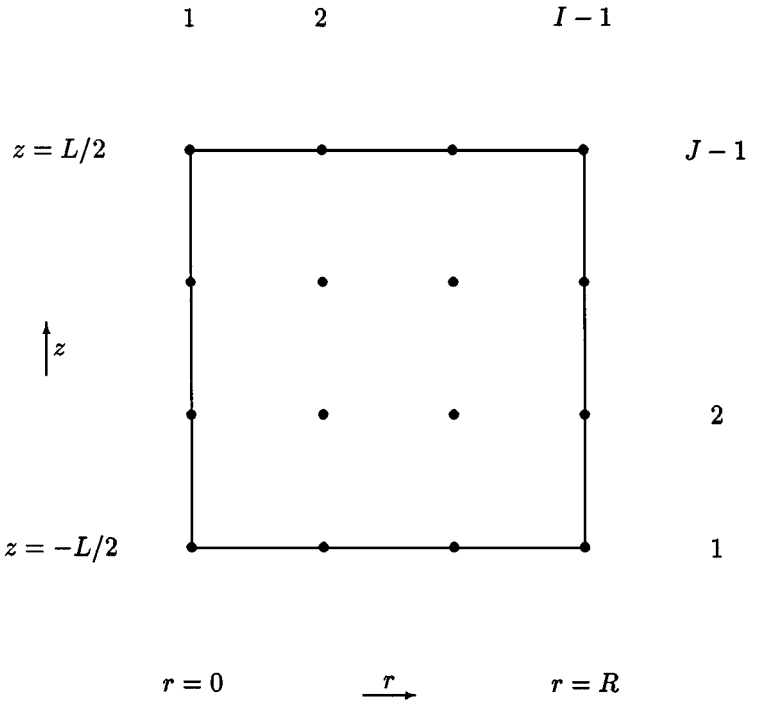

FIG. 6. Mesh for B_θ, v, and η. The square represents the physical domain in r and z. Mesh points are indicated with $\bullet$.

divergence of a curl and the curl of a gradient vanish identically. The components of **B** and **J** are naturally specified on integer or half-integer meshes according to their definition. The components of the nonlinear terms in Eqs. (1)–(4) are evaluated on the same grid of the field components on the left-hand side, using simple averaging where necessary.

Physical boundary conditions for **A** and **v** are specified at $z = \pm L/2$ and $r = R$ according to Eqs. (5)–(6).

Boundary values for p and ρ are not required to be specified in our formulation, but they can be evaluated for diagnostic purposes using extrapolation. At $r = 0$ we apply geometric boundary conditions as shown in Appendix A. It is not completely straightforward to implement such boundary conditions, as they tend to spoil the symmetry properties of the operators we have to invert. See Section 3 for further discussion.

2.2. *Temporal Approximation*

The right-hand sides of Eqs. (1)–(4) have advective, dissipative, and wave-like terms that are treated using predictor–corrector, implicit, and semi-implicit methods. We introduce a leapfrog time discretization for the various fields, defining **A** (together with ρ and p) and **V** at staggered time intervals. The resulting algorithm is

$$\frac{\mathbf{A}^* - \mathbf{A}^{n-1/2}}{\Delta t} = \mathbf{v}^n \times \mathbf{B}^{n-1/2}, \qquad \text{P} \tag{9}$$

$$\begin{aligned}\frac{\mathbf{A}^{n+1/2} - \mathbf{A}^{n-1/2}}{\Delta t} &= \mathbf{v}^n \times \mathbf{B}^* \qquad \text{C}\\ &\quad - \eta \frac{\boldsymbol{\nabla} \times \boldsymbol{\nabla} \times \mathbf{A}^{n+1/2}}{2}\\ &\quad - \eta \frac{\boldsymbol{\nabla} \times \boldsymbol{\nabla} \times \mathbf{A}^{n-1/2}}{2},\end{aligned} \tag{10}$$

$$\frac{\rho^* - \rho^{n-1/2}}{\Delta t} = -\boldsymbol{\nabla} \cdot (\rho^{n-1/2} \mathbf{v}^n), \qquad \text{P} \tag{11}$$

$$\frac{\rho^{n+1/2} - \rho^{n-1/2}}{\Delta t} = -\boldsymbol{\nabla} \cdot (\rho^* \mathbf{v}^n), \qquad \text{C} \tag{12}$$

$$\frac{p^* - p^{n-1/2}}{\Delta t} = -\boldsymbol{\nabla} \cdot (p^{n-1/2} \mathbf{v}^n), \qquad \text{P} \tag{13}$$

$$\begin{aligned}\frac{p^{n+1/2} - p^{n-1/2}}{\Delta t} &= -\boldsymbol{\nabla} \cdot (p^* \mathbf{v}^n) \qquad \text{C}\\ &\quad -(\gamma - 1) p^{n-1/2} \boldsymbol{\nabla} \cdot \mathbf{v}^n,\end{aligned} \tag{14}$$

$$\frac{\mathbf{v}^* - \mathbf{v}^n}{\Delta t} = -\mathbf{v}^n \cdot \boldsymbol{\nabla} \mathbf{v}^n, \qquad \text{P} \tag{15}$$

$$\begin{aligned}\frac{\mathbf{v}^{**} - \mathbf{v}^n}{\Delta t} &= -\mathbf{v}^n \cdot \boldsymbol{\nabla} \mathbf{v}^* \qquad \text{C}\\ &\quad + \frac{\mathbf{J}^{n+1/2} \times \mathbf{B}^{n+1/2}}{\rho^{n+1/2}} - \frac{\boldsymbol{\nabla} p^{n+1/2}}{\rho^{n+1/2}}\\ &\quad + \frac{\boldsymbol{\nabla} \cdot C^2 \Delta t^2 \rho_0^{n+1/2} \boldsymbol{\nabla} (\mathbf{v}^{**} - \mathbf{v}^n)}{\boldsymbol{\Delta} t \rho_0^{n+1/2}},\end{aligned} \tag{16}$$

$$\frac{\mathbf{v}^{n+1} - \mathbf{v}^{**}}{\Delta t} = \frac{\boldsymbol{\nabla} \nu \rho_0^{n+1/2} \boldsymbol{\nabla} \mathbf{v}^{n+1}}{\rho_0^{n+1/2}}. \tag{17}$$

We have marked with P's and C's respectively the predictor and corrector steps in the equations for $\mathbf{A}$, ρ, p, and $\mathbf{v}$. Quantities marked with a $*$ or a $**$ index are provisional values needed in the predictor–corrector schemes and for a fully implicit treatment of the viscous term in Eq. (17).

The constant C^2 is the semi-implicit coefficient. The semi-implicit method is described in [13]. This method is unconditionally stable with respect to all magneto–acoustic and shear Alfvén modes. Hence, accuracy becomes the most relevant consideration in the choice of the time step. Briefly, the method consists of adding to the original momentum equation (4) a linear term multiplied by a coefficient proportional to the time step,

$$\frac{1}{\rho_0} \boldsymbol{\nabla} \cdot C^2 \Delta t^2 \rho_0 \boldsymbol{\nabla} \frac{\partial \mathbf{v}}{\partial t}. \tag{18}$$

This removes the small time-step restriction originally introduced by the wave term. Inverting the linear operator above is much less complex and requires less computer memory than using a fully implicit scheme. The advective terms in (1)–(4) are formally only first-order accurate in Δt, while the wave-like terms are second-order accurate (centered). The use of the semi-implicit method for the wave terms leaves only the stability condition

$$|(kV)_{\max} \Delta t| < 1, \tag{19}$$

due to the explicit treatment of advection. The quantity k is the magnitude of the largest wave vector compatible with the grid size at the point (r_i, z_j, θ_k),

$$k = \sqrt{k_\theta^2 + k_r^2 + k_z^2} = \sqrt{\left(\frac{M}{3r_i}\right)^2 + \left(\frac{1}{dr_i}\right)^2 + \left(\frac{1}{dz_j}\right)^2}. \tag{20}$$

Note the presence of the factor "3" in the expression for k_θ, due to the dealiasing algorithm which restricts the largest poloidal mode to $M/3$.

In order to address the stability limits imposed by the advective terms in Eqs. (1)–(4) and to give a heuristic justification of Eqs. (19), (20), we present a one-dimensional Von Neumann stability analysis of the advection part of the algorithm in Appendix B.

A stability analysis of our algorithm indicates that the wave-like terms are stable for any choice of time step, and the advective terms are stable when Eq. (19) is satisfied. However, we have recently found that the coupling of the leapfrog advance of the wave-like terms with a predictor–corrector for the advective terms may introduce numerical instability. This instability does not develop when there is sufficient viscosity in the algorithm. The calculations we describe in Section 4 have sufficient viscosity to prevent this numerical instability from occurring. We have analyzed this coupling, and we have devised an algorithm that does not suffer from this instability [19]. The fully implicit differencing of diffusive terms in Eqs. (1) and (4) does not introduce any stability limitation in the time step.

3. SELF-ADJOINT REPRESENTATION OF THE DIFFUSIVE AND SEMI-IMPLICIT TERMS

The differential operators in the MHD equations have the property of being self-adjoint. Let us concentrate only on the diffusive terms that appear in Eq. (1) (resistive diffusion operator) and in Eq. (4) (semi-implicit and viscous). When these equations are advanced in time implicitly the problem requires solving the algebraic equation

$$Ax = b, \tag{21}$$

where A is the coefficient matrix, x is the unknown vector, and b is the known term. In our case the dimension of the matrix A is $3IJ \times 3IJ$ at worst, when the equations for the three vector components are coupled.

We shall show that it is possible to write for all the above-cited operators a matrix A that is self-adjoint and positive-definite. This preserves an important property of the analytical equation and has also a desirable numerical advantage; we can apply the conjugate gradient (CG) algorithm to rapidly compute the solution x, instead of more complicated and general methods. The theory of the CG method is given in [16], and an application to a problem similar to ours is in [17]. Briefly, the CG method is an iterative algorithm to find the solution vector of the linear system (21) through successive approximations. It involves the matrix A only in the context of matrix–vector multiplication. Differently from other iterative methods, estimates of the largest and smallest eigenvalues of the iteration matrix are not needed. However rapid convergence occurs when the ratio between the maximum and minimum eigenvalues of A (known as the condition number) is small. Since our matrix is diagonally dominant, we apply diagonal preconditioning and obtain a matrix with a smaller condition number. The techniques described in this section make the code about 10 times faster than its previous version in [15], which uses the biconjugate gradient method [17].

3.1. *One-Dimensional Model*

We present now a discussion about how to implement a self-adjoint representation of a diffusive operator in one dimension. We consider the following diffusion equation:

$$\frac{\partial \mathbf{\Phi}}{\partial t} = \frac{\partial}{\partial x}\left(\beta \frac{\partial \mathbf{\Phi}}{\partial x}\right) \equiv D_\beta \mathbf{\Phi}; \tag{22}$$

x is assumed to vary between 0 and L. We want to solve the equation above numerically. First we fix two staggered meshes as in Fig. 7. $\mathbf{\Phi}$ is defined on the half-integer mesh (marked with $\times$'s), while β lies on the integer mesh (marked with $\circ$'s). The grid points need not be uniform. A general finite difference method for solving Eq. (22) is

$$\frac{\mathbf{\Phi}^{n+1} - \mathbf{\Phi}^n}{\Delta t} = \omega \mathbf{D}_\beta \cdot \mathbf{\Phi}^{n+1} + (1-\omega)\mathbf{D}_\beta \cdot \mathbf{\Phi}^n. \tag{23}$$

Operators and vectors are in bold when we refer to them as a whole; when we consider their components we write them in normal type. The value of ω can be any number between 0 (fully explicit) and 1 (fully implicit). $\omega = \frac{1}{2}$ corresponds to a (centered) second-order accurate in Δt time discretization. We rewrite the previous equation in components as

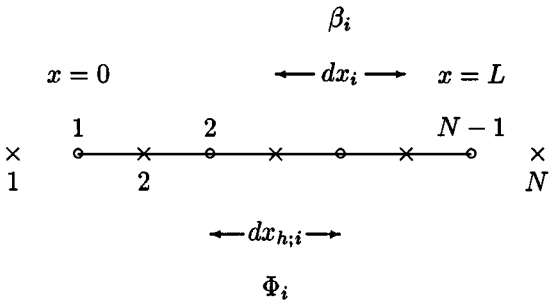

FIG. 7. Mesh used to represent Eq. (22). The physical region is between 0 and L. Φ lies on the mesh whose points are marked with $\times$; β lies on the one indicated with $\circ$'s. dx_i and $dx_{h;i}$ are the distances between neighbor points for each mesh.

$$\Phi_i^{n+1} - \omega\Delta t \frac{\beta_i \frac{\Phi_{i+1}^{n+1} - \Phi_i^{n+1}}{dx_i} - \beta_{i-1} \frac{\Phi_i^{n+1} - \Phi_{i-1}^{n+1}}{dx_{i-1}}}{dx_{h;i}}$$
$$= \Phi_i^n + (1-\omega)\Delta t \frac{\beta_i \frac{\Phi_{i+1}^{n} - \Phi_i^{n}}{dx_i} - \beta_{i-1} \frac{\Phi_i^{n} - \Phi_{i-1}^{n}}{dx_{i-1}}}{dx_{h;i}}, \tag{24}$$

where the distance between two mesh points on the half-integer and on integer mesh are indicated with dx_i and $dx_{h;i}$, respectively. If we write Eq. (24) in matrix form as it stands, we find that the matrix A is not symmetric. Hence, we cannot apply the CG algorithm (actually the matrix is tridiagonal, and we might use a fast *ad hoc* direct solver for such cases. However it loses this property when we increase the number of dimensions).

We know that for functions that are zero at the boundaries the following equality holds:

$$\int_0^L X D_\beta \Phi dx = \int_0^L \Phi D_\beta X dx. \tag{25}$$

We can write the numerical representation of Eq. (25) as a product between matrices and vectors,

$$\mathbf{X} \cdot \mathbf{dxD}_\beta \cdot \mathbf{\Phi} = \mathbf{\Phi} \cdot \mathbf{dxD}_\beta \cdot \boldsymbol{X}. \tag{26}$$

Here **dx** is a diagonal matrix whose elements are $dx_{h;i}$. With the discretization given by (24), we find that $\mathbf{dxD}_\beta$ is a self-adjoint matrix. When β is positive, the matrix is also positive definite.

Therefore, if we multiply both sides of Eq. (24) by $dx_{h;i}$, we obtain

$$(\mathbf{M} \cdot \mathbf{\Phi})_i \equiv \mathcal{M}_{0;i}\Phi_i + \mathcal{M}_{+;i}\Phi_{i+1} + \mathcal{M}_{-;i}\Phi_{i-1} = S_i; \tag{27}$$
$$\mathcal{M}_{0;i} = dx_{h;i} + \omega\Delta t \left(\frac{\beta_i}{dx_i} + \frac{\beta_{i-1}}{dx_{i-1}} \right);$$
$$\mathcal{M}_{+;i} = -\omega\Delta t \frac{\beta_i}{dx_i};$$
$$\mathcal{M}_{-;i} = \mathcal{M}_{+;i-1}.$$

Note that in this form the matrix $\mathbf{M}$ is explicitly symmetric. Here we have dropped the temporal index and have indicated the known term just with S_i. In computing $(\mathbf{M}\cdot\mathbf{\Psi})_i$ (where $\mathbf{\Psi}_i$ is either the guess solution or a temporary vector used by the CG algorithm), the index i may vary only in the range $2 \le i \le N-1$. When we know how to fix the proper boundary conditions, without spoiling the self-adjoint nature of the operator, we shall be able to apply the CG algorithm.

Boundary conditions are generally *Dirichlet conditions* (values specified at boundary points) or *Neumann conditions* (normal gradients at the boundaries). But even a combination of both is possible if we write them as

$$\begin{aligned} \mathbf{\Psi}_1 + \mathcal{C}_0\mathbf{\Psi}_2 &= \mathcal{V}_0 \quad \text{for } x = 0, \\ \mathbf{\Psi}_N + \mathcal{C}_L\mathbf{\Psi}_{N-1} &= \mathcal{V}_L \quad \text{for } x = L. \end{aligned} \tag{28}$$

Where $\mathcal{V}_{0,L}$ and $\mathcal{C}_{0,L}$ are constants whose values determine whether Dirichlet ($\mathcal{C}=0$) or Neumann ($\mathcal{C}=-1$) conditions apply. A proof that the matrix $\mathbf{M}$ is positive definite, with constrains as in Eq. (28), is given in Appendix C.

The way we implement the boundary conditions in our algorithm depends on how we intend to calculate the matrix–vector product at each iteration. We shall examine two methods: the first one modifies the matrix itself, the second one modifies the vector $\mathbf{\Psi}_i$.

To implement the first scheme we store the main diagonal $\mathcal{M}_{0;i}$ and one offset diagonal $\mathcal{M}_{+;i}$ of the matrix. The diagonal values and the right-hand side S_i must be modified according to Eq. (28) when $i=2$ and when $i=N-1$ as

$$\begin{aligned} \tilde{\mathcal{M}}_{0;2} &= \mathcal{M}_{0;2} - \mathcal{C}_0\mathcal{M}_{+;1}, \\ \tilde{S}_2 &= S_2 - \mathcal{V}_0\mathcal{M}_{+;1}; \\ \tilde{\mathcal{M}}_{0;N-1} &= \mathcal{M}_{0;N-1} - \mathcal{C}_L\mathcal{M}_{+;N}, \\ \tilde{S}_{N-1} &= S_{N-1} - \mathcal{V}_L\mathcal{M}_{+;N}. \end{aligned} \tag{29}$$

For $3 \le i \le N-2$, $\tilde{S}_i = S_i$, and $\tilde{\mathcal{M}}_{0;i} = \mathcal{M}_{0;i}$. Now only values of $\mathbf{\Psi}_i$ with $2 \le i \le N-1$ enter into the calculation of $\mathbf{M}\cdot\mathbf{\Psi}$ and $\mathbf{M}$ is a symmetric positive definite matrix. After the CG algorithm iterations, when we have found the solution $\mathbf{\Phi}_i$ for the internal points, we set $\mathbf{\Phi}_1$ and $\mathbf{\Phi}_N$ according to Eq. (28).

However, the previous method might not work when we deal with 2D or 3D problems and more complicated operators such as "curl–curl." Then it is possible that boundary conditions couple two different components of a vector field. In those cases, when we cannot write the modified diagonals for the self-adjoint matrix as in (29), we rely on the following method that has the advantage that we do not need to explicitly write the diagonals.

We have implemented a subroutine to set the boundary points of the vector $\mathbf{\Psi}$ according to Eq. (28). Another subroutine calculates $(\mathbf{M}\cdot\mathbf{\Psi})_i$, receiving an N component vector in input and yielding an $N-2$ component vector in output. We split the calculation of the solution of Eq. (27) into two parts. First, we calculate the "inhomogeneous part," fixing the boundary condition for the right-hand side. We take finite $\mathcal{V}$ terms and use $\mathbf{\Psi}^{(b)}$ as a work-array. The steps we perform are:

1. Set $\mathbf{\Psi}^{(b)} = \mathbf{0}$.
2. Set the boundary points on $\mathbf{\Psi}^{(b)}$. Since internal points are all zero we obtain $\mathbf{\Psi}^{(b)}_1 = \mathcal{V}_0$ and $\mathbf{\Psi}^{(b)}_N = \mathcal{V}_L$.

3. Get $(\mathbf{M}\cdot\mathbf{\Psi}^{(b)})_i$, $i=2, N-1$. The only nonzero components are $(\mathbf{M}\cdot\mathbf{\Psi}^{(b)})_2 = \mathcal{M}_{+;1}\mathcal{V}_0$ and $(\mathbf{M}\cdot\mathbf{\Psi}^{(b)})_{N-1} = \mathcal{M}_{+;N}\mathcal{V}_L$.

4. Find boundary contributions to the right-hand side of (27), $\tilde{S}_i = S_i - (\mathbf{M}\cdot\mathbf{\Psi}^{(b)})_i$, $i = 2, N-1$.

During the main iteration loop we calculate the "homogeneous part" of the solution, setting $\mathcal{V}=0$. At each iteration of the CG solver the matrix–vector product $(\mathbf{M}\cdot\mathbf{\Psi})_i$ is evaluated in two steps:

1. Set the boundary points on $\mathbf{\Psi}$. Since $\mathcal{V}=0$, $\mathbf{\Psi}_1 = -\mathcal{C}_0\mathbf{\Psi}_2$, and $\mathbf{\Psi}_N = -\mathcal{C}_L\mathbf{\Psi}_{N-1}$.
2. Get $(\mathbf{M}\cdot\mathbf{\Psi})_i$, $i=2, N-1$.

After the CG algorithm iterations, when the solution $\mathbf{\Phi}_i$ is retrieved, its values for $i=1$ and $i=N$ are easily deduced from Eq. (28).

3.2. *Viscosity Equation in Cylindrical Coordinates*

In order to solve Eq. (4) the viscosity and semi-implicit operators must be inverted. If we write the finite difference equivalents of Eqs. (16)–(17) as we have the numerical counterpart for the one-dimensional model in Eq. (24), we obtain matrices that are no longer self-adjoint as the analytical operators and require a slow algorithm for their inversion. However, it is possible to implement a general numerical scheme for inverting both operators. We shall represent them as symmetric and positive-definite matrices, using Subsection 3.1 as a guideline. We write

$$(\rho\, dV - \Delta t\omega_1\, dV\boldsymbol{\nabla}\cdot\alpha\boldsymbol{\nabla})\mathbf{v}^{(n+1)} = [\rho\, dV + \Delta t\omega_2\, dV\boldsymbol{\nabla}\cdot\alpha\boldsymbol{\nabla}]\mathbf{v}^{(n)}. \tag{30}$$

For the viscosity operator α is $\nu\rho$ (we write ρ instead of ρ_0 to avoid subscript overloading). Since we use a fully implicit advancement to ensure an efficient damping of the small unresolved scales, we take $\omega_1 = 1$ and $\omega_2 = 0$. For the semi-implicit algorithm $\alpha = C^2\Delta t\rho_0$ and $\omega_1 = 1$, $\omega_2 = -1$. Note that it is necessary to multiply both sides of the equation by the element of volume dV in order to obtain a self-adjoint matrix. This enables us to use the CG algorithm. The differential operator $dV\boldsymbol{\nabla}\cdot\alpha\boldsymbol{\nabla}$ must be represented in a discrete form without spoiling the symmetry of the matrix. Thus, we first write this operator in cylindrical coordinates in the following form: for the r component it is

$$\begin{aligned}\boldsymbol{\nabla}\cdot\alpha\boldsymbol{\nabla}\mathbf{v}|_r = {}& \frac{\partial}{\partial r}\frac{\alpha}{r}\frac{\partial}{\partial r}(r v_r) - \frac{\partial\alpha}{\partial r}\frac{v_r}{r} - \frac{m^2\alpha}{r^2}v_r + \frac{im}{2}\left(\frac{\partial}{\partial r}\left(\frac{\alpha v_\theta}{r}\right) - \frac{1}{r^2}\frac{\partial}{\partial r}(r\alpha v_\theta)\right)\\ & + \frac{im}{2}\left(\alpha\frac{\partial}{\partial r}\left(\frac{v_\theta}{r}\right) - \frac{\alpha}{r^2}\frac{\partial}{\partial r}(v_\theta r)\right) + \frac{\partial}{\partial z}\alpha\frac{\partial v_r}{\partial z}.\end{aligned} \tag{31}$$

For the θ component it is

$$\begin{aligned}\boldsymbol{\nabla}\cdot\alpha\boldsymbol{\nabla}\mathbf{v}|_\theta = {}& \frac{1}{r}\frac{\partial}{\partial r}r\alpha\frac{\partial v_\theta}{\partial r} - (1+m^2)\alpha\frac{v_\theta}{r^2} - \frac{im}{2}\left(\boxed{\frac{\partial}{\partial r}\left(\frac{\alpha v_r}{r}\right)} - \frac{1}{r^2}\frac{\partial}{\partial r}(r\alpha v_r)\right)\\ & - \frac{im}{2}\left(\boxed{\alpha\frac{\partial}{\partial r}\left(\frac{v_r}{r}\right)} - \frac{\alpha}{r^2}\frac{\partial}{\partial r}(v_r r)\right) + \frac{\partial}{\partial z}\alpha\frac{\partial v_\theta}{\partial z}.\end{aligned} \tag{32}$$

And, finally, for the z component we obtain

$$\boldsymbol{\nabla}\cdot\alpha\boldsymbol{\nabla}\mathbf{v}|_z = \frac{1}{r}\frac{\partial}{\partial r}r\alpha\frac{\partial v_z}{\partial r} - m^2\alpha\frac{v_z}{r^2} + \frac{\partial}{\partial z}\alpha\frac{\partial v_z}{\partial z}. \tag{33}$$

We choose as a natural grid for α the same as that used for ρ_0. This grid is shown in Fig. 4. When interpolated values are needed on other grids we put a $\hat{}$ or a $\bar{}$ over the symbol to indicate respectively interpolation along r or z. We can now write the numerical representation of Eq. (30). We shall drop the temporal index for brevity. The right-hand side is indicated with just an S_i, but its formula is analogous to the left-hand side, the only differences being the use of the old values of $\mathbf{v}$ and the factor $-\omega_2$, instead of ω_1. The equation for the v_r component is

$$\begin{aligned}
&\left(r_i dr_i dz_{h;j}\hat{\rho}_{i,j} + \omega_1\Delta t\frac{r_i^2 dz_{h;j}\alpha_{i+1,j}}{r_{h;i+1}dr_{h;i+1}} + \omega_1\Delta t\frac{r_i^2 dz_{h;j}\alpha_{i,j}}{r_{h;i}dr_{h;i}} + \omega_1\Delta t\alpha_{i+1,j}dz_{h;j}\right.\\
&\quad\left. - \omega_1\Delta t\alpha_{i,j}dz_{h;j} + \omega_1\Delta t m^2\frac{\hat{\alpha}_{i,j}dr_i dz_{h;j}}{r_i} + \omega_1\Delta t\frac{r_i dr_i\hat{\bar{\alpha}}_{i,j}}{dz_j} + \omega_1\Delta t\frac{r_i dr_i\hat{\bar{\alpha}}_{i,j-1}}{dz_{j-1}}\right)v_{r;i,j}\\
&\quad - \omega_1\Delta t\frac{r_{i+1}r_i dz_{h;j}\alpha_{i+1,j}}{r_{h;i+1}dr_{h;i+1}}v_{r;i+1,j} - \omega_1\Delta t\frac{r_{i-1}r_i dz_{h;j}\alpha_{i,j}}{r_{h;i}dr_{h;i}}v_{r;i-1,j}\\
&\quad - \omega_1\Delta t\frac{r_i dr_i\hat{\bar{\alpha}}_{i,j}}{dz_j}v_{r;i,j+1} - \omega_1\Delta t\frac{r_i dr_i\hat{\bar{\alpha}}_{i,j-1}}{dz_{j-1}}v_{r;i,j-1}\\
&\quad - i\omega_1\Delta t\frac{m}{2}(\alpha_{i+1,j} + \hat{\alpha}_{i,j})\left(\frac{r_i}{r_{h;i+1}} - \frac{r_{h;i+1}}{r_i}\right)v_{\theta;i+1,j}\\
&\quad + i\omega_1\Delta t\frac{m}{2}(\alpha_{i,j} + \hat{\alpha}_{i,j})\left(\frac{r_i}{r_{h;i}} - \frac{r_{h;i}}{r_i}\right)v_{\theta;i,j} = S_{r;i,j}.
\end{aligned} \tag{34}$$

The one for the v_θ component is

$$\begin{aligned}
&\left(r_{h;i}dr_{h;i}dz_{h;j}\rho_{i,j} + \omega_1\Delta t(1+m^2)\frac{dr_{h;i}dz_{h;j}\alpha_{i,j}}{r_{h;i}} + \omega_1\Delta t\frac{r_i dz_{h;j}\hat{\alpha}_{i,j}}{dr_i}\right.\\
&\quad\left. + \omega_1\Delta t\frac{r_{i-1}dz_{h;j}\hat{\alpha}_{i-1,j}}{dr_{i-1}} + \omega_1\Delta t\frac{r_{h;i}dr_{h;i}\bar{\alpha}_{i,j}}{dz_j} + \omega_1\Delta t\frac{r_{h;i}dr_{h;i}\bar{\alpha}_{i,j-1}}{dz_{j-1}}\right)v_{\theta;i,j}\\
&\quad - \omega_1\Delta t\frac{r_i dz_{h;j}\hat{\alpha}_{i,j}}{dr_i}v_{\theta;i+1,j} - \omega_1\Delta t\frac{r_{i-1}dz_{h;j}\hat{\alpha}_{i,j}}{dr_{i-1}}v_{\theta;i-1,j} - \omega_1\Delta t\frac{r_{h;i}dr_{h;i}\bar{\alpha}_{i,j}}{dz_j}v_{\theta;i,j+1}\\
&\quad - \omega_1\Delta t\frac{r_{h;i}dr_{h;i}\bar{\alpha}_{i,j-1}}{dz_{j-1}}v_{\theta;i,j-1}\boxed{+ i\omega_1\Delta t\frac{m}{2}(\alpha_{i,j} + \hat{\alpha}_{i-1,j})\left(\frac{r_{i-1}}{r_{h;i}} - \frac{r_{h;i}}{r_{i-1}}\right)v_{r;i-1,j}}\\
&\quad - i\omega_1\Delta t\frac{m}{2}(\alpha_{i,j} + \hat{\alpha}_{i,j})\left(\frac{r_i}{r_{h;i}} - \frac{r_{h;i}}{r_i}\right)v_{r;i,j} = S_{\theta;i,j}.
\end{aligned} \tag{35}$$

Note that when $m > 0$ the equations for v_r and v_θ are coupled. For the v_z component we

obtain

$$\left(\bar{\rho}_{i,j} r_{h;i} dr_{h;i} dz_j + \omega_1 \Delta t m^2 \frac{dr_{h;i} dz_j \bar{\alpha}_{i,j}}{r_{h;i}} + \omega_1 \Delta t \frac{r_i dz_j \tilde{\hat{\alpha}}_{i,j}}{dr_i} \right.$$
$$\left. + \omega_1 \Delta t \frac{r_{i-1} dz_j \tilde{\hat{\alpha}}_{i-1,j}}{dr_{i-1}} + \omega_1 \Delta t \frac{r_{h;i} dr_{h;i} \alpha_{i,j+1}}{dz_{h;j+1}} + \omega_1 \Delta t \frac{r_{h;i} dr_{h;i} \alpha_{i,j}}{dz_{h;j}} \right) v_{z;i,j}$$
$$- \omega_1 \Delta t \frac{r_i dz_j \tilde{\hat{\alpha}}_{i,j}}{dr_i} v_{z;i+1,j} - \omega_1 \Delta t \frac{r_{i-1} dz_j \tilde{\hat{\alpha}}_{i-1,j}}{dr_{i-1}} v_{z;i-1,j}$$
$$- \omega_1 \Delta t \frac{r_{h;i} dr_{h;i} \alpha_{i,j+1}}{dz_{h,j+1}} v_{z;i,j+1} - \omega_1 \Delta t \frac{r_{h;i} dr_{h;i} \alpha_{i,j}}{dz_{h,j}} v_{z;i,j-1} = S_{z;i,j}. \tag{36}$$

Let us consider the boxed line in Eq. (35); it contains r_{i-1} at denominator which diverges for $i = 2$. The boxed derivatives in Eq. (32) produce that term. Using the results in Appendix A we can rewrite the representation for $m = 1$ of the second diverging derivative in Eq. (32) as

$$\left.\frac{\partial}{\partial r}\frac{v_r}{r}\right|_2 = -\frac{a_0}{r_{h;2}} + a_1 + O(r^2) = \frac{v_{r;2}/r_2 - Y}{dr_{h;2}} \tag{37}$$

$$v_{r;2} = a_0 + a_1 r_{h;2} + O(r^4). \tag{38}$$

Now Y can be easily found, taking into account that

$$2r_{h;2} = r_2 = dr_{h;2}. \tag{39}$$

The first derivative can be treated in the same way. Hence, the boxed line of Eq. (35) when $i = 2$ and $m = 1$ becomes

$$+ i\omega_1 \Delta t (\alpha_{2,j} + \hat{\alpha}_{1,j}) \left(-\frac{5}{2}\frac{r_{h;2}}{r_2}\right) v_{r;1,j}. \tag{40}$$

When $m = 2$ we can write

$$\left.\frac{v_r}{r}\right|_1 = \left.\frac{\partial v_r}{\partial r}\right|_1 = -i \left.\frac{\partial v_\theta}{\partial r}\right|_1 = -i \frac{v_{\theta;2,j} - v_{\theta;1,j}}{dr_1}. \tag{41}$$

The boxed line of Eq. (35) then becomes

$$-\omega_1 \Delta t (\alpha_{2,j} + \hat{\alpha}_{1,j}) r_{h;2} \frac{v_{\theta;2,j} - v_{\theta;1,j}}{dr_1}. \tag{42}$$

For $m \geq 3$ the above-cited line equals zero for $i = 2$. It is quite easy to verify that the equations for the components of **v** have been written in a self-adjoint form. Once we have fixed the boundary conditions we can deliver the equations to the CG solver.

The field **v** may assume (in principle) arbitrary values **V** at $z = \pm L/2$ and $r = R$, while for the singular boundary at $r = 0$ the values are dictated by conditions in Appendix A.

Thus we write

$$
\begin{aligned}
v_{r;1,j} &= \begin{cases} 0 & \text{if } m \neq 1 \\ -i\, v_{\theta;2,j} & \text{if } m = 1 \end{cases} \\
v_{\theta;1,j} &= \begin{cases} -v_{\theta;2,j} & \text{if } m = 0, 2 \\ v_{\theta;2,j} & \text{otherwise} \end{cases} \\
v_{z;1,j} &= \begin{cases} v_{z;2,j} & \text{if } m \neq 1 \\ -v_{z;2,j} & \text{if } m = 1 \end{cases} \\
v_{r;N,j} &= V_{r;R,j} \\
v_{\theta;N,j} &= -v_{\theta;N-1,j} + 2V_{\theta;R,j} \\
v_{z;N,j} &= -v_{z;N-1,j} + 2V_{z;R,j} \\
v_{r;i,1} &= -v_{r;i,2} + 2V_{r;i,-L/2} \\
v_{\theta;i,1} &= -v_{\theta;i,2} + 2V_{\theta;i,-L/2} \\
v_{z;i,1} &= V_{z;i,-L/2} \\
v_{r;i,M} &= -v_{r;i,M-1} + 2V_{r;i,+L/2} \\
v_{\theta;i,M} &= -v_{\theta;i,M-1} + 2V_{\theta;i,+L/2} \\
v_{z;i,M} &= V_{z;i,+L/2}.
\end{aligned} \tag{43}
$$

The first method to implement the boundary conditions, as explained in Subsection 3.1, can be used. In fact, the coupling of v_r and v_θ in our scheme does not spoil the self-adjoint nature of the operator.

3.3. *The Induction Equation in Cylindrical Coordinates*

Let us consider how to solve the induction equation

$$
\frac{1}{\eta}\frac{\partial \mathbf{A}}{\partial t} = -\boldsymbol{\nabla} \times \boldsymbol{\nabla} \times \mathbf{A} + \mathbf{S}, \tag{44}
$$

where the diffusive "curl–curl" operator is self-adjoint. We are not interested now in the ideal part of Eq. (1), which is treated separately with the predictor–corrector and merely adds a source term to the right-hand side of

$$
dV\left(\frac{1}{\eta} + \omega \Delta t \boldsymbol{\nabla} \times \boldsymbol{\nabla} \times\right)\mathbf{A}^{(n+1)} = dV\left(\frac{1}{\eta} - (1-\omega)\Delta t \boldsymbol{\nabla} \times \boldsymbol{\nabla} \times\right)\mathbf{A}^{(n)}. \tag{45}
$$

The equation above is the finite-difference and self-adjoint representation of Eq. (44) when $\mathbf{S} = \mathbf{0}$. It can be shown that to have stability for all Δt it must be $\frac{1}{2} \leq \omega \leq 1$. In the code we choose $\omega = \frac{1}{2}$, which is also second-order accurate in time. The analytical form of the curl–curl operator, in cylindrical coordinates and after a Fourier transform in θ, is

$$
\boldsymbol{\nabla} \times \boldsymbol{\nabla} \times \mathbf{A}|_r = \frac{m^2}{r^2}A_r - \frac{\partial^2 A_r}{\partial z^2} + i\frac{m}{r^2}\frac{\partial}{\partial r}(r A_\theta) + \frac{\partial^2 A_z}{\partial r \partial z}, \tag{46}
$$

$$
\boldsymbol{\nabla} \times \boldsymbol{\nabla} \times \mathbf{A}|_\theta = im\frac{\partial}{\partial r}\frac{A_r}{r} - \frac{\partial}{\partial r}\frac{1}{r}\frac{\partial}{\partial r} r A_\theta - \frac{\partial^2 A_\theta}{\partial z^2} + i\frac{m}{r}\frac{\partial A_z}{\partial z}, \tag{47}
$$

$$
\boldsymbol{\nabla} \times \boldsymbol{\nabla} \times \mathbf{A}|_z = \frac{1}{r}\frac{\partial}{\partial r} r \frac{\partial A_r}{\partial z} + i\frac{m}{r}\frac{\partial A_\theta}{\partial z} - \frac{1}{r}\frac{\partial}{\partial r} r \frac{\partial A_z}{\partial r} + \frac{m^2}{r^2}A_z. \tag{48}
$$

Note that the three components of **A** are coupled for $m \geq 1$. We prefer to hold η constant for the sake of simplicity. We shall drop the temporal index and write the right-hand side of Eq. (45) just as S_i, whose form is easily deduced from the left-hand side. The equation for the radial component is

$$\left(\frac{r_i dr_i dz_{h;j}}{\eta} + \Delta t\omega \frac{m^2}{r_i} dr_i dz_{h;j} + \Delta t\omega \frac{r_i dr_i}{dz_j} + \Delta t\omega \frac{r_i dr_i}{dz_{j-1}}\right) A_{r;i,j}$$
$$- \Delta t\omega \frac{r_i dr_i}{dz_j} A_{r;i,j+1} - \Delta t\omega \frac{r_i dr_i}{dz_{j-1}} A_{r;i,j-1} + i\,\Delta t\omega m \frac{dz_{h;j} r_{h;i+1}}{r_i} A_{\theta;i+1,j}$$
$$- i\,\Delta t\omega m \frac{dz_{h;j} r_{h;i}}{r_i} A_{\theta;i,j} + \Delta t\omega r_i A_{z;i+1,j} - \Delta t\omega r_i A_{z;i,j}$$
$$- \Delta t\omega r_i A_{z;i+1,j-1} + \Delta t\omega r_i A_{z;i,j-1} = S_{r;i,j}. \tag{49}$$

The equation for θ-component is

$$\left(\frac{r_{h;i} dr_{h;i} dz_{h;j}}{\eta} + \Delta t\omega \frac{r_{h;i}^2 dz_{h;j}}{r_i dr_i} + \Delta t\omega \overbrace{\boxed{\frac{r_{h;i}^2 dz_{h;j}}{r_{i-1} dr_{i-1}}}}^{a} + \Delta t\omega \frac{r_{h;i} dr_{h;i}}{dz_j}\right.$$
$$\left. + \Delta t\omega \frac{r_{h;i} dr_{h;i}}{dz_{j-1}}\right) A_{\theta;i,j} - \Delta t\omega \frac{r_{h;i} r_{h;i+1} dz_{h;j}}{r_i dr_i} A_{\theta;i+1,j}$$
$$- \Delta t\omega \overbrace{\boxed{\frac{r_{h;i} r_{h;i-1} dz_{h;j}}{r_{i-1} dr_{i-1}}}}^{b} A_{\theta;i-1,j} - \Delta t\omega \frac{r_{h;i} dr_{h;i}}{dz_j} A_{\theta;i,j+1}$$
$$- \Delta t\omega \frac{r_{h;i} dr_{h;i}}{dz_{j-1}} A_{\theta;i,j-1} + i\,\Delta t\omega m dr_{h;i} A_{z;i,j} - i\,\Delta t\omega m dr_{h;i} A_{z;i,j-1}$$
$$+ i\,\Delta t\omega m \frac{r_{h;i} dz_{h;j}}{r_i} A_{r;i,j} - i\,\Delta t\omega m \overbrace{\boxed{\frac{r_{h;i} dz_{h;j}}{r_{i-1}}}}^{c} A_{r;i-1,j} = S_{\theta;i,j}. \tag{50}$$

Finally, we write the equation for the z-component as

$$\left(\frac{r_{h;i} dr_{h;i} dz_j}{\eta} + \Delta t\omega m^2 \frac{dr_{h;i} dz_j}{r_{h;i}} + \Delta t\omega \frac{r_i dz_j}{dr_i} + \Delta t\omega \frac{r_{i-1} dz_j}{dr_{i-1}}\right) A_{z;i,j}$$
$$- \Delta t\omega \frac{r_i dz_j}{dr_i} A_{z;i+1,j} - \Delta t\omega \frac{r_{i-1} dz_j}{dr_{i-1}} A_{z;i-1,j} + \Delta t\omega r_i A_{r;i,j+1}$$
$$- \Delta t\omega r_i A_{r;i,j} - \Delta t\omega r_{i-1} A_{r;i-1,j+1} + \Delta t\omega r_{i-1} A_{r;i-1,j}$$
$$+ i\,\Delta t\omega m dr_{h;i} A_{z;i,j+1} - i\,\Delta t\omega m dr_{h;i} A_{z;i,j} = S_{z;i,j}. \tag{51}$$

Note that in Eq. (50) there are diverging terms when $i = 2$. We have marked them with a, b, and c. When $m = 0$ only the first two are present. They derive from the finite difference representation of

$$\frac{1}{r}\frac{\partial}{\partial r} r A_\theta \Big|_1 \tag{52}$$

which, considering that $A_\theta = O(r)$ for $m = 0$, equals

$$2\frac{\partial A_\theta}{\partial r}\bigg|_1. \tag{53}$$

Therefore, it is easy to verify that a and b become

$$2\frac{r_{h;2}dz_{h;j}}{dr_1}. \tag{54}$$

When $m \geq 1$ we must reconsider how we treat the derivative from which all the diverging terms come, that is

$$-\frac{\partial}{\partial r}(\boldsymbol{\nabla}\times\mathbf{A})_z\bigg|_1. \tag{55}$$

The z component of a vector in cylindrical coordinates is a scalar, so its Fourier modes behave like $O(r^m)$ near the origin. Thus we can write the equation above as a finite difference

$$-\frac{(\boldsymbol{\nabla}\times\mathbf{A})_{z;2,j}}{dr_{h;2}}. \tag{56}$$

This means that a, b, and c disappear when $m \geq 1$.

The equations for the three components of $\mathbf{A}$ have been written in a self-adjoint form. Boundary conditions must be provided now through Eq. (5). Let us write the right-hand side of Eq. (5) as $\mathbf{I}$. We need not specify boundary values for A_r when $r = R$ and for A_z when $z = -L/2$ or $z = +L/2$. In fact, there is neither a radial derivative of A_r in Eq. (46) nor an axial derivative of A_z in Eq. (48). For $r = 0$ we exploit the results found in Appendix A. So we write

$$\begin{aligned}
A_{r;1,j} &= \begin{cases} 0 & \text{if } m \neq 1 \\ -iA_{\theta;2,j} & \text{if } m = 1; \end{cases} \\
A_{\theta;1,j} &= \begin{cases} -A_{\theta;2,j} & \text{if } m = 0, 2 \\ A_{\theta;2,j} & \text{otherwise;} \end{cases} \\
A_{z;1,j} &= \begin{cases} A_{z;2,j} & \text{if } m \neq 1 \\ -A_{z;2,j} & \text{if } m = 1; \end{cases} \\
A^{(n+1)}_{\theta;N,j} + A^{(n+1)}_{\theta;N-1,j} &= A^{(n)}_{\theta;N,j} + A^{(n)}_{\theta;N-1,j} + \tfrac{1}{2}\Delta t I_{\theta;R,j} \\
A^{(n+1)}_{z;N,j} + A^{(n+1)}_{z;N-1,j} &= A^{(n)}_{z;N,j} + A^{(n)}_{z;N-1,j} + \tfrac{1}{2}\Delta t I_{z;R,j} \\
A^{(n+1)}_{r;i,1} + A^{(n+1)}_{r;i,2} &= A^{(n)}_{r;i,1} + A^{(n)}_{r;i,2} + \tfrac{1}{2}\Delta t I_{r;i,-L/2} \\
A^{(n+1)}_{\theta;i,1} + A^{(n+1)}_{\theta;i,2} &= A^{(n)}_{\theta;i,1} + A^{(n)}_{\theta;i,2} + \tfrac{1}{2}\Delta t I_{\theta;i,-L/2} \\
A^{(n+1)}_{r;i,M} + A^{(n+1)}_{r;i,M-1} &= A^{(n)}_{r;i,M} + A^{(n)}_{r;i,M-1} + \tfrac{1}{2}\Delta t I_{r;i,+L/2} \\
A^{(n+1)}_{\theta;i,M} + A^{(n+1)}_{\theta;i,M-1} &= A^{(n)}_{\theta;i,M} + A^{(n)}_{\theta;i,M-1} + \tfrac{1}{2}\Delta t I_{\theta;i,+L/2},
\end{aligned} \tag{57}$$

where we had to reintroduce the temporal index for clarity.

Due to the complex intercoupling between the equations for each component of $\mathbf{A}$, only the second method to implement boundary conditions, described in Subsection 3.1, is

applicable. Hence, we have written a subroutine that, given an input vector **A**, yields the left-hand side of Eq. (45) for all the internal points, taking into account of the boundary conditions above. This subroutine is called by the CG algorithm at each iteration.

4. THE NONLINEAR KINK

As an illustration of the application of our code to a solar coronal problem, we describe the evolution of the kink instability in coronal flux tubes. An earlier version of the present code has been used to study the linear stability of the ideal line-tied kink mode [15]. A preliminary study of the nonlinear evolution of the kink mode has also been presented [21]. The kink instability may be relevant to describing compact loop flares and the heating of the solar corona.

A particular equilibrium, known as the Gold–Hoyle field [22], has been used frequently to test new algorithms. The kink stability of the Gold–Hoyle field has been studied computationally by Mikić *et al.* [15], Foote and Craig [23], Craig and Sneyd [24], and Baty and Heyvaerts [25], among others. In order to demonstrate our techniques, we present results on the linear and nonlinear evolution of the kink instability in this configuration. Although our code was designed to model more realistic equilibria, especially those that are generated when specified twisting motions are prescribed in the photosphere, we have chosen the Gold–Hoyle field as an illustration, since its linear stability is well known [6, 7, 15, 25] and its nonlinear evolution has also been studied [24, 25].

The force-free Gold–Hoyle magnetic field is defined by

$$B_r = 0 \tag{58}$$

$$B_\theta = \frac{B_\circ r}{a(1 + r^2/a^2)}, \tag{59}$$

$$B_z = \frac{B_\circ}{(1 + r^2/a^2)}, \tag{60}$$

where a defines the length scale and $B_\circ = B_z$ at $r = 0$. The Gold–Hoyle equilibrium is also known as the uniform-twist field because the twist angle, which is defined by

$$\Phi(r) \equiv \frac{L B_\theta}{r B_z} = \frac{L}{a}, \tag{61}$$

is independent of the radius. The number of turns around the axis that a field line completes while traversing from $z = -L/2$ to $z = L/2$ is given by $\Phi/2\pi$. This configuration is unstable to an ideal MHD $m = 1$ kink instability when the twist exceeds a critical value, $\Phi > \Phi_c$. The critical twist for the line-tied kink has been determined as $\Phi_c = 2.49\pi$ by Hood and Priest [6] and by Einaudi and Van Hoven [7], $\Phi_c = 2.51\pi$ by Mikić *et al.* [15], $\Phi_c = 2.46\pi$ by Foote and Craig [23], and $\Phi_c = 2.47\pi$ by Baty and Heyvaerts [25].

We choose a 101×64 $r-z$ mesh with 32 points in θ, corresponding to the modes $-10 \le m \le 10$ after dealiasing. The outer radial wall is placed at $r = R = 20a$. The z mesh is chosen to be uniform, and the radial mesh is nonuniform, with the smallest mesh cells near the axis, where $\Delta r = 0.03a$, increasing to $\Delta r = 1.4a$ near the radial wall. We use a uniform initial density $\rho = \rho_\circ$, and a uniform initial pressure $p = p_\circ$, with $p_\circ$ selected to give a beta on axis of 1%, where $\beta_\circ = 8\pi p_\circ / B_\circ^2$. (Note that a force-free equilibrium that is

immersed in a uniform-pressure plasma still remains an equilibrium.) Since the magnetic field strength in the Gold–Hoyle equilibrium falls with the radial distance from the axis, the plasma beta increases with the radius, reaching $\beta = 1$ at $r \sim 10a$. We have chosen a finite beta to make the nonlinear evolution correspond more closely to the solar corona, where the plasma β is small, but finite.

Since we expect the twist in the solar corona to be introduced by slow magnetic field footpoint motions in the photosphere, the appropriate initial state ought to be one in which the twist is only slightly larger than the critical value for linear instability. There is no more footpoint shearing at $z = \pm L/2$ and the applied surface flow considered is $\mathbf{V} = 0$ (see Eq. (6)), because the time-scale evolution of the configuration (due to the kink) is smaller than the one for the photospheric flow. We selected a twist of $\mathbf{\Phi} = 3\pi$ in order to produce a distinguishable linear phase of the instability, while at the same time keeping the excess twist (i.e., that above the stability threshold) small. In other simulations we have found that the nature of the nonlinear state does not seem to be sensitively dependent on the excess twist, as long as it is not too large [26]. The loop length is set by the condition $L = \mathbf{\Phi} a$, giving a loop with aspect ratio $L/a = 9.42$ in this case. The (radial) Alfvén time is defined by $\tau_A = a/v_A^0$, where the Alfvén speed on the axis is given by $v_A^0 = B_\circ/\sqrt{4\pi\rho_\circ}$. A uniform viscosity is used, corresponding to a viscous dissipation time $\tau_v \equiv a^2/\nu = 100\tau_A$.

We study the ideal MHD evolution of this equilibrium (with $\eta = 0$) in order to investigate whether the nonlinear evolution of the kink instability leads to the formation of current sheets. When strong gradients develop in the magnetic field during ideal MHD numerical simulations, it may be necessary to introduce plasma resistivity. In the case of the Gold–Hoyle field, as discussed below and as noted by Baty and Heyvaerts [25], the nonlinear evolution of the kink instability does not introduce current sheets, so that it is not necessary to introduce resistivity into the calculation. This is in contrast to other equilibria that we have studied, of which the zero net-current equilibrium is a particular example, for which we have found that the nonlinear evolution of the kink leads to the formation of current sheets [26], requiring the introduction of finite resistivity during the later stages of the calculation. We were thus able to perform the present calculation with the ideal MHD model. (We note that a small amount of numerical resistivity is introduced during the calculation by the upwind treatment of the advection, as described in Appendix B.)

We start the calculation at $t = 0$ with the $m = 0$ equilibrium field given by Eqs. (58)–(60), to which we add a small $m = 1$ perturbation with an amplitude $v \sim 3 \times 10^{-4}\, v_A^0$. (The perturbation was chosen to be the eigenfunction corresponding to the most unstable linear mode in a periodic cylinder, modified suitably to have zero displacement at the axial boundaries, as required by line tying. Any small initial perturbation could have been used without affecting the nonlinear results.) The equations were integrated for $500\tau_A$, requiring about three CPU hours on the Cray YMP/C-90 at NERSC. This code has also been implemented on the Cray T3D at CINECA in Bologna.

The initial time step was chosen to be $0.1\tau_A$. The time step remained $0.1\tau_A$ during the linear part of the run, decreasing to $0.05\tau_A$ during the initial phase of the nonlinear evolution as a result of the advective flow limit on the time step (Eq. 19) and increasing back to $0.1\tau_A$ after saturation of the kink toward a new equilibrium. The advantage of using the semi-implicit scheme is illustrated by the fact that the wave Courant number (i.e., the ratio of the time step to the time step required by an explicit calculation) remains significantly larger than 1 during this calculation. Initially, the wave Courant number is 16, and it increases to 30 by the end of the calculation.

In Figs. 8 and 9 we plot the magnetic energy and the kinetic energy in various modes as a function of time. Initially, the $m = 0$ mode shows the relaxation of the analytic equilibrium to the mesh (since it is not a perfect equilibrium of the discretized equations). The $m = 1$ mode grows exponentially in time, with a growth rate $\gamma\tau_A = 0.022$. The higher-m modes show growth associated with the coupling to the $m = 1$ mode. Beginning at $t \sim 200\tau_A$, when the $m = 1$ mode reaches a significant amplitude, there is a nonlinear interaction during which the higher-m modes become sizable. This phase corresponds to the observed kinking of the axis of the flux tube. Eventually, the kink mode appears to saturate, indicating that the kinked flux tube is settling toward a new equilibrium.

The linear growth rate of the $m = 1$ mode at $\Phi = 3\pi$ is lower than previous estimates because of the effect of finite beta. For the case with zero beta, the growth rate has been estimated previously as $\gamma\tau_A = 0.034$ by Mikić *et al.* [15], $\gamma\tau_A = 0.027$ by Foote and Craig [23], and $\gamma\tau_A = 0.037$ by Baty and Heyvaerts [25]. Apparently, even though the plasma beta is small on the axis, the growth rate is changed significantly by the plasma pressure. This is because the magnetic field strength falls far from the axis in this equilibrium, so that even a small pressure can affect the kinking motion of the flux tube. Indeed, when we repeated the calculation with the zero-beta model (i.e., with $p_\circ = 0$ and a constant density), we found the linear growth rate of the $m = 1$ mode to be $\gamma\tau_A = 0.038$, in good agreement with previous zero-beta results. (The growth rate determined by Foote and Craig is only intended to be a rough approximation for this equilibrium near the marginal stability point [23].) The finite pressure leads to a reduction of the growth rate, apparently due to the fact that beta is greater than one at a large radius, as described above. The finite-beta case is a more realistic representation of the solar corona than the force-free case (with $\beta = 0$), in which the flux tube kinking in the weak-field region is not impeded.

Figures 8 and 9 show that the kinked flux tube appears to settle to a new equilibrium state. This state does not appear to have any current sheets; the magnetic field remains smooth and free of discontinuities. In Fig. 10 we show the evolution of the total

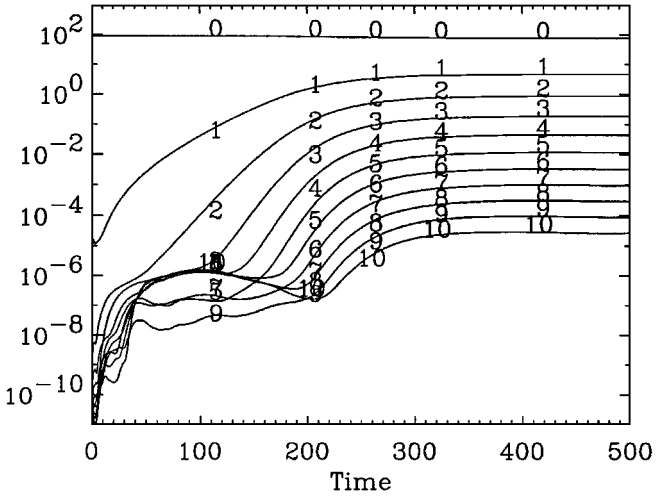

FIG. 8. Magnetic energy in various Fourier modes as a function of time for the nonlinear kink. The energy in normalized by the factor $E_\circ = B_\circ^2 a^3/(8\pi)$.

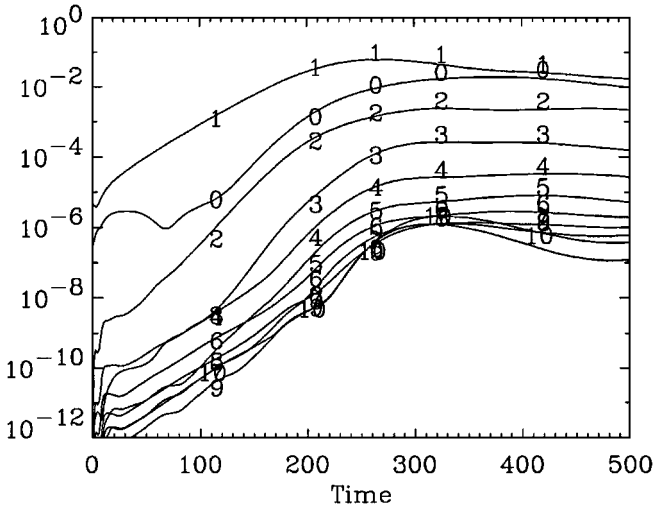

FIG. 9. Kinetic energy in various Fourier modes as a function of time for the nonlinear kink. The energy in normalized by the factor $E_\circ = B_\circ^2 a^3/(8\pi)$.

magnetic, kinetic, and thermal energies (defined by $W = \int [B^2/8\pi]\, dV$, $K = \int \frac{1}{2}\rho v^2\, dV$, and $E = \int [p/(\gamma - 1)]\, dV$, respectively). Note that as the flux tube kinks, the magnetic energy is converted into kinetic energy and, finally, into thermal energy. The kinked flux tube approaches an equilibrium that has smaller magnetic energy than the initial state.

The large-scale kinking of the flux tube is best illustrated by traces of the magnetic field lines. In Fig. 11 we show traces at four instants of time. At $t = 100\tau_A$, during the linear stage, the kink is barely perceptible in the field line plot. At $t = 250\tau_A$ the kinking pattern is clearly visible. The traces at $t = 400\tau_A$ and $t = 500\tau_A$ show that the kink is saturating to a new equilibrium state.

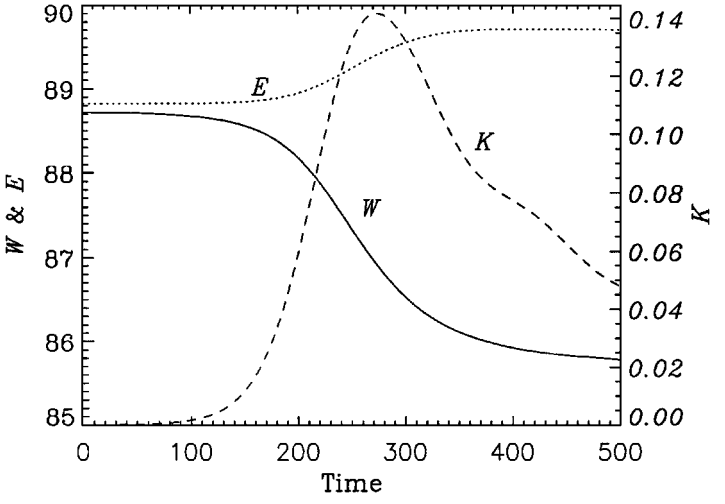

FIG. 10. The total magnetic, kinetic, and thermal energies (indicated respectively with W, K, and E) as functions of time for the nonlinear kink. The energy in normalized by the factor $E_\circ = B_\circ^2 a^3/(8\pi)$.

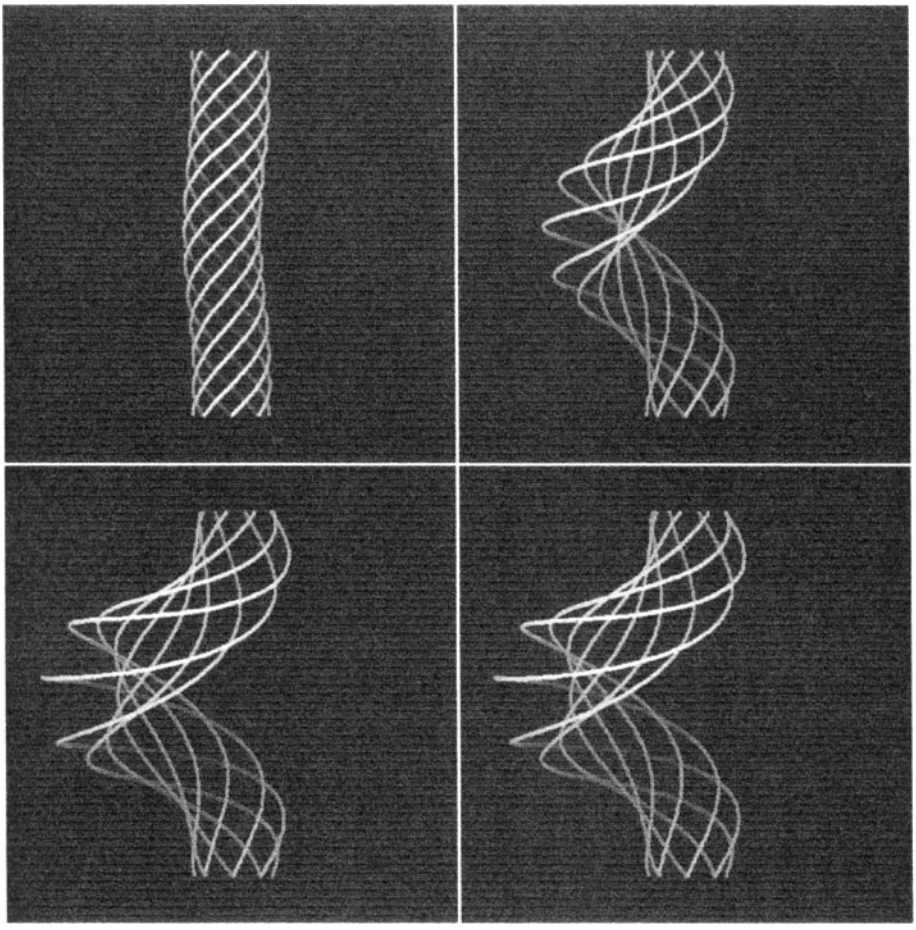

FIG. 11. Field line plots at $t = 100\tau_A$, $t = 250\tau_A$, $t = 400\tau_A$, and $t = 500\tau_A$. Field lines start from the bottom of the loop from a circle of radius a. Initially the kink pattern is barely visible, but when the instability saturates to a new equilibrium (third and fourth panel), the center of the loop has moved outward to about $r = 4a$.

5. CONCLUSIONS

We have presented a fast and accurate algorithm for the solution of the full resistive and viscous MHD equations in cylindrical coordinates in the presence of line-tied boundary conditions. The computer code based on this algorithm has been applied to the study of solar coronal flux tubes. In particular, the techniques are suited to the simulation of flux tubes whose footpoints are driven by slow photospheric motions.

The algorithm is implemented using finite differences in two dimensions, with pseudospectral derivatives along the third (periodic) dimension. The use of staggered finite-difference meshes preserves the solenoidal nature of the magnetic field and leads to a natural specification of boundary conditions on the tangential electric field and the normal magnetic field. Time advancement of the wave-like terms is performed with a leapfrog scheme. A semi-implicit operator is used in the momentum equation to give unconditional stability to wave-like terms. Advective terms are advanced using a predictor–corrector scheme, and therefore limit the time step by a Courant condition based on the flow speed. This allows

us to use significantly larger time steps than those achievable by a fully explicit algorithm. The viscous and resistive terms are discretized using a fully implicit time advance. The semi-implicit, viscous, and resistive operators are inverted using a preconditioned conjugate gradient method. Special care has been taken to maintain the self-adjointness of the discretized operators, so that a fast inversion algorithm applicable to symmetric matrices can be used.

To illustrate the application of the code, we have presented the nonlinear evolution of the ideal kink instability in the Gold–Hoyle uniform-twist field. Our results show that it is possible to follow the linear and nonlinear evolution of the kink instability. In the case of the Gold–Hoyle equilibrium, it appears that the kink instability saturates nonlinearly as the flux tube evolves to a new kinked equilibrium without the formation of current sheets. This result is in agreement with the results of Baty and Heyvaerts [25]. In contrast, Craig and Sneyd [24] concluded that the kink instability in the Gold–Hoyle field causes current sheets to form, a conclusion that is based on a calculation on a Lagrangian mesh whose accuracy is impaired when the mesh becomes significantly distorted by the finite-amplitude kink displacement. The evolution observed here for the Gold–Hoyle equilibrium contrasts sharply with the nonlinear evolution of the kink mode in a tokamak in which the nonlinear evolution causes current sheets (i.e., true discontinuities in the magnetic field) to form [27], a difference that has been attributed to the effect of line tying in the case of the coalescence instability by Longcope and Strauss [28]. In our case, it was thus possible to study the ideal MHD evolution. In general, instabilities can introduce current sheets, in which case it is necessary to study the resistive evolution. Equilibria in which the kink instability creates current sheets are discussed in [26, 29]. The role of a resonant surface in the formation of current sheets as a result of the nonlinear evolution of kink instabilities has been addressed previously [9, 25].

Therefore, the kink instability in the Gold–Hoyle equilibrium is not likely to play an important role in the solar corona, since it does not appear to cause significant heating or to lead to impulsive motions. On the other hand, other equilibria, in particular those in which the nonlinear evolution causes current sheets to form, leading to significant plasma heating, magnetic reconnection, and particle acceleration, are likely to be of interest in understanding coronal phenomena. Numerical algorithms and codes such are the one detailed here will be an important tool in this endeavor.

The code has also been used elsewhere [29, 26] to study the nonlinear evolution of instabilities in more realistic equilibria that are intended to model coronal loops formed by the twisting of uniform ambient fields and from the emergence of magnetic flux tubes from the photosphere. In these cases we have modeled the formation of current sheets, magnetic reconnection, and fast energy release.

APPENDIX A: FOURIER COEFFICIENTS IN POLAR COORDINATES

Let us consider a scalar function $F(x, y)$. We assume it is regular near the origin and we expand it in Taylor series

$$F(x,y) = F_0 + x\left.\frac{\partial F}{\partial x}\right|_0 + y\left.\frac{\partial F}{\partial y}\right|_0 + \frac{x^2}{2}\left.\frac{\partial^2 F}{\partial x^2}\right|_0 + \frac{y^2}{2}\left.\frac{\partial^2 F}{\partial y^2}\right|_0 + xy\left.\frac{\partial^2 F}{\partial x\,\partial y}\right|_0 + \cdots. \quad (62)$$

Since $x = r\cos\theta$ and $y = r\sin\theta$, We can rewrite Eq. (62) as

$$F(x,y) = F_0 + r\left[\frac{1}{2}\left(\left.\frac{\partial F}{\partial x}\right|_0 - i\left.\frac{\partial F}{\partial y}\right|_0\right)e^{i\theta} + \frac{1}{2}\left(\left.\frac{\partial F}{\partial x}\right|_0 + i\left.\frac{\partial F}{\partial y}\right|_0\right)e^{-i\theta}\right]$$
$$+ r^2\left[\frac{1}{8}\left(\left.\frac{\partial^2 F}{\partial x^2}\right|_0 - 2i\left.\frac{\partial^2 F}{\partial x\partial y}\right|_0 r - \left.\frac{\partial^2 F}{\partial y^2}\right|_0\right)e^{i2\theta} + \frac{1}{4}\left(\left.\frac{\partial^2 F}{\partial x^2}\right|_0 + \left.\frac{\partial^2 F}{\partial y^2}\right|_0\right)\right.$$
$$\left. + \frac{1}{8}\left(\left.\frac{\partial^2 F}{\partial x^2}\right|_0 + 2i\left.\frac{\partial^2 F}{\partial x\,\partial y}\right|_0 - \left.\frac{\partial^2 F}{\partial y^2}\right|_0\right)e^{-i2\theta}\right] + \cdots. \tag{63}$$

Hence the r^n term is a certain combination of exponential functions $e^{im\theta}$, with $-n \le m \le n$. Since $e^{im\theta}$ may appear only when $n \ge |m|$ when we calculate the Fourier series of F, we obtain

$$\tilde{F}^{(m)}(r) = \sum_{n=|m|}^{\infty} c_n^{(m)} r^n. \tag{64}$$

Thus we have $\tilde{F}^{(m)}(r) = O(r^{|m|})$ for small r.

Let us consider now only one term of the Fourier series $a^{(m)}(x,y) = \tilde{F}^{(m)}(r)e^{im\theta}$. Notwithstanding r is always defined to be greater than zero, we notice that in an algebraic point of view we are allowed to write

$$r \to -r, \tag{65}$$
$$\theta \to \theta + \pi. \tag{66}$$

In this case x and y do not change and so

$$\tilde{F}^{(m)}e^{im\theta}(r) = \tilde{F}^{(m)}(-r)e^{im\theta}(-1)^m. \tag{67}$$

Let us expand both members of Eq. (67) in Taylor series around $r = 0$, obtaining

$$\sum_{n=|m|}^{\infty} c_n^{(m)} r^n (1 - (-1)^{n+m}) = 0. \tag{68}$$

This imposes the following condition on the Taylor series terms of the mth Fourier coefficient of $F(x,y)$,

$$c_n^{(m)} = 0 \begin{cases} n+m = 2k+1 \\ k = 0, 1, 2, \ldots. \end{cases} \tag{69}$$

This means that the Taylor series of an even coefficient has only even terms and, vice versa, if m is odd only odd terms are found.

Let us examine now a vector $\mathbf{U} = (U_x, U_y)$, where the vector components $U_x(x,y)$ and $U_y(x,y)$ are scalar functions with the same properties of $F(x,y)$. The components of $\mathbf{U}$ in polar coordinates are

$$U_r = U_x\cos\theta + U_y\sin\theta, \tag{70}$$
$$U_\theta = -U_x\sin\theta + U_y\cos\theta. \tag{71}$$

From this follows that the Fourier series coefficients are

$$\tilde{U}_r^{(m)} = \tfrac{1}{2}\left(\tilde{U}_x^{(m-1)} + \tilde{U}_x^{(m+1)} - i\tilde{U}_y^{(m-1)} + i\tilde{U}_y^{(m+1)}\right), \tag{72}$$

$$\tilde{U}_\theta^{(m)} = \tfrac{1}{2}\left(i\tilde{U}_x^{(m-1)} - i\tilde{U}_x^{(m+1)} + \tilde{U}_y^{(m-1)} + \tilde{U}_y^{(m+1)}\right). \tag{73}$$

Thus $\tilde{U}_r^{(m)}$ and $\tilde{U}_\theta^{(m)}$ are $O(r^{\min(|m-1|,|m+1|)})$. When m is even only odd terms of the Taylor series are found and vice versa. Note that for $r = 0$ and $m \geq 1$ the following equality holds:

$$\tilde{U}_r^{(m)} = -i\tilde{U}_\theta^{(m)}. \tag{74}$$

In cylindrical coordinates the third component U_z behaves as a scalar function.

APPENDIX B: STABILITY OF PREDICTOR–CORRECTOR ADVECTION ALGORITHMS

The typical advection equation in one dimension is

$$\frac{\partial f}{\partial t} + v\frac{\partial f}{\partial x} = 0. \tag{75}$$

To solve the equation above we employ the predictor–corrector algorithm

$$\begin{aligned} \frac{f_j^* - f_j^n}{\Delta t} &= -vDf^n, \\ \frac{f_j^{n+1} - f_j^n}{\Delta t} &= -\alpha v Df^* - (1-\alpha)vDf^n, \end{aligned} \tag{76}$$

where $0 \leq \alpha \leq 1$. For centered differences

$$Df = \frac{f_{j+1} - f_{j-1}}{2\Delta x}, \tag{77}$$

while for upwind differences

$$Df = \begin{cases} \dfrac{f_j - f_{j-1}}{\Delta x} & \text{if } v > 0, \\ \dfrac{f_{j+1} - f_j}{\Delta x} & \text{if } v < 0. \end{cases} \tag{78}$$

The scheme above is first-order accurate in time. In order to perform a Von Neumann stability analysis we suppose that a local solution behaves like $f(j\Delta x, t_n) = z^n \exp(ikj\Delta x)$ and we assume for simplicity that $v > 0$. The amplification factor $z(k)$ must have modulus less than 1 for stability (see [30] for a more complete discussion of the method). Substituting f into Eq. (76) we obtain

$$z = 1 - \frac{v\Delta t}{\Delta x} Q\left(\alpha\frac{v\Delta t}{\Delta x}Q - 1\right); \tag{79}$$

$$Q = \begin{cases} i\sin(k\Delta x) & \text{centered differences} \\ 1 - \cos(k\Delta x) + i\sin(k\Delta x) & \text{upwind differences.} \end{cases}$$

The case of centered differences and no predictor–corrector ($\alpha = 0$) yields $|z|$ always greater than one and is unconditionally unstable. If we introduce upwind differences then we have $|z| \leq 1$ when

$$v\frac{\Delta t}{\Delta x} \leq 1, \tag{80}$$

the so-called Courant condition. With the predictor–corrector and centered differences we obtain

$$\frac{2\alpha - 1}{\alpha^2} \geq \left(\frac{v\Delta t}{\Delta x}\right)^2 \tag{81}$$

for stability. For a fully advanced corrector ($\alpha = 1$) we find again the Courant condition. Using these methods introduces a numerical viscosity term

$$\nu_n \frac{\partial^2 f}{\partial x^2} \tag{82}$$

into Eq. (75), which is useful to damp small unresolved scales. Let us write $z = \exp(-i\omega_r \Delta t + \gamma \Delta t)$ and then find γ from Eq. (79), limiting ourselves to the case $k\Delta x \ll 1$. From Eq. (82) it follows that the numerical viscosity coefficient is $\nu_n = -\gamma/k^2$. With upwind differences and $\alpha = 0$ its value is

$$\nu \simeq \frac{v\Delta x}{2}\left(1 - \frac{v\Delta t}{\Delta x}\right), \tag{83}$$

and for $\alpha = 1$ and centered differences

$$\nu \simeq \frac{v^2 \Delta t}{2}. \tag{84}$$

The situation in the code is complicated, with respect to this simple example, by the presence of nonuniform three-dimensional meshes in a non-Cartesian frame of reference. Furthermore, the conditions above are only *necessary* and not sufficient for stability. Fully advanced predictor–corrector is used to stabilize advection in the periodic direction θ, since we cannot upwind θ-derivative. We normally combine this method with upwind differences in r and z, originating the stability condition showed in Eq. (19) since both must obey the Courant condition.

APPENDIX C: POSITIVE DEFINITENESS OF A SYMMETRIC TRIDIAGONAL MATRIX

Let us consider a symmetric tridiagonal matrix A of the form:

$$\begin{aligned} A_{i,i} &= \begin{cases} c_2 + b_1 + b_2 + \mathcal{C}_0 b_1, & i = 2, \\ c_i + b_{i-1} + b_i, & 3 \leq i \leq N-2, \\ c_{N-1} + b_{N-2} + b_{N-1} + \mathcal{C}_L b_{N-1}, & i = N-1, \end{cases} \\ A_{i,i+1} &= b_i \quad 2 \leq i \leq N-2 \\ A_{i,i-1} &= b_{i-1} \quad 3 \leq i \leq N-1, \end{aligned} \tag{85}$$

where c_i and b_i are positive and $\mathcal{C}_0$ and $\mathcal{C}_L$ may be either 1 or -1. A matrix A is positive definite if and only if

$$x \cdot A \cdot x \geq 0 \quad \forall x \neq 0. \tag{86}$$

The condition above becomes for our matrix

$$\sum_{i=2}^{N-1} x_i^2 (c_i + b_i + b_{i-1}) + \sum_{i=3}^{N-1} x_i x_{i-1} b_{i-1} \tag{87}$$

$$+ \sum_{i=2}^{N-2} x_i x_{i+1} b_i + x_2^2 b_1 \mathcal{C}_0 + x_{N-1}^2 b_{N-1} \mathcal{C}_L \geq 0. \tag{88}$$

We can rewrite it as

$$\sum_{i=2}^{N-1} x_i^2 c_i + \sum_{i=2}^{N-2} b_i (x_i - x_{i+1})^2 + x_2^2 b_1 (1 + \mathcal{C}_0) + x_{N-1}^2 b_{N-1} (1 + \mathcal{C}_L) \geq 0, \tag{89}$$

that is manifestly true.

ACKNOWLEDGMENTS

We would like to thank Claudio Chiuderi, Giorgio Einaudi, Jon Linker, Gerard Van Hoven, and Marco Velli for many helpful discussions. Computations were performed at the Interuniversity Consortium of the Northeastern Italy for Automatic Computing (CINECA) and at the National Energy Research Supercomputer Center (NERSC) at Livermore, California. This work was supported in part by a NASA contract to Science Applications International Corporation (SAIC). One of us (RL) wishes to thank Roberto Casini for collaboration in developing imaging software; the University of California at Irvine, and SAIC for their kind hospitality during the development of this work. The visit of RL to UCI was supported, in part, by NSF and NASA.

REFERENCES

1. B. M. Haisch and M. Rodonò (Eds.), *Solar and Stellar Flares* (Kluwer Academic, Dordrecht, 1989).
2. P. Ulmschneider, E. R. Priest, and R. Rosner (Eds.), *Mechanisms of Chromospheric and Coronal Heating* (Springer-Verlag, Berlin, 1991).
3. G. Belvedere, M. Rodonò, and G. M. Simnett (Eds.), *Advances in Solar Physics* (Springer-Verlag, Berlin, 1994).
4. E. N. Parker, *Spontaneous Current Sheets in Magnetic Fields with Applications to Stellar X-rays* (Oxford Univ. Press, New York, 1994).
5. G. Van Hoven, Y. Mok, and Z. Mikić, Coronal loop formation resulting from photospheric convection, *Astrophys. J.* **440**, L105 (1995).
6. A. W. Hood and E. R. Priest, Critical conditions for magnetic instabilities in force-free coronal loops, *Geophys. Astrophys. Fluid Dyn.* **17**, 297 (1981).
7. G. Einaudi and G. Van Hoven, The stability of coronal loops: finite-length and pressure-profile limits, *Sol. Phys.* **88**, 163 (1983).
8. M. Velli, G. Einaudi, and A. W. Hood, Boundary effects on the magnetohydrodynamic stability of a resistive plasma, *Astrophys. J.* **350**, 419 (1990).
9. M. Velli, G. Einaudi, and A. W. Hood, Ideal kink instabilities in line-tied coronal loops: growth rates and geometrical properties, *Astrophys. J.* **350**, 428 (1990).

10. A. Y. Aydemir and D. C. Barnes, An implicit algorithm for compressible three-dimensional magnetohydrodynamic calculations, *J. Comput. Phys.* **59**, 108 (1985).
11. D. S. Harned and W. Kerner, Semi-implicit method for three-dimensional compressible magnetohydrodynamic simulations, *J. Comput. Phys.* **60**, 62 (1985).
12. D. S. Harned and D. D. Schnack, Semi-implicit method for long time scale magnetohydrodynamic computations in three dimensions, *J. Comput. Phys.* **65**, 57 (1986).
13. D. D. Schnack, D. C. Barnes, Z. Mikić, D. S. Harned, and E. J. Caramana, Semi-implicit magnetohydrodynamic calculation, *J. Comput. Phys.* **70**, 330 (1987).
14. K. Lerbinger and J. F. Luciani, A new semi-implicit method for MHD computations, *J. Comput. Phys.* **97**, 444 (1991).
15. Z. Mikić, D. D. Schnack, and G. Van Hoven, Dynamical evolution of twisted magnetic flux tubes. I. Equilibrium and linear stability, *Astrophys. J.* **361**, 690 (1990).
16. G. H. Golub and C. F. Van Loan, *Matrix Computations* (Johns Hopkins Univ. Press, 1989), p. 516.
17. Z. Mikić and E. C. Morse, The use of a preconditioned bi-conjugate gradient method for hybrid plasma stability analysis, *J. Comput. Phys.* **61**, 154 (1984).
18. D. Gottlieb and S. A. Orszag, *Numerical Analysis of Spectral Methods* (SIAM, Philadelphia, 1977).
19. Z. Mikić, to be published.
20. D. D. Schnack, D. C. Baxter, and E. J. Caramana, A pseudospectral algorithm for three-dimensional magnetohydrodynamic simulation, *J. Comput. Phys.* **55**, 485 (1984).
21. Z. Mikić, Magnetohydrodynamic modeling of the solar corona, *Phys. Fluids B* **2**, 1450 (1990).
22. T. Gold and F. Hoyle, On the origin of solar flares, *Mon. Not. R. Astron. Soc.* **120**, 89 (1960).
23. B. J. Foote and I. J. D. Craig, The linear stability of line-tied coronal magnetic fields, *Astrophys. J.* **350**, 437 (1990).
24. I. J. D. Craig and A. D. Sneyd, Nonlinear development of the kink instability in coronal flux tubes, *Astrophys. J.* **357**, 653 (1990).
25. H. Baty and J. Heyvaerts, Electric current concentration and kink instability in line-tied coronal loops, *Astron. Astrophys.* **308**, 935 (1996).
26. M. Velli, R. Lionello, and G. Einaudi, Kink modes and current sheets in coronal loops, *Sol. Phys.* **172**, 257 (1996).
27. M. N. Rosenbluth, R. Y. Dagazian, and P. H. Rutherford, Non-linear properties of the internal kink instability in the cylindrical tokamak, *Phys. Fluids* **16**, 1894 (1973).
28. D. W. Longcope and H. R. Strauss, The form of ideal current layers in line-tied magnetic fields, *Astrophys. J.* **437**, 851 (1994).
29. G. Einaudi, R. Lionello, and M. Velli, Magnetic reconnection in solar coronal loops, *Adv. Space Res.* **19**, 1875 (1997).
30. W. H. Press, B. P. Flannery, Saul A. Teukolsky, and W. T. Vetterling, *Numerical Recipes* (Cambridge Univ. Press, Cambridge, 1986).