JOURNAL OF COMPUTATIONAL PHYSICS 42, 288–308 (1981)

Accurate Calculations of Field-Reversed Axisymmetric Equilibria and Their MHD Stability Properties*

DAVID V. ANDERSON

Lawrence Livermore National Laboratory, University of California, Livermore, California 94550

AND

DANIEL C. BARNES†

Los Alamos National Laboratory, Los Alamos, New Mexico 87545

Received November 14, 1980

Finite elements with bi-cubic, *B*-spline basis functions are used to represent the flux functions of an axisymmetric, field-reversed plasma. Equilibria, in this representation, are obtained by the collocation method and a generalized ICCG algorithm. For the infinite, toroidal-mode number limit of the MHD energy principle, coupled Sturm–Liouville-like equations result on each flux line. A secant-bisection shooting method employing the gridless Gear integrator is used to find very accurate eigenfunctions, eigenvalues, and growth rates from the continuous-spline representation of the fields. Precise agreement with some exact analytic work is obtained.

1. INTRODUCTION

In recent years several field-reversed theta-pinch experiments have been found to be stable [1–3] when in fact the ideal MHD stability theory indicated they should be unstable. To understand the mechanisms which might stabilize these configurations, we have set out to first have a good quantitative understanding of the ideal MHD theory. The theoretical calculations are done in two parts—equilibrium and stability. Because we are concerned that inconsistent representations and truncation errors from difference formulas might produce erroneous results—particularly in the stability calculations—we have made an effort to do both the equilibrium and stability calculations very accurately with a consistent, continuous representation employing finite elements.

* Work performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under Contract W-7405-ENG-48 and by the Los Alamos National Laboratory under Contract W-7405-ENG-36.

† Current address: Institute for Fusion Studies, University of Texas, Austin, Texas 78712.

0021-9991/81/080288–21$02.00/0

The problem we address here, first studied by Bernstein *et al.* [4], uses the energy principle to derive equations appropriate to the infinite, toroidal-mode number limit of axisymmetric, field-reversed equilibria having no toroidal fields. Applications to field-reversed mirrors and theta pinches are intended. Earlier calculations, employing finite difference methods, on Astron equilibrium and stability were performed by codes which were the forerunners of the algorithms described here [5]. Related analyses for multipoles has been given by D'lppolito *et al.* [6]. The modes we investigate, commonly referred to as ballooning modes, differ somewhat from those found in tokamaks [7] because we have no toroidal field and hence all field lines are closed.

Several other approaches to the MHD-stability studies have been pursued and the equilibria we generate are made available to and have been used in other stability calculations. The equilibria code, CYLEQ, to be described below, produces disk files of its results which form the input to several stability codes:

1. MALICE (Magnetic Almost Lagrangian Implicit Continuous Eulerian) is a 3-D MHD, time-dependent, nonlinear code for simulating the plasma motion [8]. Taking the CYLEQ-generated equilibria as initial conditions, we have used MALICE to study the long-wavelength, MHD-stability properties of both stationary and rotating equilibria [9].

2. MHD-2D is a fully nonlinear, 2-D, r, z-resistive, MHD code used primarily to study resistive tearing modes and the associated transport of axisymmetric equilibria [10]. In one mode the CYLEQ equilibria are used to initialize these simulations.

3. RIPPLE VI [11] and ALIMO [12] calculate the 2-D, nonlinear, MHD, r, z flow given a pure Fourier mode for the azimuthal dependence. CYLEQ-generated equilibria for the FRX-B experiment have been shown by the RIPPLE VI code to have the tilting instability.

4. STABGROW is the stability code to be described in this paper. It, too, uses the CYLEQ results from the disk file to form its representation of the equilibrium.

The equilibrium model and its numerical algorithms are described in Sections 2 and 3. Then, in Sections 4 and 5, we give the stability model and the numerical methods used in its solution. Results, including a demonstration of the correctness of the methods, are presented in Section 6 and some conclusions are offered in Section 7.

Although the results generated by these models turn out to be pessimistic for ideal MHD stability, the models do provide a means of making quantitative studies of the ideal MHD properties of various equilibria. Other mechanisms, such as the finite Larmor radius (FLR) effect, can then be expected to stabilize some of them. Our codes may give us a handle for determining the least unstable configurations—perhaps those most amenable to stabilization by these other phenomena.

2. Equilibrium Model

We obtain a simple model for the equilibrium fields by assuming a single-fluid, scalar-pressure plasma and neglecting electrostatic fields and any plasma motion. Further, we consider only axisymmetric configurations with no toroidal field and model them in the r, z plane. With these restrictions, pressure balance reduces to

$$J_\theta(\psi) = r\frac{\partial P}{\partial \psi}, \tag{1}$$

where $\psi = rA_\theta$ is the flux function. Throughout this article rationalized Gaussian units are employed. Of the Maxwell equations, only Ampere's law remains. In terms of the flux function it is

$$\frac{\partial^2 \psi}{\partial z^2} + \frac{\partial^2 \psi}{\partial r^2} - \frac{1}{r}\frac{\partial \psi}{\partial r} = -rJ_\theta(\psi) = -r^2\frac{\partial P}{\partial \psi}. \tag{2}$$

We note this is a nonlinear equation in ψ, sometimes referred to as the Grad–Shafranov equation. Pressure balance along flux lines requires $P = P(\psi)$ alone. One choice for the pressure function we employ is

$$P(\psi) = (1-H)P_1\left(\frac{\psi_s - \psi}{\psi_s - \psi_v}\right)^m + HP_1(1 + \cos\{\pi[\alpha + \gamma\tanh(a\psi + b)]^m\}). \tag{3}$$

A special case, Hill's vortex, is obtained for $H = 0$, $\psi_s = 0$, $m = 1$. The structure of the formula is intended to allow for hollowed-out current profiles and gives control over the width and position of the current distribution. This empirical formula for $P(\psi)$ has been found to give good agreement with field-reversed theta-pinch experiments. Furthermore, it should be noted, $\psi_s > 0$ allows fnite pressure on open field lines beyond the separatrix (a feature also seen in experiment), which improves the MHD stability to interchange modes.

Equation (2) with appropriate boundary conditions is then solved for $\psi = \psi(r, z)$. Since ψ arises both from plasma currents as well as currents in the confining coils we consider these contributions separately and write

$$\psi = \psi_{\text{pl}} + \psi_{\text{coil}}. \tag{4}$$

In our model the coils are assumed to be axisymmetric hoops at arbitrary r, z locations; a straightforward evaluation of complete elliptic integrals gives ψ_{coil} everywhere on the domain. Within the computational domain, where only plasma currents are allowed, the equilibrium equation can be written

$$\frac{\partial^2 \psi_{\text{pl}}}{\partial z^2} + \frac{\partial^2 \psi_{\text{pl}}}{\partial r^2} - \frac{1}{r}\frac{\partial \psi_{\text{pl}}}{\partial r} = -r^2\frac{\partial P(\psi)}{\partial \psi}. \tag{5}$$

The boundary conditions for ψ_{pl} can be obtained self-consistently from the Biot–Savart law or, for example, in the case of a perfectly conducting boundary, $\psi_{pl} = 0.0$ can be used.

3. Finite Element Equilibrium Solver

We solve Eq. (5) on a 2-D, rectangular, r, z domain. The grid cells are rectangles and the spline formula for ψ_{pl} is given as

$$\psi = \sum_{kl} \alpha_{kl} U_k(r)\, W_l(z), \tag{6}$$

where U and W are cubic B-spline basis functions. Figure 1 shows the localized, twice-differentiable form of them. We define $U_k(r)$ by

$$U_k(r) = \frac{1}{6\,\Delta r}\left[(r - r_{k+1})_+^3 - 4(r - r_k)_+^3 + 6(r - r_{k-1})_+^3 - 4(r - r_{k-2})_+^3 + (r - r_{k-3})_+^3\right], \tag{7}$$

where $(r)_+ = r$ for $r > 0$ and $(r)_+ = 0$ for $r \leqslant 0$. Among other properties we note

$$U_k(r_i) = \frac{1}{6}, \frac{2}{3}, \frac{1}{6}, 0 \qquad \text{for} \quad k = i+1, i, i-1, \text{other } i, \text{ respectively,}$$

$$\left.\frac{\partial U_k}{\partial r}\right|_{r_i} = \frac{1}{\Delta r}\left(-\frac{1}{2}, 0, \frac{1}{2}, 0\right) \qquad \text{for} \quad k = i+1, i, i-1, \text{other } i, \text{ respectively,}$$

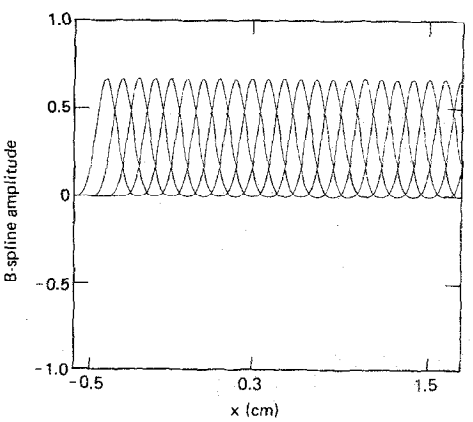

FIG. 1. A set of cubic B-spline basis functions.

and

$$\left.\frac{\partial^2 U_k}{\partial r^2}\right|_{r_i} = \frac{1}{\Delta r^2}(1, -2, 1, 0) \qquad \text{for} \quad k = i+1, i, i-1, \text{ other } i, \text{ respectively,} \tag{8}$$

which are useful in deriving the discrete equations below. Analogous definitions apply to the (z) functions. Substitution of Eqs. (6) and (8) into Eq. (5) yields the discrete system of equations to be solved:

$$\begin{aligned} &A_{ij}\alpha^{n+1}_{i-1,j-1} + B_{ij}\alpha^{n+1}_{i,j-1} + C_{ij}\alpha^{n+1}_{i+1,j-1} + D_{ij}\alpha^{n+1}_{i-1,j} + E_{ij}\alpha^{n+1}_{i,j} \\ &\quad + F_{ij}\alpha^{n+1}_{i+1,j} + G_{ij}\alpha^{n+1}_{i-1,j+1} + H_{ij}\alpha^{n+1}_{i,j+1} + K_{ij}\alpha^{n+1}_{i+1,j+1} \\ &\quad = -r_j^2\left(\frac{\partial P}{\partial \psi}\right)^n_j \end{aligned} \tag{9}$$

for $i = 1, i_{\max}; j = 1, j_{\max}$. Here

$$A_{ij} = C_{ij} = \frac{1}{6\Delta_z^2} + \frac{1}{6\,\Delta r^2} + \frac{1}{12r\,\Delta r},$$

$$G_{ij} = K_{ij} = \frac{1}{6\Delta_z^2} + \frac{1}{6\Delta_r^2} - \frac{1}{12r\,\Delta r},$$

$$D_{ij} = F_{ij} = \frac{2}{3\Delta_z^2} - \frac{1}{3\,\Delta r^2},$$

$$E_{ij} = \frac{-4}{3\Delta_z^2} - \frac{4}{3\,\Delta r^2},$$

$$B_{ij} = \frac{1}{3\Delta_z^2} + \frac{2}{3\Delta_z^2} + \frac{1}{3r\,\Delta r},$$

and

$$H_{ij} = \frac{1}{3\Delta_z^2} + \frac{2}{3\,\Delta r^2} - \frac{1}{3r\,\Delta r}. \tag{10}$$

These hold for a regularly spaced grid. The system in Eq. (9) has $N = i_{\max} j_{\max}$ equations and is a nonsymmetric, banded-matrix equation with nine nonzero diagonals. Equations represented by rows in the matrix with less than nine nonzero entries correspond to boundary value operators and the corresponding entries in the right-hand-side vector are the boundary values [rather than the $-r_j^2(\partial P/\partial\psi)^n_{ij}$ as implied in Eq. (9)]. Thus general boundary conditions are included naturally in the matrix form. More details about the implementation of these quite general boundary conditions are given in Ref. [18].

The flux function ψ is now smoothly represented by the bi-cubic splines—twice

differentiable in each coordinate. This property allows other quantities such as **B** and **VB** to be defined analytically anywhere on the domain.

The method given in Eq. (9) and used here is called collocation and gives second-order spatial accuracy. Use of the Galerkin procedure (which essentially modifies the right-hand side to be a local average of $r^2\,\partial P/\partial\psi$) would give third-order spatial accuracy. Equation (9) is solved by the ILUCG algorithm [13]—a generalization of the ICCG method [14, 15] which will not be discussed here.

Since Eq. (9) is given with iteration levels $n+1$ and n on the left and right sides, respectively, it is a linear matrix equation for the α^{n+1} array. Every time a new α_{ij}^{n+1} array is determined we update $\partial P/\partial\psi$ to be a function of the ψ's at the new iteration level. These nonlinear iterations—a Picard iteration sequence—continue until the α^n converge.

We have discovered that quicker convergence may be obtained in practice by a procedure called mixed iterations [16]. Because ILUCG uses an iterative process itself to solve the linear problem in Eq. (9), it is possible to mix the linear and nonlinear iterations. So perhaps every few linear iterations are followed by a nonlinear update of the right side of Eq. (9). At the extreme we have tried the locked iterative method where the nonlinear update is done on every linear iteration but we have found the mixed iterations to work best.

A comparison with an older method, ADI (useful for solving five-banded-matrix approximations to the equilibrium equations), shows ILUCG requires about the same computer time per iteration as ADI. But ILUCG tends to use less than half as many iterations to a given level of convergence and is consistent with our continuous representation. All difficulties associated with a splitting method like ADI are absent for ILUCG. For example, ADI is known to converge to an oscillating pair of non-solutions for some cases—this does not happen with ILUCG. Further, ADI users must provide a convergence parameter which determines the speed of convergence or causes numerical instability; no such parameter appears in ILUCG.

We routinely use this algorithm, ILUCG, in the equilibrium code CYLEQ to obtain solutions with very small relative error in Eq. (9) of the order of 10^{-8} or less. For a 25×49 grid roughly 250 iterations (linear and nonlinear) are required for this accuracy in a typical problem. This takes about 20 sec on the CDC 7600 computer.

4. Stability Model for Ballooning

Among the possible stability models, we have chosen to study the ideal MHD modes in the large toroidal-mode number limit. Equations resembling Sturm–Liouville equations are derived from the energy principle. The normalized displacements X, Y, Z are defined

$$\xi_\phi = \frac{1}{rB} X,$$

$$\xi_\theta = \frac{r}{m} Y_{m_{m\to\infty}} = 0,$$

and

$$\xi_\chi = BZ, \tag{11}$$

in terms of the physical displacements (length) $\xi_\psi, \xi_\theta, \xi_\chi$. In the well known work of Bernstein *et al.* [4], the equation

$$\frac{\partial}{\partial s}\left(\frac{1}{r^2B}\frac{\partial X}{\partial s}\right) + (\Lambda - P'D)\frac{X}{B} = \frac{D}{B(H + U/\gamma P)}\int \frac{DX}{B}\,ds \tag{12}$$

was obtained by using the mathematically convenient norm

$$1 = \frac{\pi}{2}\int ds\,\frac{X^2}{B}. \tag{13}$$

Here Λ is the eigenvalue (positive for stability), $D = 2k/rB$ is a curvature-dependent factor (where k is the curvature), $U \equiv \int ds/B$, $H \equiv \int ds/B^3$, s is arc length along a field line, and γ the usual thermodynamic ratio of specific heats. The norm chosen is not a physical constraint on the system and the eigenvalues cannot be related to growth rates.

If we alternatively minimize δW in the physically appropriate kinetic energy norm then two coupled equations result:

$$\frac{\partial}{\partial s}\left(\frac{1}{r^2B}\frac{\partial X}{\partial s}\right) + \left(\frac{\Lambda}{r^2B^2} - P'D - \frac{D^2}{1/B^2 + 1/P}\right)\frac{X}{B} = \frac{D}{1/B^2 + 1/P}\frac{\partial Z}{\partial s},$$

and

$$\frac{\partial}{\partial s}\left(\frac{B}{1/B^2 + 1/P}\frac{\partial Z}{\partial s}\right) + BZ = -\frac{\partial}{\partial s}\left(\frac{DX}{1/B^2 + 1/\gamma P}\right). \tag{14}$$

The kinetic energy norm is given by

$$1 = \frac{\pi}{2}\int \frac{ds}{B}\rho\left[\left(\frac{X}{rB}\right)^2 + (BZ)^2\right]. \tag{15}$$

Now $\omega^2\rho = \Lambda$ holds so growth rates $i\omega$ can be obtained when $\omega^2 < 0$ (which is of course the condition for MHD instability). In the derivations of Eq. (14) from the energy principle, it is assumed that the perturbed motion of the MHD fluid is adiabatic. We have also derived equations similar in form to Eq. (14), in which the double adiabatic model was used in the perturbations, and other models for the perturbations could lead to equations of this form. The code we describe here should generalize to study these other stability models. For this report we shall restrict ourselves to the single adiabatic law assumed in Eq. (14).

Equation (14) can be written in the form

$$\begin{aligned} &\frac{\partial}{\partial s}\left(f\frac{\partial X}{\partial s}\right)+hX=c\frac{\partial Z}{\partial s}, \qquad h=h(\Lambda);\\ &\frac{\partial}{\partial s}\left(g\frac{\partial Z}{\partial s}\right)+kZ=-\frac{\partial}{\partial s}(cX), \qquad k=k(\Lambda). \end{aligned} \tag{16}$$

Since the Gear solver [17, 18] we shall use to integrate these equations requires first-order ordinary differential equations, we transform to such a system. We define

$$U=f\frac{\partial X}{\partial s} \qquad \text{and} \qquad W=g\frac{\partial Z}{\partial s}+cX. \tag{17}$$

Then the first-order system is

$$\frac{\partial U}{\partial s}=c\frac{\partial Z}{\partial s}-hX,$$

$$\frac{\partial X}{\partial s}=\frac{U}{f},$$

$$\frac{\partial W}{\partial s}=-kZ,$$

and

$$\frac{\partial Z}{\partial s}=\frac{W-cX}{g}. \tag{18a}$$

To these we add the two equations governing the location of the flux line itself:

$$\frac{\partial r}{\partial s}=\frac{B_r}{B},$$

and

$$\frac{\partial z}{\partial s}=\frac{B_z}{B}. \tag{18b}$$

All of the equilibria we consider are symmetric about $z=0$. Hence it is only necessary to integrate in the $z>0$ sub-domain. The eigenfunctions X will be either even or odd functions in z. And Z will be either odd or even, respectively. Similarly, in the arc length coordinate s, X (and Z) will be even (odd) or odd (even). Defining s_m to be the arc length halfway around a flux line (back at the $z=0$ midplane) we also require X (and Z) to be even (odd) or odd (even) there. This over-specification of boundary data is what makes the system given in Eq. (14) an eigenvalue problem.

In the spirit of the Sturm–Liouville theory [19], we have tried to obtain bounds on

the eigenvalues—in our case on the four smallest Λ_i (two even and two odd). First, we explicitly display Λ in the factors h and k in Eq. (16) by

$$h = \eta(h^* + \Lambda),$$

and

$$k = B\Lambda. \tag{19}$$

Next, we define maxima and minima of all the coefficients such that

$$\begin{aligned} F_- &\leqslant f \leqslant F_+, \\ G_- &\leqslant g \leqslant G_+, \\ E_- &\leqslant \eta \leqslant E_+, \\ H^*_- &\leqslant h^* \leqslant H^*_+, \\ B_- &\leqslant B \leqslant B_+, \end{aligned}$$

and

$$C_- \leqslant c \leqslant C_+. \tag{20}$$

These upper case coefficients are all constants in s and can be used to define a set of associated eigenvalue problems—each one of the form

$$\begin{aligned} \frac{\partial}{\partial s}\left(F\frac{\partial X}{\partial s}\right) + E(H^* + \Lambda)X &= C\frac{\partial Z}{\partial s}, \\ \frac{\partial}{\partial s}\left(G\frac{\partial Z}{\partial s}\right) + B\Lambda Z &= -\frac{\partial}{\partial s}(CX). \end{aligned} \tag{21}$$

For this system the solutions are just $X = \cos(n\pi s/s_m)$ and $Z = \varepsilon \sin(n\pi s/s_m)$ for the X-even case. Analogously, the X-odd case has sin and cos solutions. The odd and even spectra are degenerate, so we need only study the X-even case. Substitutions of these solutions into Eq. (21) and the elimination of ε lead to a quadratic equation for Λ. We get

$$\Lambda = -\frac{\left(H^* - \dfrac{\alpha^2 F}{E} - \dfrac{\alpha^2 G}{B} \pm \left[H^* - \dfrac{\alpha^2 F}{E} - \left(\dfrac{\alpha^2 G}{B}\right)^2 + 4\alpha^2\left(\dfrac{C^2}{EB} + \dfrac{H^* G}{B} - \dfrac{\alpha^2 FG}{EB}\right)\right]^{1/2}\right)}{2} \tag{22}$$

where $\alpha = n\pi/s_m$.

For each mode—each choice for n—we establish bounding eigenvalues as follows. Every possible combination of the bounding constant coefficients is used to generate a pair of Λ's from Eq. (22). There are 64 of these pairs. We postulate the maximum and minimum of these sets to be bounding eigenvalues. Though we have attempted no

proof that these are indeed bounds, our experience has confirmed our assumption. Knowing these bounds gives one a place to start to find the eigenvalues of Eq. (14).

5. Algorithms for Stability Calculation

5.1. *Code Setup and Overview*

The stability code, STABGROW, is given the fields ψ, P, and P' on an r, z grid from a disk file produced by CYLEQ or HILLSV (analytic Hill's vortex equilibria). The routines from the tensor-produced spline interpolation code, TPSIC [20], are used to invert these arrays into the corresponding arrays of bi-cubic spline coefficients; these are checked to be precisely the same coefficients generated in CYLEQ to verify the consistency of the two codes. The right and left sides of Eq. (2) are evaluated and compared; we print out the relative error of these at every grid point to assure ourselves of a proper equilibrium. These errors are typically 10^{-8}.

Next, for the chosen flux lines, the trajectories of these lines are obtained from the Gear integration applied to Eq. (18b). A grid is superimposed on this line and the grid spacings are constructed to be proportional to B.

To determine the fastest growing modes we look for the lowest eigenvalues—these correspond to low poloidal mode numbers, i.e., $n = 0, 1, 2$. In doing this we first evaluate the extreme bounds given in the last section.

Then, in three passes over a field line, the code STABGROW does the following:

1. scans from the lower bound to upper bound using the finite difference integrator until it finds the first few approximate eigenvalues for both even and odd cases;

2. takes each approximate eigenvalue found above and applies secant-bisection iterations using the finite difference integrator and obtains more accurate eigenvalues; and

3. further refines these by using the Gear integrator with the secant-bisection iterations.

5.2. *Criterion for Eigenvalues*

The test for finding an eigenvalue depends on how well the trial solution fits the right-hand boundary conditions. We present the method used for X-even and Z-odd and note the other symmetry has an analogous method. At the left-hand boundary we have $X'(0) = 0$ and $Z(0) = 0$, but we cannot explicitly set both $X(0)$ and $Z'(0)$. One of them, or a combination, can be set to determine the norm of the eigenfunctions. We require

$$X^2(0) + Z'^2(0) = 1 \tag{23}$$

at the left-hand boundary to give the norm. To obtain the $X(0)$ and $Z'(0)$, we proceed as follows. Pick two pairs, $X_a(0)$, $Z'_a(0)$ and $X_b(0)$, $Z'_b(0)$, both satisfying Eq. (23).

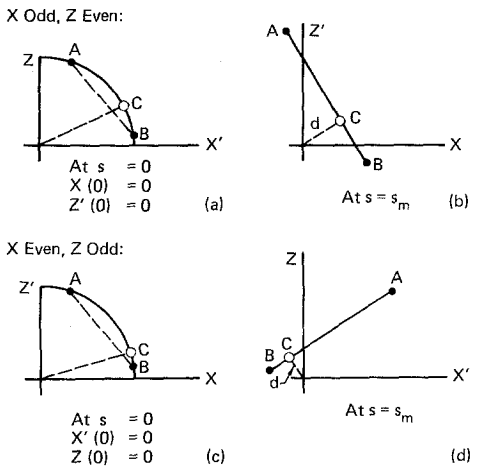

FIG. 2. Graphical description of the linear calculation to determine the ratio of Z/X' or Z'/X at $s = 0$. In (a) and (c) the possible initial conditions (A or B) lie on a circle for X-odd and X-even, respectively. In (b) and (d) the mappings to the Z', X and Z, X' planes at $s = s_m$ are given, respectively.

Use the integrator—finite difference or Gear—to obtain $X'(s_m)$, $Z(s_m)$ and $X'(s_m)$, $Z_b(s_m)$. Note in Fig. 2d we can represent these values in the Z, X' plane. Because $Z(s_m) = 0$, $X'(s_m) = 0$ is the desired condition for an eigensolution, we can get closest to this condition by taking a linear combination of X'_a, Z_a and X'_b, Z_b (shown as X'_c, Z_c in Fig. 2), which corresponds to minimizing the distance function d where $d \equiv \sqrt{X'^2_c + Z^2_c}$. Owing to the linear properties of our equations X'_c, Z_c lies on a straight line. That is,

$$X'_c(s_m) = \theta X'_a(s_m) + (1 - \theta)\, X'_b(s_m),$$

and

$$Z_c(s_m) = \theta Z_a(s_m) + (1 - \theta)\, Z_b(s_m). \tag{24}$$

Determining d and θ is a straightforward analytic geometry calculation. From this we find

$$X_c(0) = \theta X_a(0) + (1 - \theta)\, X_b(0),$$

and

$$Z'_c(0) = \theta Z'_a(0) + (1 - \theta)\, Z'_b(0). \tag{25}$$

This point lies on the chord shown in Fig. 2c. Because it no longer satisfies the norm given in Eq. (23) we scale $X_c(0)$ and $Z'_c(0)$ larger until it does. Point c on the circle then gives a better guess to the left-hand-side boundary conditions. Iterations are performed to move points a and b closer to c until the chord is a good approximation to the circle.

The distance vector **d** depends on Λ; we monitor both the phase and magnitude of **d** in our initial scan looking for eigenvalues. We increment upward until the phase of d changes $\sim\pi$ rad; this Λ_i value is then stored to be used later as an initial guess for the shooting method.

5.3. *The Secant-Bisection Shooting Procedure*

We have built an algorithm for finding the root Λ_i of $d(\Lambda_i)=0$ which enjoys the guaranteed convergence property of the bisection method and further has half the usual arithmetical operations per iteration required by Newton's method. To start, the user supplies two values Λ_A and Λ_B $(\Lambda_B > \Lambda_A)$ which may or may not bracket the root.

The algorithm is given by

$$
\begin{aligned}
&\text{(a) } W = 1.0 \text{ if } d(\Lambda_B)\cdot d(\Lambda_A) < 0,\ W = 2.0; \text{ otherwise}\\
&\text{(b) set } \Lambda = \Lambda_B - W d(\Lambda_B)\Big/\frac{d(\Lambda_B)-d(\Lambda_A)}{\Lambda_B-\Lambda_A};\\
&\text{(c) if } d(\Lambda_B)\cdot d(\Lambda_A) > 0 \text{ and } \Lambda < \Lambda_A, \text{ then set } \Lambda_A = \Lambda \text{ and } d(\Lambda_A) = d(\Lambda);\\
&\text{(d) if } d(\Lambda_B)\cdot d(\Lambda_A) > 0 \text{ and } \Lambda > \Lambda_B, \text{ then set } \Lambda_B = \Lambda \text{ and } d(\Lambda_B) = d(\Lambda);\\
&\text{(e) if } d(\Lambda_B)\cdot d(\Lambda_A) \leqslant 0 \text{ and } d(\Lambda)\cdot d(\Lambda_B) > 0, \text{ then set } \Lambda_B = \Lambda \text{ and } d(\Lambda_B) = d(\Lambda);\\
&\text{(f) if } d(\Lambda_B)\cdot d(\Lambda_A) \leqslant 0 \text{ and } d(\Lambda)\cdot d(\Lambda_B) \leqslant 0, \text{ then set } \Lambda_A = \Lambda \text{ and } d(\Lambda_A) = d(\Lambda).
\end{aligned}
\tag{26}
$$

The procedure is then repeated starting again at step a. It is iterated until $\Lambda_B - \Lambda_A$ is suitably small. Steps c and d are implemented only when the solution is not bracketed and using $W = 2.0$ has the effect of moving Λ_A and Λ_B farther apart until the solution is bracketed. Unlike the pure secant method, steps e and f always keep values Λ_A and Λ_B such that they bracket the solution. After some initial iterations where c or d are invoked until bracketing occurs, the algorithm continues all of its remaining iterations with steps e or f.

Stated in other words, steps e and f are the standard bisection algorithm logic, and step b generalizes the notion of picking the midpoint to that of using linear interpolation to find a point between Λ_A and Λ_B. And steps c and d help the process get started. Since evaluating $d(\Lambda_i)$ for a given Λ_i requires iterating the left-hand boundary condition in the manner described above, the secant-bisection iterations form an outer cycle or loop.

5.4. *Integrators*

We employ a fast finite difference integrator to solve Eq. (16) directly, or we use the more accurate but slower Gear solver to integrate the equivalent first-order system given in Eqs. (18a) and (18b). The Gear algorithm is adequately described elsewhere [17, 18]; suffice it to note that it can be made accurate to round-off error if it is given analytic (exact) coefficients.

To use the finite difference integrator, we first generate a gridded field line and define all of the coefficients on it. To get started, we use the Gear integrator on both

parts of Eq. (18b), which then gives the trajectory of the field line. On this line we put grid points spaced proportional to B. At each grid point location, we use the 2-D spline formula and its derivatives to specify all of the coefficients in Eq. (16). The finite difference version of Eq. (16) we use is

$$\frac{1}{\delta_{i-1/2}}\left[\frac{f_{i+1/2}(X_{i+1}-X_i)}{\delta_i}-\frac{f_{i-1/2}(X_i-X_{i-1})}{\delta_{i-1}}\right]+h_iX_i$$
$$=C_i\frac{Z_{i+1}-Z_{i-1}}{2\delta_{i-1/2}},$$

and

$$\frac{1}{\delta_{i-1/2}}\left[\frac{g_{i+1/2}(Z_{i+1}-Z_i)}{\delta_i}-\frac{g_{i-1/2}(Z_i-Z_{i-1})}{\delta_{i-1}}\right]+k_iZ_i$$
$$=\frac{C_{i+1}X_{i+1}-C_{i-1}X_{i-1}}{2\delta_{i-1/2}}, \tag{27}$$

where

$$\delta_i=s_{i+1}-s_i,$$
$$\delta_{i+1/2}=\frac{s_{i+2}-s_i}{2},$$
$$f_{i+1/2}=\frac{f_i+f_{i+1}}{2},$$

and

$$g_{i+1/2}=\frac{g_i+g_{i+1}}{2}.$$

This amounts to a 2×2 matrix equation for the pair X_{i+1}, Z_{i+1} at each grid point. Since they amount to initial value problems in the coordinates they are solved in one sweep across the gridded field line. We might note that δ_i are not the chord lengths but the actual arc lengths between grid points.

Using the Gear algorithm avoids the use of a gridded field line and in fact it generates the trajectory of the field line as it goes, picking points on it at whatever interval it requires to satisfy its error criterion (supplied by the user). The user-supplied subroutine DIFFUN which gives the integrator $\partial r/\partial s$, $\partial z/\partial s$, $\partial U/\partial s$, $\partial X/\partial s$, $\partial W/\partial s$, and $\partial Z/\partial s$ at arbitrary r, z locations actually calculates these "source" terms analytically from the spline formula for ψ and its derivatives. Thus, given a spline representation for ψ, it can calculate arbitrarily accurate eigenfunctions and eigenvalues—hence growth rates—limited of course by round-off error. Since the spline representation does not perfectly represent a true equilibrium, one need not demand more accuracy than is implicit in the equilibrium. However, particularly for elongated

flux surfaces and those near the separatrix, it is useful to have an integrator capable of representing very spiked eigenfunctions; the Gear spline method can resolve such spikes while the finite difference integrator will have zero-order errors there.

It has been pointed out that the 2-D fields calculated by our spline formula could be mapped onto the 1-D gridded field line and then resplined in 1-D [21]. The Gear method would then be faster because each spline evaluation in 1-D is four times faster than a 2-D evaluation; how this would effect the overall economy of the code has not been determined. A further advantage of the 1-D spline is that the r and z coordinates would not be needed once the 1-D grid was defined so Gear would need to integrate only four equations instead of six. No longer would arbitrarily accurate eigenfunctions result but the errors might be very small reflecting the discrepancy between the 1-D fit to the 2-D spline formula.

6. Results

6.1. *Code Verification with Hill's Vortex Configuration*

In recent work Newcomb has calculated exact analytic growth rates for elliptical Hill's vortex equilibria [22, 23]. He has found the growth time τ_v to be simply the Alfvén transit time around a field line for field lines near the vortex point. Moreover, he has determined a ψ-dependent factor, $g(\psi, a/b)$, to accurately calculate growth times on flux surfaces away from the vortex where the growth times are

$$\tau_g = g(\psi, a/b)\,\tau_v. \tag{28}$$

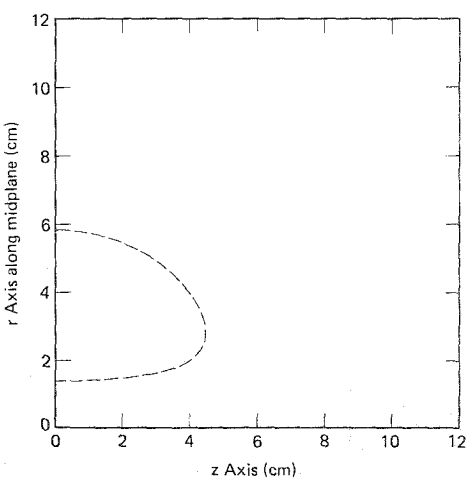

FIG. 3. Flux surface $\psi = 0.2\psi_v$ for the spherical Hill's vortex generated by STABGROW.

Here we indicate that g also depends on the elongation a/b given by the ratios of the axial to radial extent of the separatrix. The Hill's vortex equilibrium is given analytically by

$$\psi = \frac{B_0 r^2}{2}\left(-1 + \frac{r^2}{b^2} + \frac{z^2}{a^2}\right) \quad \text{and} \quad P = \frac{-B_0(4a^2 + b^2)\psi}{a^2 b^2}. \tag{29}$$

Since these equilibria are known by these formulae we need not employ the equilibrium code to generate these fields. Rather we have built a small code, HILLSV, which evaluates these fields on the grid and produces an output disk file of the same form as that generated by CYLEQ. This disk file is read by STABGROW.

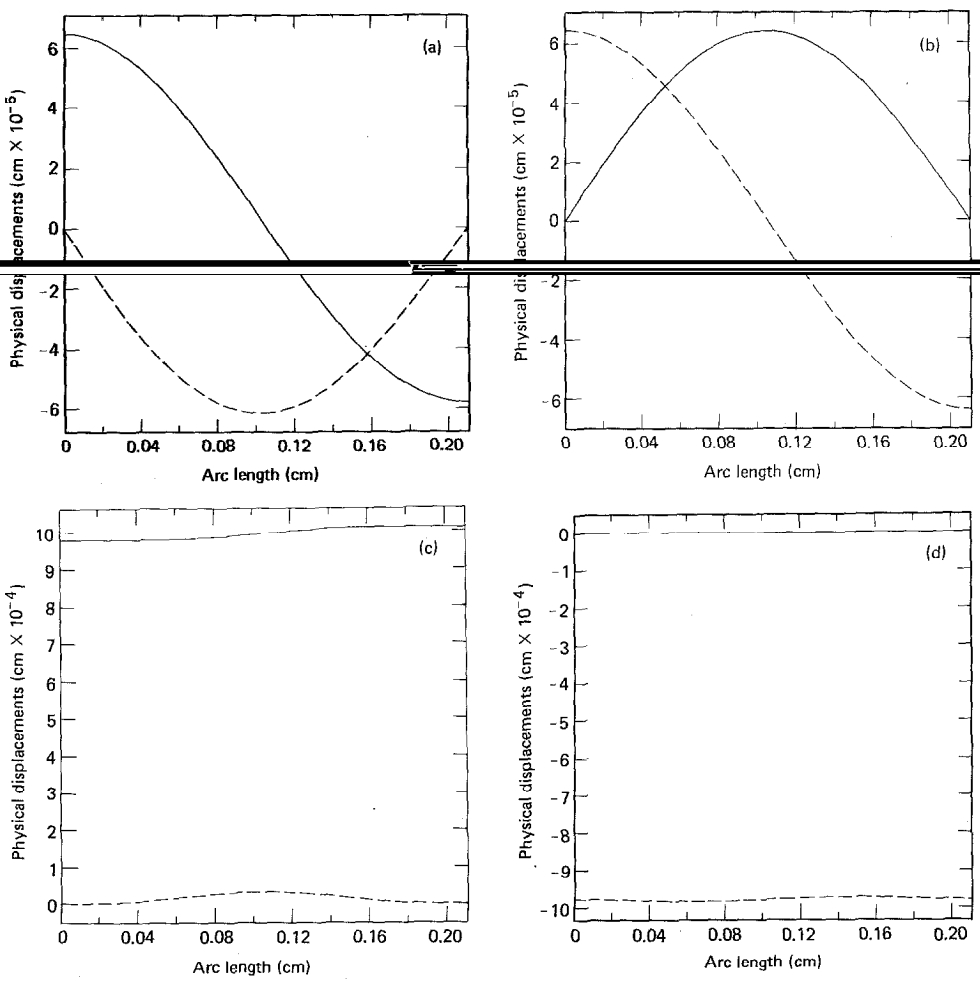

FIG. 4. For a circular flux line near the vortex, the lowest symmetric and antisymmetric displacements ξ_ω are shown by the solid lines in (a) and (b). In (c) and (d) the next highest symmetric and antisymmetric displacements are shown. The broken lines give the ξ_χ displacements.

The most tractable configuration for analysis is the elliptical Hill's vortex with $a/b = \frac{1}{2}$, because it has circular flux lines near the vortex. This means the coefficients in Eq. (16) are all constants. Hence, the most unstable eigenvalue—and its associated growth rate—is given by Eq. (22) for $n = 1$. In fact, the even and odd modes are degenerate here and physically correspond to rigid displacements of the circular flux tube, radially and axially, respectively. The result is $\gamma = 2\pi/\tau_v$. Newcomb's analysis was more general, giving this result for elliptical flux tubes near the vortex.

We then chose three flux-line configurations to check the validity of the code:

1. For the oblate elliptical Hill's vortex $a/b = \frac{1}{2}$, we chose the $\psi = 0.999\psi_v$ flux surface (a circle), where ψ_v is the flux function at the vortex.

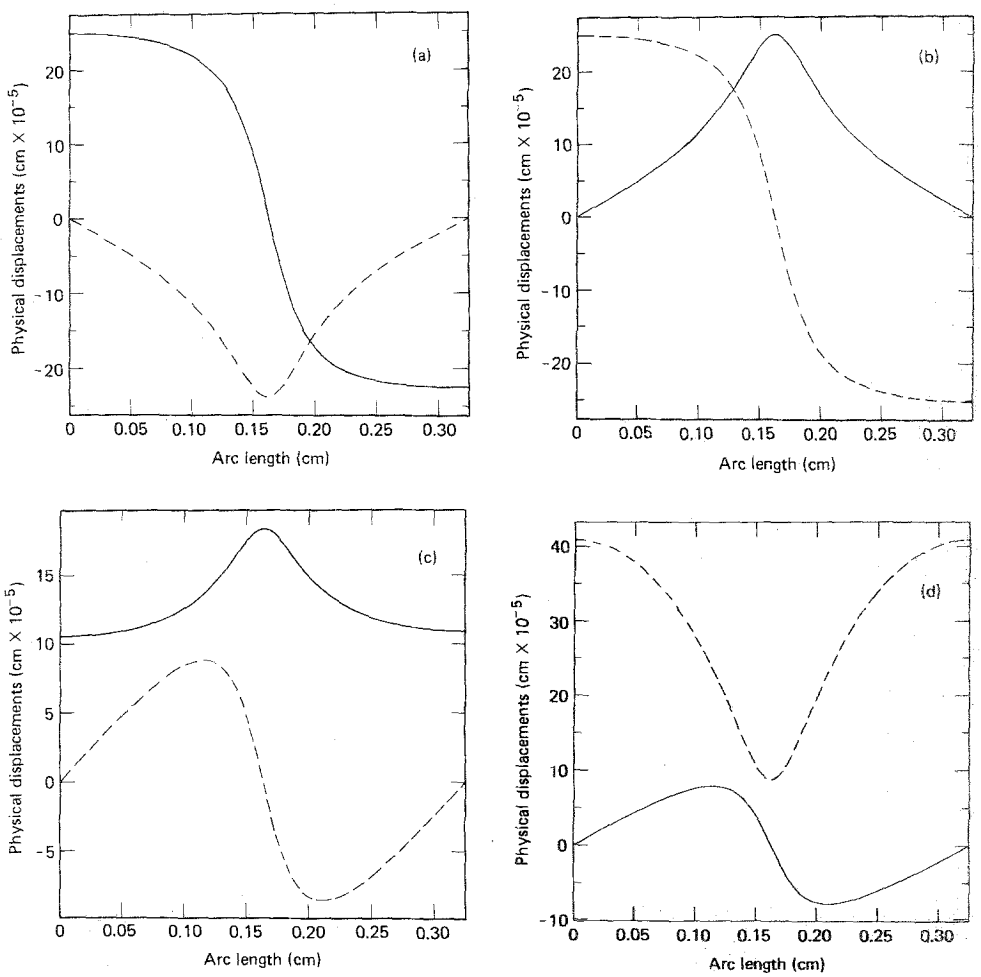

FIG. 5. For an elliptical flux line near the vortex the displacements are shown analogous to those of Fig. 4.

2. For the spherical Hill's vortex $a/b = 1$, we again use the $\psi = 0.999\psi_v$ flux surface (an ellipse of elongation = 2).

(3) For this same spherical Hill's vortex, we use the flux surface $\psi = 0.2\psi_v$, which is a complicated curve away from the vortex near the separatrix. Figure 3 displays this flux surface.

The growth rates for the most unstable mode obtained by STABGROW are compared to the exact ones in Table I.

Eigenfunctions corresponding to the three cases in Table I are shown in Figs. 4, 5, and 6, respectively. Each figure displays the eigenfunctions of the two lowest even modes and the two lowest odd modes. We have shown the actual displacements ξ_ψ (solid line) and ξ_χ (broken line) rather than the normalized functions X and Z. In Figs. 4a and b we see the lowest eigenfunctions of the circular flux line near the

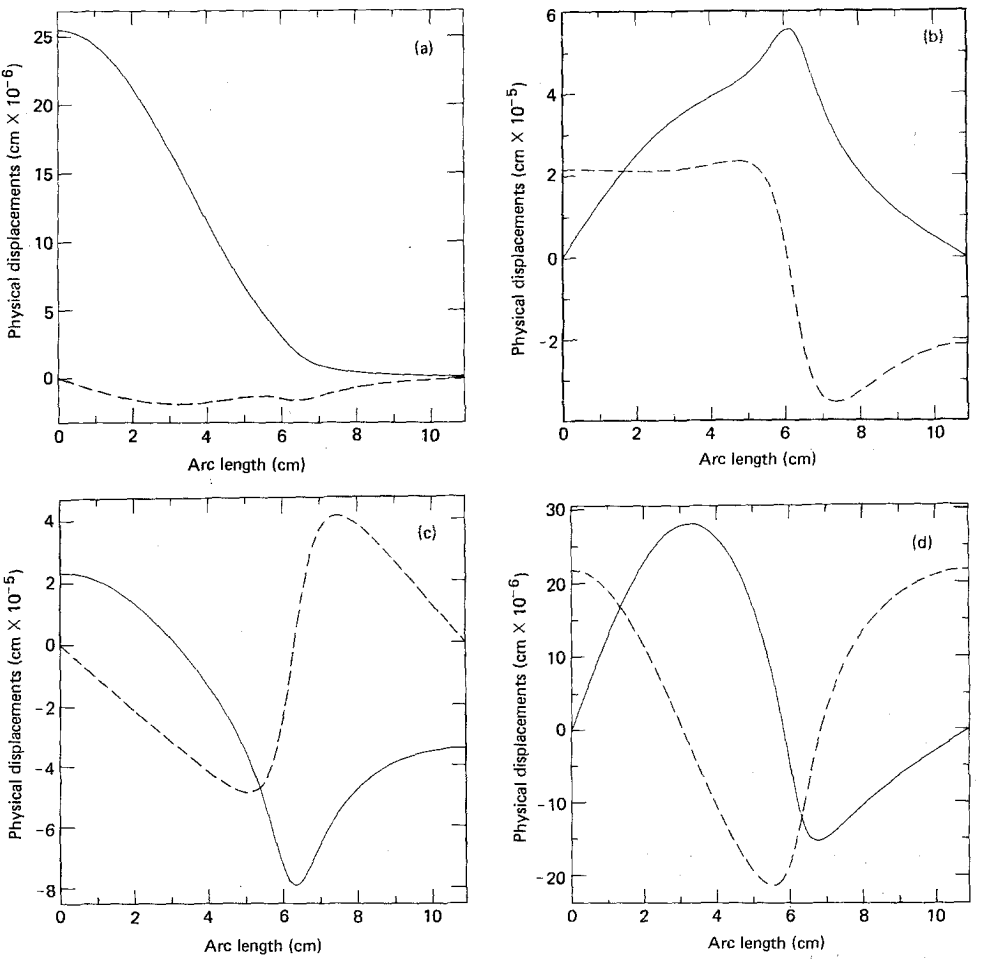

FIG. 6. Displacements for the flux line plotted in Fig. 3.

TABLE I

Growth Rates of Hill's Vortex Equilibria for the Most Unstable Mode Calculated by STABGROW vs Those Obtained Analytically

Case	Analytic γ of Newcomb	Numerical γ of STABGROW
$a/b = \frac{1}{2}$; $\psi = 0.999\psi_v$	4.6437×10^6	4.6426×10^6
$a/b = 1$; $\psi = 0.999\psi_v$	2.9367×10^6	2.9362×10^6
$a/b = 1$; $\psi = 0.2\psi_v$	9.5617×10^6	9.5594×10^6

vortex take the very simple sinusoidal form we obtained from Eq. (21). The next higher ones, shown in Figs. 4c and d, represent a pure interchange and a pure parallel displacement mode ($\gamma = 0$). The corresponding plots in Fig. 5 show the distortion introduced when an elliptical flux line is near the vortex. Even for this ellipse of elongation = 2.0, the eigenfunctions are very steep near the point of maximum field-line curvature. Finally, in Fig. 6 the eigenfunctions of the $\psi = 0.2\psi_v$ flux line are displayed.

6.2. *Studies of More Realistic Equilibria—The FRX-B Calculations*

Configurations have been obtained from the equilibrium code, CYLEQ, which agree well with measured profiles in the FRX-B (field-reversed theta pinch) experiment at Los Alamos [24]. Figure 7 shows some of the flux surfaces obtained in this calculation. This equilibrium was used in other stability studies as described in the Introduction.

For three typical flux lines the most unstable modes were calculated in STABGROW. Unlike the Hill's vortex equilibria (where P' is constant) here P' varies

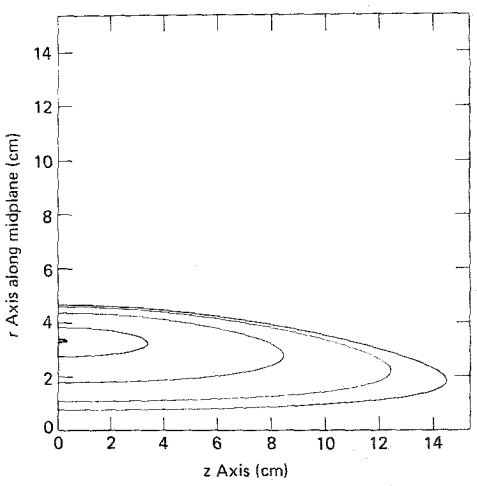

FIG. 7. Flux surfaces $\psi = (0.999, 0.9, 0.5, 0.2, 0.1)\,\psi_v$ are shown for the FRX-B equilibrium generated by CYLEQ.

TABLE II

Growth Rates of CYLEQ Equilibria for the Most Unstable Mode Calculated by STABGROW for Three Typical Flux Lines

Case	Numerical γ of STABGROW
FRX-B; $a/b = 3.3$; $\psi = 0.99\psi_v$	8.99×10^5
FRX-B; $a/b = 3.3$; $\psi = 0.90\psi_v$	Stable
FRX-B; $a/b = 3.3$; $\psi = 0.50\psi_v$	3.19×10^5

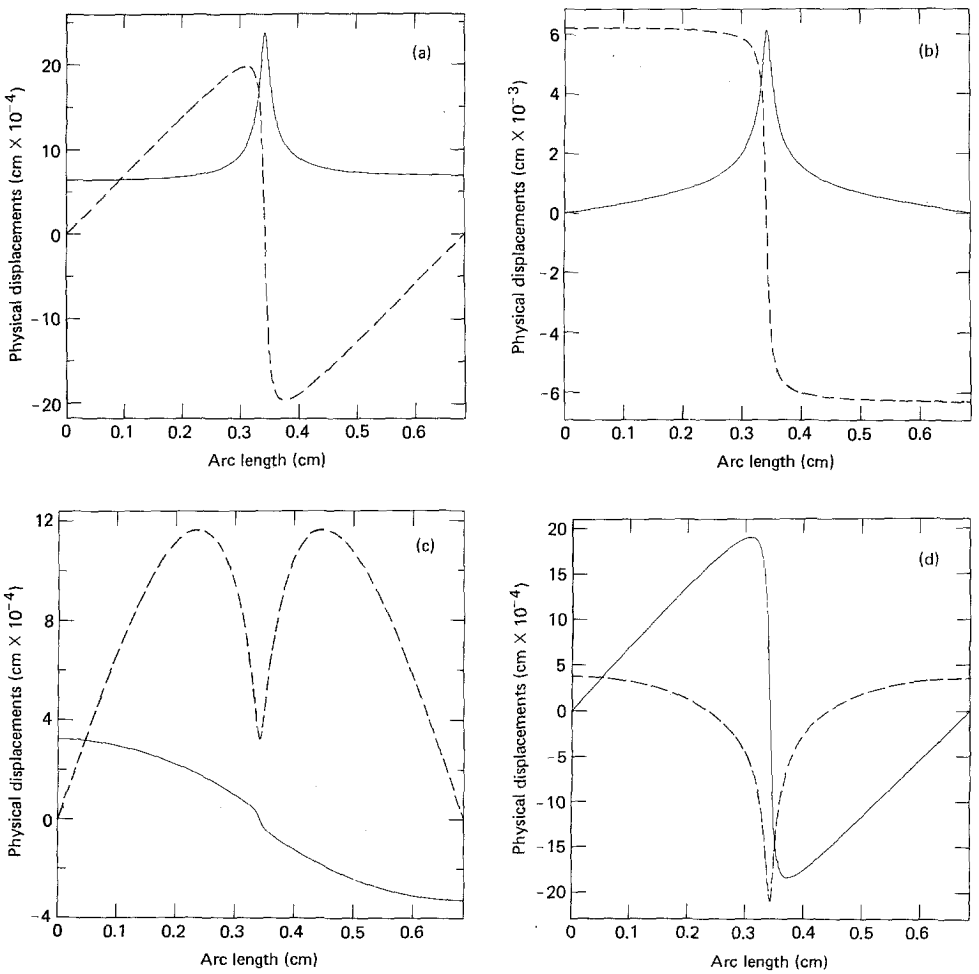

FIG. 8. Displacements for the $\psi = 0.99\psi_v$ flux surface for the FRX-B equilibrium are shown. The nomenclature is the same as that in Fig. 4.

and is small near the $\psi = 0.90\psi_v$ flux surface. The destabilizing term is then small enough to allow stability on this flux surface. Table II shows the results.

For these numerically generated equilibria modeling FRX-B, we find the curvature is quite large at the axial end of each flux line. It is not surprising then to see the very spiked profiles of the eigenfunctions, shown in Fig. 8, where we plot the results obtained for the $\psi = 0.99\psi_v$ flux surface.

7. Conclusions

We have learned from the theory and our calculations that Hill's vortex equilibria are always unstable on every flux line. The more realistic FRX-B equilibria are found to be unstable on some flux lines—particularly near the vortex—but stable on some intermediate lines.

In spite of the pessimistic results generated by this model we can still obtain useful information about growth rates. If a plasma is MHD unstable it still may be made stable by other effects such as finite Larmor radius modifications. An understanding of what sort of pressure profiles lead to configurations of low growth rates and to ones with large stable regions may also be obtained from studies made with CYLEQ one may find ideal MHD stability in the center of the region between the separatrix and the vortex. Near the separatrix and near the vortex kinetic effects are expected to stabilize the ideal instabilities we find there.

It may be possible to extend this stability model to better represent the kinetic stabilizing mechanisms without having to abandon the analysis of separate flux lines. However, it may be unwise to proceed in that direction; rather effort in the use of spectral models, such as those embodied in the GATO code [25, 26], might give one the desired results in a more timely and physical manner with less uncertain approximations.

References

1. A. Eberhagen and W. Grossman, *Z. Phys.* **22** (1971), 130.
2. A. G. Eskov *et al.*, *in* "Proceedings, 7th International Conference on Plasma Physics and Controlled Thermonuclear Research, Innsbruck, August 1978," Intern. At. Energy Agency, Vienna, 1979.
3. R. K. Linford *et al.*, *in* "Proceedings, 7th International Conference on Plasma Physics and Controlled Thermonuclear Research, Innsbruck, August 1978," Intern. At. Energy Agency, Vienna, 1979.
4. I. B. Bernstein, E. A. Frieman, R. M. Kulsrud, and M. O. Kruskal, *Proc. Soc. London Ser. A* **244** (1958), 17.
5. D. V. Anderson, J. Killeen, and M. E. Rensink, *Phys. Fluids* **15** (1972), 351.
6. D. A. D'Ippolito, E. A. Adler, and Y. C. Lee, *Phys. Fluids* **23** (1980), 794.
7. D. Dobrott, D. B. Nelson, J. M. Greene, A. H. Glasser, M. S. Chance, and E. A. Frieman, *Phys. Rev. Lett.* **39** (1977), 943.
8. D. C. Barnes and J. U. Brackbill, *Nucl. Sci. Eng.* **16** (1977), 18.

9. D. C. BARNES AND D. V. ANDERSON, *Phys. Rev. Lett.* **46** (1981), 1337.
10. D. D. SCHNACK AND J. KILLEEN, *J. Comput. Phys.* **35** (1980), 110.
11. A. I. SHESTAKOV, D. D. SCHNACK, AND J. KILLEEN, *in* "Proceedings, Symposium on Compact Toruses and Energetic Particle Injection, Princeton, 1979."
12. J. KILLEEN, D. D. SCHNACK, AND A. I. SHESTAKOV, *in* "Proceedings, 4th IRIA International Symposium on Computing Methods in Applied Sciences and Engineering, Versailles, 1979"; also as Lawrence Livermore National Laboratory Report UCRL-83332 (1979).
13. D. S. KERSHAW, *in* "Proceedings, 8th Conference on Numerical Simulation of Plasmas, Monterey, California, 1978."
14. J. A. MEIJERINK AND H. A. VAN DER VORST, *Math. Comput.* **31** (1977), 148.
15. D. S. KERSHAW, *J. Comput. Phys.* **26** (1978), 43.
16. D. V. ANDERSON AND T. A. CUTLER, *in* "Proceedings, 8th Conference on Numerical Simulation of Plasmas, Monterey, California, 1978."
17. C. W. GEAR, "Numerical Initial Value Problems in Ordinary Differential Equations," Prentice–Hall, Englewood Cliffs, N.J., 1971.
18. A. C. HINDMARSH, "GEAR, an Ordinary Differential Equation Solver," Lawrence Livermore Laboratory Report UCID-30001, Rev. 3, Livermore, Calif., 1974.
19. E. L. INCE, "Ordinary Differential Equations," Dover, New York, 1956.
20. D. V. ANDERSON, "The Three-Dimensional Tensor Product Spline Interpolation Code (TPSIC)," Lawrence Livermore National Laboratory Report UCRL-51958 Livermore, Calif., 1976.
21. L. D. PEARLSTEIN, Lawrence Livermore National Laboratory, Livermore, Calif., private communication.
22. W. A. NEWCOMB, "MHD Instability in the Neighborhood of the Vortex Point of a Field-Reversed Magnetic-Mirror System," Lawrence Livermore National Laboratory Report UCRL-83867 Livermore, Calif., 1980, submitted for publication.
23. W. A. NEWCOMB, Lawrence Livermore National Laboratory, Livermore, Calif., private communication.
24. W. T. ARMSTRONG, R. K. LINFORD, J. LIPSON, D. A. PLATTS, AND E. G. SHERWOOD, *in* "Proceedings, U.S.-Japan Joint Symposium on Compact Toruses and Energetic Particle Injection, Princeton, New Jersey, December 1979."
25. L. C. BERNARD, F. J. HELTON, AND R. W. MOORE, *in* "Proceedings, 9th Conference on Numerical Simulation of Plasmas, Evanston, Illinois, June 1980."
26. H. E. DALHED, JR., Lawrence Livermore National Laboratory, Livermore, Calif., private communication.