Computation of Magnetic Coordinates
and Action-Angle Variables

A. H. REIMAN AND N. POMPHREY

Princeton Plasma Physics Laboratory, Princeton, New Jersey 08543

We have developed a new algorithm for calculating magnetic surfaces and coordinates for a given three-dimensional magnetic field. The algorithm serves also to solve the equivalent problem of computing invariant tori and action-angle variables for a one-dimensional time-dependent numerically specified Hamiltonian (or a two-dimensional time-independent Hamiltonian). Our approach combines features of both iterative and trajectory following methods. This allows us to overcome the inefficiency of trajectory following methods near low order rational surfaces, while retaining some of the robustness of these methods. © 1991 Academic Press, Inc.

I. INTRODUCTION

A magnetic coordinate system for a given magnetic field is a set of toroidal coordinates providing a Clebsch representation for the field. These coordinates play a fundamental role in the study of magnetic fields for plasma confinement. They correspond to action-angle variables for the Hamiltonian flow defined by the magnetic field lines. In this paper, we discuss the numerical computation of magnetic coordinates for a given three-dimensional magnetic field. This corresponds to the numerical computation of action-angle variables for a one-dimensional, time-dependent (or two-dimensional, time-independent) numerically specified Hamiltonian.

Algorithms in use for the calculation of magnetic coordinates in three dimensions [1–4] reconstruct the coordinates from information obtained by following magnetic field lines. These methods are inefficient when the rotational transform (winding number of the Hamiltonian) is near a low order rational. The field line must then be followed over a great distance to sample adequately the corresponding flux surface. Iterative methods of sampling the flux surface naturally suggest themselves. Iterative methods, however, are sensitive to the initial guess, and can display instabilities, in contrast to the field line following methods, which are robust. We have developed a hybrid approach which uses information obtained by field line following both in the formulation of an iteration procedure and the provision of initial conditions for the iteration. This appears to combine the best features of both methods.

225

In other contexts, the numerical calculation of action-angle variables has been an active field of research, with a large literature. The primary applications have been to the semiclassical quantization of atomic and molecular systems. The focus here has been on the calculation of the energy levels corresponding to the quantized levels of the action, with the action-angle variables themselves generally of secondary interest. A review of this area can be found in Refs. [5, 6]. References [7, 8] contain a discussion of more recent work. The numerical computation of action-angle variables has also been studied in the context of applications to dynamical problems in astrophysics [9, 10] and for application to high energy particle accelerators [11]. There has also been closely related work on the theory of area preserving mappings [12, 13].

Generally speaking, there have been two different approaches to the problem of numerically calculating action-angle variables, trajectory following and iterative. (We do not include here the adiabatic switching approach [14], which locates quantizing trajectories without actually solving for the action-angle variables.) Trajectory following methods integrate Hamilton's equations to determine the trajectories, and use information about the trajectories to construct action-angle variables. A widely used method that has been developed independently in the context of plasma applications [1, 2, 4], astrophysical applications [9], and applications to semiclassical quantization [15, 16] employs a Fourier decomposition of the coordinates along the trajectories. Identification of the discrete frequencies in these spectra, along with a calculation of the corresponding spectral amplitudes, allows a direct determination of the action-angle variables. This Fourier approach has the attractive feature that it uses all of the information obtained in following trajectories (field lines), and thus minimizes the distance over which the computationally expensive trajectory integrations must be performed.

Shenker and Kadanoff [13] have introduced a similar method for studying area preserving maps. They construct a transformation to a coordinate system where the map becomes a twist map, analogous to the transformation of a Hamiltonian system to action-angle variables. The transformation is computed by performing Fourier decompositions along a set of periodic orbits which successively approximate the trajectory of interest. By using a sequence of periodic orbits, this approach avoids the need for a window function in the Fourier decomposition along the trajectory. The approach is otherwise very close to the Fourier methods cited above.

Methods that use trajectory information suffer from the fundamental limitation that they must follow the trajectories sufficiently far to see the longest periodicities in the orbit. That is, the orbit must adequately sample the surface of the invariant toroid. In following the field lines of a magnetic field, the lines must be followed sufficiently far to sample adequately the corresponding flux surface. When the winding number of the Hamiltonian (rotational transform of the magnetic field) is nearly rational, the orbit is nearly periodic. This is illustrated in Fig. 1, which schematically shows a return map on a near-rational surface. The numbers in the figure label successive intersections of the trajectory with the plane of the figure. Only after
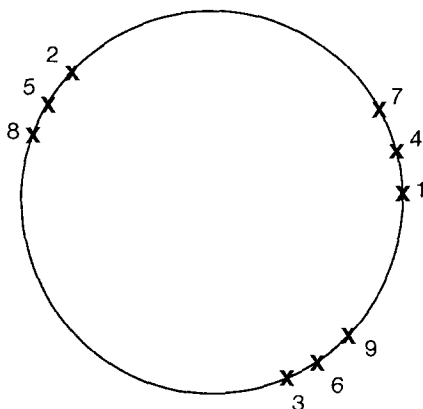
FIG. 1. Sketch of the return map on a near-rational surface. The numbers label successive inter-sections of the trajectory with the plane of the figure.

many returns of these nearly periodic orbits does the trajectory sample the whole of the invariant toroid. This qualitative picture can be made precise by determining the accuracy of the calculated action-angle variables as a function of the distance over which trajectories are followed [4]. The distance required to maintain a fixed accuracy goes to infinity as a rational surface is approached. (Of course, only a finite number of Fourier modes is retained in any numerical calculation, so that in practice it is only the low order rationals that pose difficulties.)

The other approach to calculating action-angle variables has been iterative. (The perturbative methods can be viewed as a special case of the iterative ones. The iterative methods, in fact, often reduce to perturbative ones in appropriate limits.) Iterative methods do not suffer the same degradation in efficiency near rational sur-faces as trajectory following methods. This is illustrated in Fig. 5, which shows the number of magnetic field evaluations required to calculate action-angle variables by trajectory following and iterative methods for a series of cases approaching a resonance. The details of this calculation will be discussed in Section V.

In contrast to the field line following methods, which are robust, iterative methods display numerical instabilities. The stability properties can be sensitive to the initial guess for the action-angle variables. We have developed a new code which uses information obtained by field line following both in the formulation of an iteration procedure and the provision of initial conditions for the iteration. This appears to combine the best features of both methods.

The iterative part of our approach is close to the approach of Refs. [10, 11, 17–19] for calculating action-angle variables. A similar iterative procedure was also used by Percival in Ref. [12] to study the breakup of a KAM surface for an area-preserving map. The Newton's method we study is also closely related to the super-convergent schemes for determining action-angle variables in Fourier space which serve as the basis for KAM proofs. We Fourier decompose Hamilton's equations with respect to the angular and time coordinates to obtain a set of coupled, non-

linear, algebraic equations. These are the equations to which we apply iterative methods of solution, solving for the Fourier components of the transformation from action-angle variables to the initial coordinate system.

To enhance the stability of the iteration procedure, and to speed its convergence, we can eliminate some of the Fourier components of Hamilton's equations by making use of information obtained in following trajectories. Following trajectories over a relatively short distance, we determine a set of linear relations between the Fourier components. These can be used to eliminate some of the nonlinear equations in favor of linear equations, and to provide a good initial guess for the iterative scheme. We find that Picard iteration is not sufficiently stable to form the basis of a useful iterative scheme, but that Newton's method does serve our purposes.

Throughout this paper we will be studying only integrable trajectories. Only for such trajectories do invariant tori and action-angle variables exist.

Section II describes the relation between magnetic coordinates and action-angle variables. Section III discusses trajectory following methods and their degradation in efficiency near low order rational surfaces. In Sections IV and V we describe a series of numerical experiments in which we test iterative methods for the calculation of action-angle variables, both alone and in combination with trajectory following methods. Section IV discusses Picard iteration, and Section V discusses Newton's method. In Section VI we use information obtained in our numerical experiments to derive general estimates of the efficiency of iterative methods. Finally, Section VII discusses and summarizes our results.

II. MAGNETIC COORDINATES AND ACTION-ANGLE VARIABLES

In this section we describe the relation between magnetic coordinates and action-angle variables. The magnetic field is a three-dimensional Hamiltonian vector field. Integral curves (trajectories) of the Hamiltonian correspond to magnetic field lines. Integrable trajectories lie on invariant toroids in phase space. The corresponding surfaces for magnetic fields are called flux surfaces. Action-angle variables for the invariant toroids correspond to magnetic coordinates for the flux surfaces. Various correspondences between Hamiltonian systems and three-dimensional magnetic fields are summarized in Table I.

In magnetic coordinates with the radial coordinate normalized to the enclosed toroidal flux, the magnetic field **B** can be written in covariant form as [20, 21]

$$\mathbf{B} = \nabla \psi \times \nabla \theta + \mathit{\iota} \nabla \phi \times \nabla \psi, \tag{1}$$

where $\mathit{\iota}(\psi)$ is the rotational transform (winding number or twist) of the magnetic field line on a given ψ surface, and θ and ϕ are, respectively, poloidal and toroidal angles. The covariant expression for **B** is closely related to the Clebsch form

$$\mathbf{B} = \nabla \psi \times \nabla (\theta - \mathit{\iota} \phi). \tag{2}$$

TABLE I

Correspondence between Various Quantities in the
Language of Magnetic Coordinates and the Language
of Hamiltonian Mechanics

Hamiltonian systems	Magnetic fields
Trajectory	Magnetic field line
Invariant toroid	Flux surface
Elliptic fixed point	Magnetic axis
Time t	Toroidal angle ϕ
Action J	Toroidal flux ψ
Angular coordinate θ	Poloidal angle θ
Hamiltonian H	Poloidal flux ψ_p
Winding number ω	Rotational transform ι

To bring out the Hamiltonian structure of the magnetic field, we define a poloidal flux function $\psi_p(\psi)$ which satisfies

$$d\psi_p/d\psi = \iota(\psi).$$ (3)

The equations for the magnetic field line trajectories take the form

$$d\psi/d\phi = B^\psi/B^\phi = -\partial\psi_p/\partial\theta = 0,$$
$$d\theta/d\phi = B^\theta/B^\phi = \partial\psi_p/\partial\psi = \iota(\psi).$$ (4)

These are Hamilton's equations in action-angle variables, where ψ_p is the Hamiltonian, ψ is the action variable, θ is the angle variable, and ϕ plays the role of time. The contravariant magnetic field component B^ψ is defined by

$$B^\psi = \mathbf{B}\cdot\nabla\psi,$$ (5)

with similar definitions holding for B^θ and B^ϕ.

Computational details of a method for determining action-angle variables by Fourier decomposing the coordinates along trajectories are presented in Section III. In this method, one seeks a Fourier decomposition of the Cartesian (physical) coordinates as a function of magnetic coordinates,

$$\mathbf{x}(\psi, \theta, \phi) = \sum_{nm} \mathbf{x}_{nm}(\psi)\exp[i(n\phi - m\theta)].$$ (6)

In the language of Hamiltonian mechanics, the function $\mathbf{x}(\psi, \theta, \phi)$ represents a ("time-dependent") transformation from action-angle variables (ψ, θ, ϕ) to a laboratory coordinate system represented by the vector $\mathbf{x}$. Since the trajectory of

a field line in magnetic coordinates is $\theta = \jmath\phi + \theta_0$, Fourier decomposition of the coordinates x along a field line assumes the form

$$\mathbf{x} = \sum_{nm} \mathbf{x}_{nm}(\psi) \exp[i(n - \jmath m)\phi]. \tag{7}$$

The identification of the mode numbers, n and m, associated with each of the peaks in the one-dimensional spectrum completes the determination of the transformation to magnetic coordinates from trajectory data. Details are summarized in Section III.

An alternative method for determining magnetic coordinates is suggested by rewriting the canonical form of **B**, Eq. (1), in contravariant form,

$$\mathbf{B} = \frac{1}{\mathcal{J}} \left(\frac{\partial \mathbf{x}}{\partial \phi} + \jmath \frac{\partial \mathbf{x}}{\partial \theta} \right), \tag{8}$$

where

$$\mathcal{J} = \frac{\partial \mathbf{x}}{\partial \psi} \cdot \frac{\partial \mathbf{x}}{\partial \theta} \times \frac{\partial \mathbf{x}}{\partial \phi} = 1/B^{\phi} \tag{9}$$

is the Jacobian of the transformation from Cartesian to magnetic coordinates. A Fourier transformation gives

$$(n - \jmath m)\, \mathbf{x}_{nm} = (\mathcal{J}\mathbf{B})_{nm}, \tag{10}$$

where n is the toroidal mode number and m is the poloidal mode number. The iterative solution of Eq. (10), supplemented by the condition

$$\jmath = (B^{\theta}/B^{\phi})_{00} = (\mathcal{J}B^{\theta})_{00} \tag{11}$$

for the winding number $\jmath$, forms the basis for an alternative calculation of magnetic coordinates and will be discussed in detail in Sections IV, V, and VI.

In the language of Hamiltonian mechanics, the "time rate of change" of the laboratory coordinates along a trajectory is given by

$$\frac{d\mathbf{x}(\psi, \theta, \phi)}{d\phi} = \frac{\partial \mathbf{x}}{\partial \phi} + \frac{\partial \mathbf{x}}{\partial \theta} \frac{d\theta}{d\phi} = \frac{\partial \mathbf{x}}{\partial \phi} + \jmath \frac{\partial \mathbf{x}}{\partial \theta}. \tag{12}$$

Thus, Eq. (8) expresses Hamilton's equations in laboratory coordinates.

A number of test cases are used in this paper to study properties of the algorithms for calculating magnetic coordinates. The test case we use most often corresponds to an anharmonic oscillator Hamiltonian,

$$H = (1/2)(p^2 + \jmath_0^2 q^2) + \beta_3 q^3 + \beta_4 q^4, \tag{13}$$

where β_3 and β_4 are adjustable parameters. The Hamiltonian is time independent, corresponding to a two-dimensional magnetic field, independent of the toroidal

angle ϕ. An equivalent magnetic field in a periodic cylinder is obtained by identifying q and p with stretched Cartesian coordinates and transforming to polar coordinates r, θ:

$$q = \ell_0^{-1/2} x; \qquad x = r \cos \theta \qquad\qquad (14)$$

$$p = \ell_0^{1/2} y; \qquad y = r \sin \theta. \qquad\qquad (15)$$

Identifying H with the poloidal flux function ψ_p, and making the replacement $\alpha_3 = \ell_0^{-5/2} \beta_3$ and $\alpha_4 = \ell_0^{-3} \beta_4$, yields

$$\psi_p = \ell_0(\tfrac{1}{2} r^2 + \alpha_3 r^3 \cos^3 \theta + \alpha_4 r^4 \cos^4 \theta), \qquad\qquad (16)$$

in terms of which the magnetic field is

$$\mathbf{B} = R_0^2 \nabla \phi + \nabla \phi \times \nabla \psi_p. \qquad\qquad (17)$$

For $\alpha_3 = \alpha_4 = 0$ the flux surfaces are circular. If either of these parameters are non-zero, then the flux surfaces are distorted, and, with an appropriate choice of α_3 and α_4, a separatrix can be introduced. In the calculations to be presented, the outer edge of the plasma corresponds to a surface through the point $r = 1$ at $\theta = 0$. It is on this surface that convergence properties of the algorithms are investigated.

III. A Trajectory Following Method and Its Efficiency

In this section we estimate the efficiency of a trajectory following method [1, 2, 4, 9, 15, 16] for obtaining action-angle variables. We will see that the method becomes inefficient when the winding number is near a low order rational. We will also see that the difficulty is not restricted to the described method, but is a fundamental limitation of all trajectory following methods. This will motivate our consideration of iterative methods, both alone and in combination with trajectory following.

We wish to obtain a Fourier decomposition of the Cartesian coordinates as a function of magnetic coordinates, as expressed by Eq. (6). The correspondence with Eq. (7) allows us to determine the required Fourier coefficients from a Fourier decomposition of the coordinates along a field line. (For details of how we make this determination, see Ref. [4].) Equation (7) shows that, in the limit, a Fourier transform along a field line from $\phi = -\infty$ to $\phi = \infty$ consists of a set of delta functions. In practice we follow the field lines a finite number of times around the torus, and the spectral lines become envelopes of finite width. Figure 2 shows a Fourier transform of the Cartesian x coordinate along a field line for a stellarator magnetic field (described in Section V and Appendix A). A portion of the spectrum has been magnified to show the envelopes more clearly. The width of the envelopes is determined by the distance over which the field lines are followed. If we wish to reconstruct $\mathbf{x}(\psi, \theta, \phi)$ from the information along the field line, we must follow the
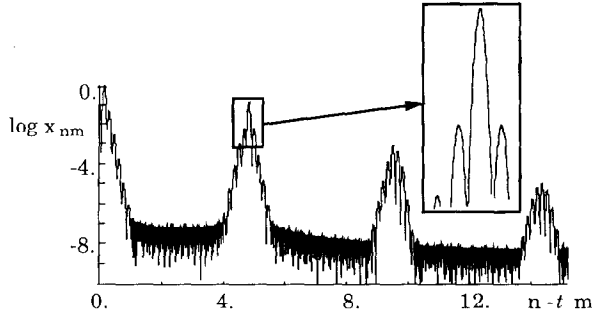
REIMAN AND POMPHREY

FIG. 2. Fourier transform of the x Cartesian coordinate along a field line as a function of ϕ. The magnetic field is that for the ATF stellarator described in Section V and Appendix A, with the field line lying on the outer flux surface ($\jmath = 0.975$).

field line sufficiently far to resolve the major lines of the Fourier spectrum. The distance over which the field line must be followed determines the computer time expended in calculating the magnetic coordinates by this method.

The overlap of the spectral envelopes, known as "spectral leakage," limits the accuracy with which the x_{nm}'s can be determined. To control the shape of the spectral envelopes, we follow the practice widely used in the Fourier transformation of time series of multiplying by a window function before Fourier transforming [22, 23]. In the absence of the window function, a Fourier transform of $x(\phi)$ produces envelopes with slowly decaying side-lobes. This is the well-known Gibb's phenomenon, which appears here because x is not a periodic function of ϕ. (The winding number $\jmath$ is generally an irrational number.)

The use of a Gaussian window has the advantage of allowing an explicit calculation of spectral leakage [4]. If k denotes a particular wave number, then multiplying $x(\phi)$ by the Gaussian window $\exp[-(\phi/\phi_g)^2]$ and Fourier transforming from $\phi = -\phi_f$ to $\phi = \phi_f$ obtains a Fourier spectrum in which the spectral envelope corresponding to x_{nm} has a tail of amplitude

$$x_{nm}\left\{\exp\left[-\frac{1}{4}(\Delta k\phi_g)^2\right] - \frac{1}{\sqrt{\pi}}\frac{\cos(\Delta k\phi_f)\exp[-(\phi_f/\phi_g)^2](\phi_f/\phi_g)}{(\phi_f/\phi_g)^2 + (\Delta k\phi_g)^2/4}\right\} \quad (18)$$

at any given value of k, where

$$\Delta k = (n - \jmath m) - k. \quad (19)$$

The first term in Eq. (18) is small if ϕ_g is sufficiently large. If the magnetic surface is nearly rational so that Δk is small, then ϕ_g must be especially large. The condition that the second term in Eq. (18) be small requires that ϕ_f/ϕ_g be sufficiently large. Therefore, on a magnetic surface that is nearly rational, the required field line distance, ϕ_f, must become large and the efficiency of the field line following algorithm is reduced.

The computer time consumed by the field line following algorithm is typically dominated by the evaluations of the magnetic field. Let $\Delta k_{\min}$ denote the smallest distance to be resolved in Fourier space, i.e.,

$$\Delta k_{\min} = \min_{i,j} [(n_i - \ell m_i) - (n_j - \ell m_j)] \tag{20}$$

for the set of modes whose amplitudes are to be calculated. To reduce the spectral leakage below a given tolerance of ε, the width of the window function must be greater than

$$\phi_g = \sqrt{-4 \ln(\varepsilon)}/\Delta k_{\min}. \tag{21}$$

The second term in Eq. (18) requires that the integration length, ϕ_f, be large compared to ϕ_g,

$$\phi_f \approx \phi_g \sqrt{-\ln(\varepsilon)}. \tag{22}$$

In following a magnetic field line (i.e., integrating Eq. (4)) with a differential equation solver, the step size in ϕ must be less than the smallest spatial scale over which the magnetic field is varying, $\Delta \phi$. The number of evaluations of the magnetic field required by the equation solver is, therefore, determined by the highest frequency component of the magnetic field. This is the same as the highest frequency component we must retain in our Fourier decomposition,

$$\Delta \phi = \pi / \Delta k_{\max}$$
$$= \pi / (n - \ell m)_{\max}. \tag{23}$$

Let n_i denote the number of integration steps required to traverse a distance $\Delta \phi$. Typically, $n_i \approx 10$. Putting together Eqs. (21)–(23), and recalling that we must evaluate each of the three components of the magnetic field at each integration step, we find that the total number of magnetic field evaluations required to calculate the magnetic coordinates is

$$\frac{6n_i \ln \varepsilon}{\pi} \frac{\Delta k_{\max}}{\Delta k_{\min}} \approx 20 \ln \varepsilon \, \Delta k_{\max} / \Delta k_{\min}. \tag{24}$$

Near a low order rational surface, $\Delta k_{\min}$ becomes small, and the value of the expression in Eq. (24) becomes large. Field line following methods are inefficient there. The Fourier decomposition along the field line, Eq. (7), has modes with very long wavelength, requiring field line following over long distances for their accurate calculation. This is the Fourier space counterpart of the observation that many transits are required to sample adequately the flux surface (Fig. 1). It is clear from these considerations that the inefficiency is not limited to a particular scheme, but is common to all methods which construct action-angle variables from trajectory information. For this reason, we turn now to a consideration of iterative methods.

IV. PICARD ITERATION

Picard iteration offers a relatively simple procedure for determining the solution of Eq. (10) and Eq. (11). When the Picard iteration procedure works, it is found to be an efficient method for calculating action-angle variables. It is therefore of interest to know for what type of problem the procedure works. Numerical instabilities are found in certain cases. These can sometimes be cured by incorporating spectral information from field line following in the iteration procedure.

Given a nonlinear vector equation

$$\mathbf{z} - \mathbf{f}(\mathbf{z}) = 0, \tag{25}$$

the Picard iteration procedure [24] for determining a root of Eq. (25) takes the form

$$\mathbf{z}^{(l+1)} = \mathbf{f}(\mathbf{z}^{(l)}), \tag{26}$$

where l indicates the iteration number. For the Fourier space representation of Hamilton's equations the iteration scheme takes the form

$$
\begin{aligned}
\jmath^{(l+1)} &= (\mathscr{J} B^\theta)_{00}^{(l)}, \\
\mathbf{x}_{nm}^{(l+1)} &= (\mathscr{J} \mathbf{B})_{nm}^{(l)} / (n - \jmath^{(l+1)} m).
\end{aligned}
\tag{27}
$$

Clearly, Eq. (27) cannot be used to determine the $n = 0$, $m = 0$ Fourier components of $\mathbf{x}$. To calculate these modes we make the modification of adding a term $c_{00} \mathbf{x}_{00}$ to the left and right sides of Eq. (10). A sensible choice for c_{00} is suggested by Taylor expanding around the true solution and seeking optimal convergence for $\mathbf{x}_{00}$. This is achieved if c_{00} is equal to the gradient of $(\mathscr{J}\mathbf{B})_{00}$. To avoid calculating derivatives, however, we prefer to choose $c_{00} = \jmath$ which is motivated by the fact that this can be shown to be an optimal choice when the flux surfaces are circular. (See, for example, the Hamiltonian oscillator model with $\alpha_3 = \alpha_4 = 0$). The equation for $\mathbf{x}_{00}$ is therefore

$$\mathbf{x}_{00}^{(l+1)} = \mathbf{x}_{00}^{(l)} - (\mathscr{J}\mathbf{B})_{00} / \jmath^{(l+1)}. \tag{28}$$

This modification is equivalent to explicitly pulling the $\jmath_0^2 q^2$ harmonic oscillator piece out of the Hamiltonian [17].

For the calculations presented in this paper, a flux surface is specified by fixing its intersection with the midplane at $\theta = 0$, $\phi = 0$. To enforce this restriction the $n = 0$, $m = 1$ Fourier component of x is evaluated using

$$x_{01} = 1 - \sum_{nm} x_{nm}, \tag{29}$$

instead of using Eq. (27) (cf. Ref. [17], where a constraint of constant action was imposed to calculate the fundamental harmonics).

To study convergence of the Picard iteration procedure we consider an application to the anharmonic oscillator model Eq. (16) for different values of the nonlinear coupling parameters. In particular, results for three cases are presented:

  (i) $\iota_0 = 0.14$, $\alpha_3 = 0.0$, $\alpha_4 = 0.2$

  (ii) $\iota_0 = 0.14$, $\alpha_3 = 0.0$, $\alpha_4 = 20.0$

  (iii) $\iota_0 = 0.14$, $\alpha_3 = 0.0$, $\alpha_4 = -0.248$.

Flux surfaces for the three cases are shown in Fig. 3. As α_4 is increased from zero, the outer flux surfaces become elongated and squared off at the top and bottom. As α_4 is decreased from zero the outermost surface approaches a separatrix, which is achieved for $\alpha_4 = -0.25$. Since the poloidal flux function is axisymmetric (independent of ϕ), modes with different toroidal mode number are decoupled from one another. We restrict attention to $n = 0$, and set $x_{nm} = 0$ for all $n \neq 0$.

The iteration is initialized using the Fourier components and ι corresponding to circular flux surfaces. At each step of the iteration the difference is calculated between values of x_{nm} obtained from the Picard iteration and "exact" values obtained by Fourier transforming along the field lines. Figure 4 is a plot showing the convergence of the fundamental harmonic, y_{01}, as a function of iteration number. Since $(\iota m)x_{0m} = -B^x = \partial\psi_p/\partial y$ is linear in y, whereas $(\iota m)y_{0m} = -B^y = \partial\psi_p/\partial x$ is nonlinear in x for the model problem, the Picard iteration scheme produces substantial changes in a given Fourier component only on alternate iterations. This behavior is clearly visible in the convergence for $\alpha_4 = 0.2$ in Fig. 4.

When α_4 is small the rate of convergence is independent of α_4. The rate is approximately independent of iteration number for iterations after the second, with the error decreasing by a factor of about 0.1 per alternate iteration. The rate of convergence decreases as α_4 and the degree of surface distortion increases. For $\alpha_4 = 20.0$ the error decreases by a factor of about 0.8 per iteration. For this case the outer flux surface has an elongation of about 6.5. With $\alpha_4 = -0.248$, the outer flux surface is close to a separatrix surface, yet the error still decreases at a rate of about 0.7 per iteration.

As α_4 is increased, an increasing number of Fourier harmonics are needed to provide an accurate representation of a flux surface. The rate of convergence of the Picard iteration appears to be insensitive to the number of harmonics retained. For example, we have run $\alpha_4 = 1.0$ with six Fourier harmonics, and then with eleven harmonics. Also, we have run $\alpha_4 = 5.0$ with 12 harmonics and with 19. We find in both cases that the convergence is almost identical until the magnitude of the residual becomes comparable with the magnitude of the Fourier truncation error.

For $\alpha_3 = 0$ the symmetry of the flux function dictates that x_{00} is fixed throughout the iterations. To test the iteration scheme for x_{00} requires setting α_3 to a non-zero value. When this is done, the scheme using Eq. (28) typically converges for the first few iterations, but ultimately diverges. To cure this problem, information from field line following has been used to develop an alternative iteration equation for x_{00}.
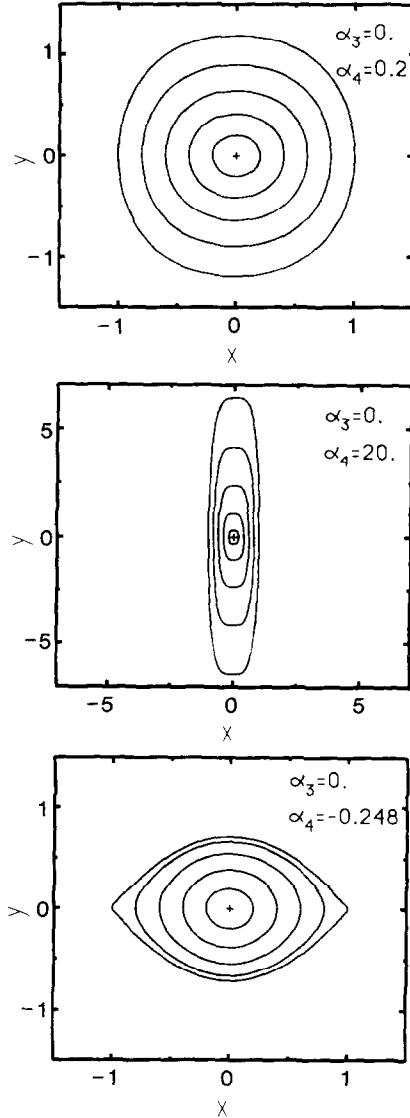
FIG. 3. Flux surfaces for the anharmonic oscillator with three different values of the nonlinear coupling parameters.

Equation (18) gives a set of linear relations between the numerically calculated Fourier transforms along the field lines and the Fourier components of x,

$$\tilde{x}_{nm} = \sum_{n'm'} x_{n'm'} \exp[-(n - \jmath m - n' + \jmath m')^2 \, \phi_g^2/4], \tag{30}$$

where $\tilde{x}_{nm}$ is the amplitude of the numerically evaluated one-dimensional Fourier transform along a sufficiently long field line at $k = n - \jmath m$. Since the left-hand side
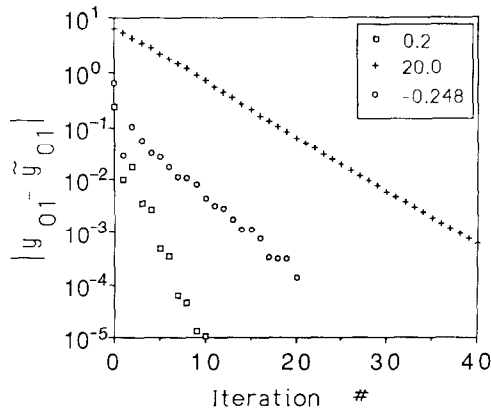
FIG. 4. Convergence of Picard iteration of the anharmonic oscillator case with $\alpha_3 = 0$ and the values of α_4 shown in the figure. The vertical axis corresponds to the difference between the value of the $n = 0$, $m = 1$ component of y from the direct Fourier transform calculation and that from the iterative calculation.

of Eq. (30) is a known quantity, this equation defines a set of linear relations between the x_{nm}'s which can be used to eliminate one or more of the nonlinear equations (27) and (28). In particular, we cure the Picard iteration scheme for calculating x_{00} by substituting the $n = 0$, $m = 0$ component of Eq. (30) for Eq. (28). The modified Picard scheme is found to converge rapidly, and the convergence properties for finite α_3 are similar to those for $\alpha_3 = 0$.

We find that the stability of Picard iteration is unreliable in three dimensions. In particular, it does not converge for the stellarator fields described in Appendix A. Even for the two-dimensional anharmonic oscillator, $n \neq 0$ Fourier components, x_{nm}, grow up from noise when they are not set equal to zero at each stage of the iteration.

The question remains whether Picard iteration can be stabilized by incorporating a sufficient number of constraints from field line following of the form of Eq. (30). A straightforward stability analysis of Picard iteration for α_3 and α_4 small indicates that near-resonant modes, $|n - jm| < |j|$, pose the greatest difficulty for stability. Since these are also precisely the modes whose amplitude we are most interested in calculating iteratively, this suggests that Picard iteration is not sufficiently stable for our purposes. We therefore turn to Newton's method.

V. NEWTON'S METHOD

Given the nonlinear vector equation, Eq. (25), the Newton iteration scheme takes the form [24]

$$z^{(l+1)} = z^{(l)} - J^{-1} f,$$

where J is the Jacobian matrix of first derivatives

$$J_{ij} = \partial f_i / \partial z_j.$$

To apply Newton's method to Eq. (10), we need to evaluate gradients of the form $\partial(\mathscr{J}\mathbf{B})_{n_1 m_1}/\partial \mathbf{x}_{n_2 m_2}$. In Appendix B we show how these gradients can be conveniently expressed in terms of the Fourier components of $\nabla(\mathscr{J}\mathbf{B})$.

To examine the convergence of the Newton iterations, we monitor the residual

$$\varepsilon_{\max} = \max(|\mathbf{x}_{nm} - (\mathscr{J}\mathbf{B})_{nm}/(n - \jmath m)|, |\jmath - (\mathscr{J}B^0)_{00}|). \tag{31}$$

When this number is less than some preset tolerance, typically of the order of 10^{-5}, we consider the iteration procedure to be converged. We verify that the Newton iteration scheme gives the same results as field line following, to within the preset tolerances and the expected truncation errors.

We first study the convergence properties for the simple anharmonic oscillator Hamiltonian with $\alpha_3 = 0$ and small α_4. To determine the dependence on the number of poloidal modes, we have run the $\alpha_4 = 0.2$ case with 6 and with 41 poloidal harmonics, and with the single toroidal harmonic $n = 0$. For the initial guess, circular flux surfaces with the appropriate $\jmath$ independent of r were assumed. For both runs, Newton's method was found to converge to a maximum residual less than 10^{-5} after three iterations. Despite the simplicity of the magnetic field model used in the calculation, the time required for the Newton iterations was found to be dominated in each case by the magnetic field evaluations rather than the Jacobian matrix inversion.

The number of magnetic field evaluations required by the field line following method depends only weakly on the number of poloidal harmonics, and varies from 890 for 6 harmonics to 929 for 41 harmonics. In contrast, the number of evaluations required to Fourier decompose the field for the iterative schemes is approximately proportional to the number of harmonics. This number is 520 for the Newton iteration with 6 modes, and 3320 for 41 modes. In making these comparisons, it should be kept in mind that in practice the number of modes retained will be determined by the need to resolve adequately the structure of the flux surfaces. Any fine-scale structure that requires us to retain a large number of modes would also require a large number of evaluations in following the field lines.

When additional toroidal harmonics are included, resonances may be introduced. To study the effect of resonances, we have run a sequence of cases with $\alpha_4 = 0.2$ with three toroidal harmonics ($n = -1, 0, 1$) and three poloidal modes ($m = 0, 1, 2$). For the chosen value of α_4 the $\jmath$ profile varies little across the plasma, increasing by about 10% from the magnetic axis to the edge. The value of $\jmath$ at the magnetic axis is specified by the input parameter $\jmath_0$. We run with a range of values of $\jmath_0$ to investigate the effect of $\jmath$ approaching $\frac{1}{4}$ (an $n = 1$, $m = 4$ resonance) on the outer flux surface. Figure 5 is a plot of the number of magnetic field evaluations needed for both the field line following method and Newton's method as a function of $\jmath$,
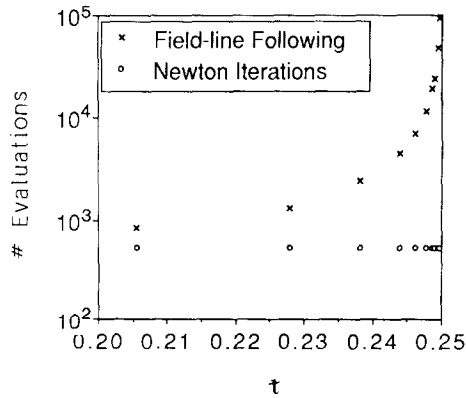
FIG. 5.   A series of runs for the anharmonic oscillator with $\alpha_3 = 0$ and $\alpha_4 = 0.2$, using both trajectory following and Newton iterative algorithm. The x-coordinate is ℓ, the rotational transform on the outer flux surface; ℓ is varied by choosing different values for ℓ_0. The y-coordinate is the number of evaluations of the magnetic field required to calculate the magnetic coordinates on the outer surface.

the transform on the outer flux surface. The resonance does not affect Newton's method at all for this range of winding numbers. This is not an obvious result, since a near resonant $n - \ell m$ denominator appears in the function whose zero we are seeking, $\mathbf{x}_{nm} - (\oint \mathbf{B})_{nm}/(n - \ell m)$. The number of evaluations required by the field line following method goes like $1/(n - \ell m)$. Although the $n = 1$ harmonic is not, of course, required for the calculation in this case, the addition of a small $n = 1$ perturbation to the magnetic field would force us to retain that mode when calculating action-angle coordinates, with results similar to those shown in Fig. 5.

When α_4 is increased, difficulties appear for a pure Newton's method. For $\alpha_4 = 0.5$, the number of Newton iterations required with 41 modes increases to 5, a 60 % increase, while the number of evaluations required for the field line following calculation increases by 32 %. For $\alpha_4 = 1.0$, the pure Newton's method no longer converges with the surfaces initially circular and the corresponding initial guess for ℓ.

To study the effects of the initial guess on the stability of Newton's method, we next improve our initial guess by using accurate values (to within a tolerance of 10^{-5}) for ℓ and y_{01} calculated by field line following. The case with $\alpha_4 = 1.0$ now converges in three iterations, as does the case with $\alpha_4 = 2.0$. The algorithm remains divergent for $\alpha_4 = 5.0$. It converges in two iterations for $\alpha_4 = -0.15$, and in four iterations for $\alpha_4 = -0.22$, $\alpha_4 = -0.24$, and $\alpha_4 = -0.245$. The algorithm diverges for $\alpha_4 = -0.248$. In Ref. [11], an improvement in convergence properties was achieved by first iterating with a small number of modes, and then using the results to initialize a calculation with a larger number of modes. For the anharmonic oscillator case with $\alpha_4 = -0.248$, we find Newton's method to be divergent for both 19 and 4 modes, even with both ℓ and y_{10} initialized to accurate values, and with the x_{00} and x_{10} equations replaced by linear relations.

The situation changes when we improve the initial guess for the remaining Fourier harmonics. We follow the field lines for a moderate distance to determine the Fourier coefficients to a relatively low accuracy, and we use these values to specify the initial guess for Newton's method. The Newton iterations now converge for both $\alpha_4 = 20.0$ and $\alpha_4 = -0.248$. A plot of the convergence as a function of the iteration number is shown in Fig. 6 for three different values of α_4.

For $\alpha_4 = 20.0$, we initialize by following the field lines nine times around the torus, giving an initial residual of about 0.5. After three Newton iterations, the residual goes down to 6.5×10^{-5}. The pure field line following approach requires us to follow the field lines 38 times around the torus to achieve this accuracy. The field line following method is near the limit of its accuracy for this value of the residual. The roundoff errors in following the field lines prevent us from decreasing the residual further by this method. The pure field line following approach requires 11,043 field line evaluations, compared with a total of 8656 evaluations required by the hybrid approach. We retain 41 poloidal modes in these calculations.

In all of the calculations described thus far, we have substituted Eq. (11) for the $n = 0$, $m = 1$ component of Eq. (10). We have rerun the cases shown in Fig. 6, retaining the $n = 0$, $m = 1$ equation rather than Eq. (11). The results are shown in Fig. 7. We find that the convergence is somewhat more rapid with this scheme. We therefore retain this scheme for the following calculations with stellarator fields.

The numerical experiments with the anharmonic oscillator Hamiltonian lead us to conclude that Newton's method alone is insufficiently stable to be a generally useful algorithm for finding action-angle variables. A moderately accurate initial guess is required to provide stability. A hybrid scheme, in which field line following is used to provide this moderately accurate guess appears to work well. In the absence of resonances, the efficiency of the hybrid scheme is comparable to that of pure field line following. When resonances appear, pure field line following becomes
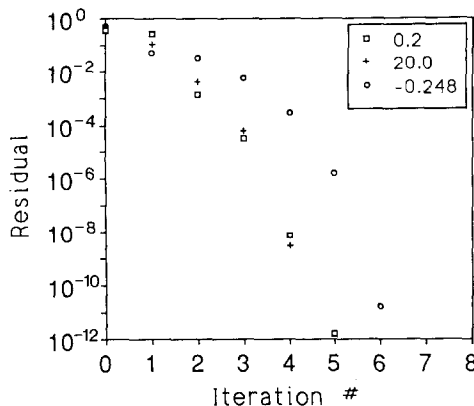

FIG. 6. Convergence of Newton's method initialized by following field lines a moderate distance to determine the initial guess.
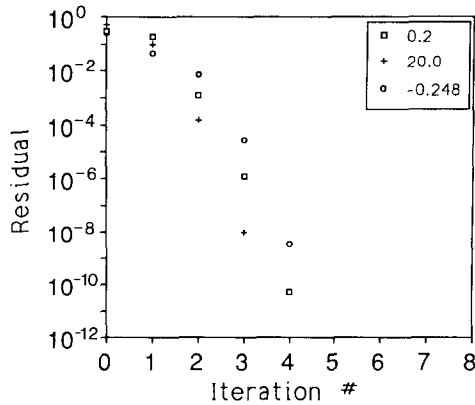
FIG. 7.   Convergence porperties when the $n = 0$, $m = 1$ component of Eq. (10) is retained rather than Eq. (11).

slow, and we would expect the hybrid scheme to give much superior efficiency. We test this expectation for a case of practical interest, a three-dimensional stellarator vacuum field.

The stellarator example we consider is the same as that used in Ref. [4]. The vacuum field solution is described in Appendix A. We choose parameters corresponding to the ATF stellarator at Oak Ridge. The winding number ranges from 0.308 at the magnetic axis to 0.975 at the edge. There are $N = 12$ toroidal field periods.

For the anharmonic oscillator Hamiltonian, we have an analytic expression for the x and y Cartesian coordinates components of the magnetic field. This allowed a direct evaluation of the right-hand side of Eq. (10) in Cartesian coordinates. The general situation is somewhat more complicated. The magnetic field is often specified in a non-Cartesian coordinate system. For the numerically determined stellarator vacuum field, we work in a coordinate system in which the outer coordinate surface coincides with the plasma boundary. For our applications to the solution of three-dimensional equilibria [25, 26], we need to work in a general curvilinear coordinate system. The modifications to the algorithm necessary to handle this are discussed in Appendix C.

To determine the effect of resonances for the stellarator model, three cases were run, one in which we retain all modes with $-3 \leqslant n \leqslant 3$, $0 \leqslant m \leqslant 6$, one in which we retain $-3 \leqslant n \leqslant 3$, $0 \leqslant m \leqslant 10$, and another in which we retain poloidal mode numbers up to $m = 13$. In the first case no resonance condition ($\Delta k_{\min} = 0$ in Eq. (20)) is satisfied among the modes, since $\min_{ij}(N(n_i - n_j)/(m_i - m_j)) \geqslant 1.0$ and $\iota \leqslant 0.975$ in the plasma region. Resonance effects begin to appear when modes are retained having $m > 6$. Field line following methods have difficulty distinguishing the $n = 0$, $m = 7$ mode from the $n = 1$, $m = 6$ mode near the $\iota = \frac{12}{13}$ rational surface. Similarly, the $n = 1$, $m = 7$ mode is difficult to distinguish from the $n = 0$, $m = 6$ mode. The identification of a pair of modes at the $\iota = \frac{6}{7}$ surface is also problematical. The number of modes which are difficult to distinguish increases as we improve our angular

resolution, with the problem localized at the low order rational surfaces. When the $m = 13$ modes are included, the $n = 1$, $m = 13$ mode is itself resonant near the $\iota = \frac{12}{13}$ rational surface.

We calculate magnetic coordinates on 20 flux surfaces, ranging from the magnetic axis to the edge of the plasma. Every second one of these flux surfaces is shown in Fig. 8. The difficulties posed by the low order rational surfaces are compounded by

field line the same number of times around the torus. The number is determined by the stringent requirements posed near the low order rational surfaces. On the other hand, if we do not follow each field line the same distance, the gains are more than offset by the loss of vectorization.

To apply the field line following method when we retain poloidal mode numbers up to $m = 6$, we find that field lines must be followed approximately 13.4 times around the torus, consuming about 70 s of cpu time on a Cray 2. When we retain poloidal modes up to $m = 13$, we require 100 circuits of the torus to do the calculation purely by the line following method, and we consume 840 s of cpu time. Applying our hybrid approach to the same case, we first follow the field lines about 14 times around the torus to determine the initial guess for Newton iteration, and then we iterate. The initial guess has a maximum residual of 5.5×10^{-3}. One Newton iteration brings this down to 1.4×10^{-5}, as compared with a residual of 1.5×10^{-5} obtained from the pure line following method. The total time consumed by the hybrid approach is 245 s, of which about 145 are consumed in the field line following part of the calculation.
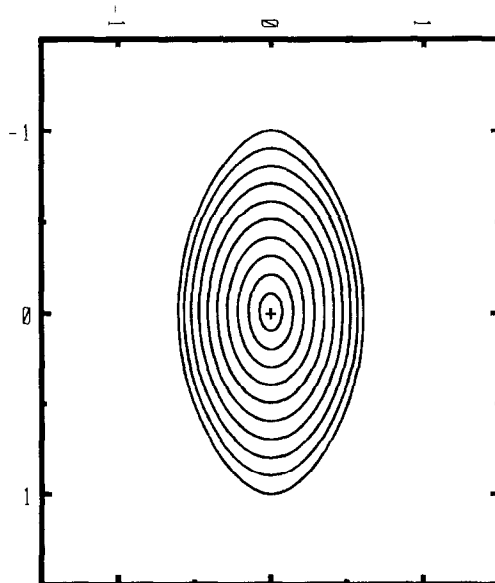

FIG. 8.   Flux surfaces of the ATF stellarator model.

VI. Efficiency of Iterative Methods

In this section we use information obtained from the numerical experiments described in Sections IV and V to derive some general estimates of the efficiency of iterative methods. The efficiency of the iterative methods is comparable to that of trajectory-following methods for nonresonant calculations, but they do not see the same degradation in efficiency near low order rational surfaces.

An iterative solution of Eq. (10) requires a Fourier decomposition of the right-hand side at each iteration. This requires that the right-hand side be evaluated on an $(m+1) \times (2n+1)$ (θ, ϕ) grid on each surface. The three components of the magnetic field must be evaluated at each grid point, so there are $3(m+1)(2n+1)$ evaluations required at each iteration.

For a linearly convergent scheme, such as Picard iteration, the error decreases by a fixed ratio at each iteration. Let us call this ratio A, the error amplification factor. For the cases discussed in Section IV, the value of A ranges from about 0.3 to about 0.8. To achieve a required tolerance of ε requires n_s steps, where

$$n_s = (\ln \varepsilon - \ln \varepsilon_0)/\ln A,$$

$$\approx \ln \varepsilon/\ln A,$$

and where ε_0 is the initial error, which is typically of order one. The total number of evaluations required is

$$3(m+1)(2n+1) \ln \varepsilon/\ln A. \tag{32}$$

Division of Eq. (32) by Eq. (24) determines the relative number of evaluations required by Picard iteration and field line following. The ratio is

$$\frac{\pi(m+1)(2n+1)}{2n_i \ln A} \frac{\Delta k_{\min}}{\Delta k_{\max}}.$$

For $n = 0$, the ratio is

$$\frac{\pi}{2n_i \ln A} \frac{m+1}{m}.$$

In the absence of resonances, the two methods are comparable, with Picard iteration being somewhat more efficient. The advantage of Picard iteration would be great for the finite n near-resonant cases, but the method is generally unstable for these cases.

Newton's method requires the evaluation of the derivatives of the magnetic field components at each iteration. Assuming that radial derivatives are evaluated by finite differencing, we require a total of $9(m+1)(2n+1)$ evaluations at each iteration. The Jacobian matrix must also be inverted at each iteration. The time to invert a matrix goes up as the cube of the matrix dimension (the total number of

modes), while the number of required magnetic field evaluations scales linearly with the total number of modes. The time for Newton's method will, in principle, be dominated by the matrix inversions rather than by the magnetic field evaluations when there is a sufficiently large number of modes retained.

The time required for our matrix inversions on the Cray 2 is $1.5 \times 10^{-8} N^3$ s. for large N, where N is the number of modes retained in the calculation. For the numerically specified magnetic fields (for example, for the stellarator calculations), each evaluation of a magnetic field component requires us to interpolate radially a set of Fourier components between grid surfaces, and then sum the interpolated Fourier amplitudes. The time required by a magnetic field evaluation therefore scales like N. For the stellarator calculations, the cpu time on the Cray 2 was $5.3 \times 10^{-6} N$ s. The $9N$ evaluations required at each step of the Newton iteration procedure consume $4.8 \times 10^{-5} N^2$ cpu s. The crossover point at which matrix inversions begin to dominate the cpu time is $N = 3200$ modes. This is a very large number of modes. In particular, for all the runs described in Section V, matrix inversions consumed a very small fraction of the cpu time.

Newton's method is quadratically convergent. The error decreases at a rate given by $\varepsilon_{j+1} = A\varepsilon_j^2$ for some A, where ε_j denotes the error at the jth iteration. The required precision ε is reached after n_N iterations, where

$$\ln \varepsilon = \ln A^n + \ln \varepsilon_0^{2^{n_N}} \approx \ln \varepsilon_0^{2^{n_N}}.$$

This gives

$$n_N \approx \ln \ln \varepsilon / \ln 2.$$

For example, to achieve a precision of $\varepsilon = 10^{-5}$, about four iterations should be required. This is consistent with the observations of Section V. The total number of magnetic field evaluations required is therefore approximately

$$36(m+1)(2n+1).$$

This is comparable to the number of evaluations required by Picard iteration, and is also comparable to the number of evaluations required by trajectory following methods for nonresonant cases. For near-resonant cases, iterative methods are more efficient.

VII. DISCUSSION

We have found that both trajectory-following and iterative methods for computing action-angle variables have important drawbacks, and that the drawbacks of each can be ameliorated by hybrid schemes.

Trajectory-following methods for calculating action-angle variables see a significant deterioration of their efficiency when the winding number is near a low order

rational. In the neighborhood of such a rational, the required integration length scales as $1/(n - \ell m)$, where ℓ is the local winding number and the mode numbers are m and n. The resulting inefficiency is compounded when vectorization of the calculation is desirable. For many applications the action-angle variables are desired on a set of surfaces. It is then desirable to vectorize the trajectory following. This requires that we follow each field line the same number of times around the torus, a number determined by the stringent requirements posed near the low order rational surfaces. On the other hand, if we do not follow each field line the same distance, the gains are more than offset by the loss of vectorization.

Trajectory-following methods also see an absolute limit on the attainable accuracy due to the propagation of roundoff errors. The local numerical errors are amplified linearly as a function of the distance the trajectory is followed (assuming that the trajectory is integrable). The amplification factor can get large near a separatrix or near the stochasticty threshold. The local accuracy needed to give a specified global accuracy can be quite stringent. There is a limit on the local accuracy due to roundoff. This places a limit on the achievable global accuracy.

Iterative methods do not see the same deterioration in efficiency near low order rational surfaces. Nor do they suffer from the limitations on accuracy due to propagation of roundoff errors. They do, however, display numerical instabilities. Trajectory-following methods, in contrast, are robust.

We have studied the use of both Picard and Newton iteration methods for the calculation of action-angle variables, both alone and in combination with field line following methods.

Instability is a serious problem for Picard iteration. Its stability can be improved somewhat by using information from trajectory following to obtain modified iteration schemes. Nevertheless, even in this hybrid form, Picard iteration is not sufficiently stable to provide a generally useful procedure for calculating action-angle variables.

Instabilities are also a major liability for a pure Newton's method, in which a rough initial guess is obtained from analytical estimates. Newton's method is extremely sensitive to the initial guess. However, given a sufficiently accurate initial guess, we have always found it to be stable. This suggests a hybrid scheme in which the initial guess for the action-angle variables is obtained by trajectory-following methods, with the solution then refined by Newton iteration.

For the stellarator cases that we studied, which are of practical interest for plasma physics applications, we found that there were a small number of relatively low amplitude Fourier modes whose accurate calculation by trajectory-following methods would require long integration lengths. The remaining modes could be efficiently calculated by trajectory-following methods. A single pass of the Newton iteration procedure then sufficed to complete the accurate calculation of the magnetic coordinates.

Further improvements in the hybrid scheme are also possible, with the nonlinear equations for a large subset of the modes replaced by the corresponding linear equations obtained from trajectory following, Eq. (30). Because the linear equations

are solved exactly with each inversion of the Jacobian matrix, we would expect that the convergence of the Newton method could be speeded up in this way. Improved equations for a subset of the modes could also relax the accuracy requirement for the initial guess of the remaining mode amplitudes. We have implemented such a replacement only for the $m = 0$, $n = 0$ Fourier component of our equation.

These observations can be reformulated as a set of recommendations on when to use the various computational methods. If the winding number is not near a low order rational (i.e., if $n - \omega m$ is not much less than one for the mode numbers retained in the calculation), then trajectory following is comparable in efficiency to iterative methods. The robustness of trajectory following then recommends it as the method of choice, as long as the required accuracy is not so small that roundoff errors are a problem. Roundoff errors become an issue for trajectory following methods even at moderate accuracy if a trajectory approaches the transition to chaos.

Considerations of efficiency make Newton's method increasingly desirable as the winding number approaches a low order rational. (Picard iteration appears to be generally unstable near low order rationals, and is therefore not a viable alternative.) For those special cases where an accurate initial guess is at hand, Newton's method alone may be adequate. In general, a hybrid scheme is desirable. If roundoff errors are an issue, Newton's method is again called for, either alone or in combination with trajectory following.

In conclusion, we have found that a hybrid approach to the computation of action-angle variables, using both trajectory following methods and Newton's method, can combine the best features of both methods. Given a moderately accurate initial guess obtained from line following, Newton's method has been found to be robustly stable. The refinement of the solution by Newton's method eliminates the need to follow trajectories over inordinate distances when the winding number is near a low order rational.

APPENDIX A: Test Cases

In Sections IV and V we describe a series of numerical experiments in which we test iterative methods for the calculation of action-angle variables, both alone and in combination with trajectory following methods. The relatively simple anharmonic oscillator test case is described in Section II. In this appendix we give a detailed description of our stellarator test case.

For a case which is of practical interest in plasma physics, we construct an $l = 2$ stellarator vacuum field in a torus, with parameters corresponding to the ATF device at Oak Ridge [27]. To obtain the vacuum field we solve Laplace's equation for the scalar potential of the magnetic field,

$$\mathbf{B} = \nabla \chi, \tag{A1}$$

with the boundary condition $\mathbf{n} \cdot \nabla \chi = 0$, where $\mathbf{n}$ is the unit normal to the outer flux

surface. Laplace's equation is solved using a Fourier representation in the toroidal and poloidal angles, and second-order finite differences in the radial direction, with the resulting block tridiagonal matrix inverted by Gaussian elimination [26]. For the examples discussed in Section V we used 20 radial grid surfaces. The outer flux surface is specified by the Bessel function solution for the cylindrical vacuum field,

$$\chi = B_0\phi + b_2 I_2(Nr/R)\sin(2\theta - 2N\phi), \tag{A2}$$

where $N = 12$ is the number of periods, and $R = 7$ is the major radius. The ratio b_2/B_0 determines the winding number. We choose a value which is appropriate for ATF, $2Nb_2/B_0 = 0.633$, giving a winding number which ranges from 0.308 at the axis to 0.975 at the edge. To evaluate the stellarator magnetic field at a given point, we interpolate its Fourier coefficients radially using splines, and then we sum the interpolated Fourier components.

APPENDIX B: EVALUATION OF THE JACOBIAN MATRIX

To apply Newton's method to Eq. (10), we need to evaluate the gradients of the Fourier coefficients of $\mathscr{J}\mathbf{B}$ with respect to the Fourier coefficients of $\mathbf{x}$, $\partial(\mathscr{J}\mathbf{B})_{n_1m_1}/\partial\mathbf{x}_{n_2m_2}$. We could directly calculate each of these derivatives numerically by independently varying each $\mathbf{x}_{nm}$ and determining the corresponding variation in each of the Fourier components of $\mathscr{J}\mathbf{B}$. This would require $6(m+1)(2n+1)$ magnetic field evaluations for each Fourier component of $\mathbf{x}$, for a total of $12(m+1)^2(2n+1)^2$ evaluations. If that many evaluations were necessary, it would make Newton's method uninteresting. Fortunately, we can express $\partial(\mathscr{J}\mathbf{B})_{n_1m_1}/\partial\mathbf{x}_{n_2m_2}$ in terms of the Fourier components of $\nabla(\mathscr{J}\mathbf{B})$. The calculation of $\nabla(\mathscr{J}\mathbf{B})$ requires three evaluations at each point of an $(m+1) \times (2n+1)$ grid.

In this appendix we show how to express $\partial(\mathscr{J}\mathbf{B})_{n_1m_1}/\partial\mathbf{x}_{n_2m_2}$ in terms of the Fourier components of $\nabla(\mathscr{J}\mathbf{B})$. For simplicity, we include details only for $\partial(\mathscr{J}B^x)_{n_1m_1}/\partial y_{n_2m_2}$. The derivation is similar for the other vector components.

From

$$y \equiv \sum_{nm} y_{nm}\sin(n\phi - m\theta), \tag{B1}$$

and

$$\partial(\mathscr{J}B^x)/\partial y \equiv \sum_{nm} b_{nm}^{xy}\cos(n\phi - m\theta), \tag{B2}$$

we obtain

$$\partial(\mathscr{J}B^x)/\partial y_{nm} = (\partial(\mathscr{J}B^x)/\partial y)(\partial y/\partial y_{nm})$$

$$= \sum_{n_1m_1} b_{n_1m_1}^{xy}\cos(n_1\phi - m_1\theta)\sin(n\phi - m\theta).$$

This gives

$$\partial(\mathscr{J}B^x)/\partial y_{nm} = \sum_{n_2 m_2} D^{xy}_{n_2 m_2} \sin(n_2\phi - m_2\theta), \tag{B3}$$

where

$$D^{xy}_{n_2 m_2} = (1/2)(b^{xy}_{n_2 - n, m_2 - m} - b^{xy}_{n_2 + n, m_2 + m}). \tag{B4}$$

Similarly, we find

$$D^{xx} = (1/2)(b^{xx}_{n_2 - n, m_2 - m} + b^{xx}_{n_2 + n, m_2 + m});$$

$$D^{yx} = (1/2)(b^{yx}_{n_2 - n, m_2 - m} + b^{yx}_{n_2 + n, m_2 + m});$$

$$D^{yy} = -(1/2)(b^{yy}_{n_2 - n, m_2 - m} - b^{yy}_{n_2 + n, m_2 + m}).$$

APPENDIX C: COMPUTATION OF MAGNETIC COORDINATES FOR A MAGNETIC FIELD GIVEN IN CURVILINEAR COORDINATES

In describing algorithms for the computation of magnetic coordinates at the beginning of Sections IV and V, we assumed that the magnetic field is specified in Cartesian coordinates. In that case, Eq. (10) can be used to solve iteratively for $\mathbf{x}(\psi, \theta, \phi)$, the transformation from magnetic coordinates to Cartesian coordinates. More generally, the magnetic field is often given in a curvilinear coordinate system. In this appendix we describe the modifications required to compute magnetic coordinates in that case.

Suppose that the magnetic field is given in a coordinate system (ρ, θ, ϕ), and that $\mathbf{x}(\rho, \theta, \phi)$ is also given. For simplicity, we assume that ϕ is the uniform geometric toroidal angle used throughout this paper. The iterative solution of Eq. (10) for $\mathbf{x}(\psi, \theta, \phi)$ would require an evaluation of the magnetic field $\mathbf{B}$ on a (ψ, θ, ϕ) grid, so that we could Fourier decompose $\mathbf{B}$ in θ and ϕ for a fixed ψ. Since $\mathbf{x}$ is a given function of (ψ, θ, ϕ) at each step, this could be accomplished by an inversion of $\mathbf{x}(\rho, \theta, \phi)$. Inversion of this function at each iteration step would dominate the time required by the iterative algorithm, and would greatly decrease the attractiveness of such algorithms. Alternatively, one could construct a scheme in which this function is inverted only once at the beginning, and the inverse stored in some form for application at each iteration of the algorithm. The inversion would have to be performed over a sufficiently large domain so that the subsequent evaluations of $\mathbf{x}(\psi, \theta, \phi)$ all lie within its range. This approach would be cumbersome.

We formulate an alternative iterative scheme by defining "pseudo-Cartesian coordinates,"

$$\xi \equiv \rho \cos(\theta), \qquad \eta \equiv \rho \sin(\theta).$$

The chain rule for differentiation gives

$$\mathbf{B} \cdot \nabla\xi = (\partial\xi/\partial\theta)\mathbf{B} \cdot \nabla\theta + (\partial\xi/\partial\phi)\mathbf{B} \cdot \nabla\phi,$$

or

$$(\partial \xi/\partial \phi) + \imath(\partial \xi/\partial \theta) = B^{\xi}/B^{\phi}.$$

Fourier decomposition in θ and ϕ gives an equation analogous to Eq. (10). A similar equation is obtained for η. These equations can be solved iteratively for $\xi(\psi, \theta, \phi)$ and $\eta(\psi, \theta, \phi)$ in precisely the same way that Eq. (10) is solved for $\mathbf{x}(\psi, \theta, \phi)$. The solution is substituted into $\mathbf{x}(\rho, \theta, \phi)$ to determine $\mathbf{x}(\psi, \theta, \phi)$.

ACKNOWLEDGMENT

REFERENCES

1. A. H. Boozer, *Phys. Fluids* **25**, 520 (1982).
2. G. Kuo-Petravic, A. H. Boozer, J. A. Rome, and R. H. Fowler, *J. Comput. Phys.* **51**, 261 (1983).
3. H. L. Berk, M. N. Rosenbluth, and J. L. Shohet, *Phys. Fluids* **26**, 2616 (1983).
4. A. H. Reiman and H. S. Greenside, *J. Comput. Phys.* **75**, 423 (1988).
5. D. W. Noid, M. L. Koszyowski, and R. A. Marcus, *Ann. Rev. Phys. Chem.* **32**, 267 (1981).
6. W. H. Miller, *Science* **233**, 171 (1986).
7. C. C. Martens and G. S. Ezra, *J. Chem. Phys.* **86**, 279 (1987).
8. J. M. Bowman (Ed.), *Thematic Issue on Molecular Vibrations*, Comput. Phys. Commun., Vol. 51 (North-Holland, Amsterdam, 1988).
9. J. Binney and D. Spergel, *Mon. Not. Roy. Astronom. Soc.* **206**, 159 (1984).
10. S. J. Ratcliff, K. M. Chang, and M. Schwarzschild, *Astrophysical Journal* **279**, 610 (1983).
11. R. L. Warnock and R. D. Ruth, *Phys. D* **26**, 1 (1987).
12. I. C. Percival, "Variational Principles for Invariant Tori," in *International Conference on Nonlinear Dynamics*, Vol. 357, Annals of the New York Academy of Sciences (New York Academy of Sciences, New York, 1979).
13. S. J. Shenker and L. P. Kadanoff, *J. Statist. Phys.* **27**, 631 (1982).
14. R. T. Skodje, F. Borondo, and W. P. Reinhardt, *J. Chem. Phys.* **82**, 4611 (1985).
15. C. W. Eaker, G. C. Schatz, N. D. Leon, and E. J. Heller, *J. Chem. Phys.* **81**, 5913 (1984).
16. C. C. Martens and G. S. Ezra, *J. Chem. Phys.* **83**, 2990 (1985).
17. I. C. Percival and N. Pomphrey, *Mol. Phys.* **31**, 97 (1976).
18. I. C. Percival and N. Pomphrey, *J. Phys. B* **9**, 3131 (1976).
19. S. Chapman, B. C. Garrett, and W. H. Miller, *J. Chem. Phys.* **64**, 502 (1976).
20. S. Hamada, *Nucl. Fusion* **1**, 23 (1962).
21. J. M. Greene and J. L. Johnson, *Phys. Fluids* **5**, 510 (1962).
22. E. O. Brigham, *The Fast Fourier Transform* (Prentice–Hall, Englewood Cliffs, NJ, 1974).
23. F. J. Harris, *Proc. IEEE* **66**, 51 (1978).
24. J. M. Ortega and W. C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables* (Academic Press, New York, 1970).
25. A. Reiman and H. Greenside, *Comput. Phys. Commun.* **43**, 157 (1986).
26. H. Greenside, A. Reiman, and A. Salas, *J. Comput. Phys.* **81**, 102 (1989).
27. J. F. Lyon *et al.*, *Fusion Technol.* **10**, 179 (1986).