August 1987 LRP 329/87

FBT - A FREE BOUNDARY TOKAMAK EQUILIBRIUM CODE FOR HIGHLY ELONGATED AND SHAPED PLASMAS

F. Hofmann

(to be published in Computer Physics Communications)

FBT - A FREE BOUNDARY TOKAMAK EQUILIBRIUM CODE FOR HIGHLY ELONGATED AND SHAPED PLASMAS

F. Hofmann

Centre de Recherches en Physique des Plasmas
Association Euratom - Confédération Suisse
Ecole Polytechnique Fédérale de Lausanne
21, Av. des Bains, CH-1007 Lausanne, Switzerland

ABSTRACT

A new axisymmetric, free-boundary equilibrium code, FBT, has been developed and tested. The code allows the computation of arbitrarily shaped tokamak equilibria, with external or internal separatrices and multiple magnetic axes. It is shown that bifurcations can be controlled by an iterative procedure using shape feedback. Three specific applications are discussed: (1) The problem of shape accuracy, (2) the calculation of a startup evolution for a highly elongated tokamak, and (3) methods to control saddle points.

1. INTRODUCTION

Plasma equilibrium calculations have always been an important tool for the design of magnetic confinement devices and for the analysis of experimental data. Recently, however, the scope of these calculations is being considerably expanded as the devices are becoming more and more complex. In highly elongated tokamaks, for example, the shape of the plasma boundary must be accurately controlled, in order to maintain axisymmetric stability throughout the discharge. It is therefore important to know how accurately a certain plasma shape can be produced with a given set of poloidal field coils. In addition, one would like to know how the coil currents must be changed in order to produce a predetermined shape modification. In divertor tokamaks, the question arises how a saddle point can be maintained at a given point in space while the plasma parameters change. Finally, one is often confronted with the conflicting requirements of shape accuracy on the one hand and minimum power dissipation in the poloidal field coils on the other hand, and one is looking for an optimum compromise.

Most of these questions can be answered by free-boundary equilibrium calculations, and several numerical codes [1-16] have been developed for this purpose. These codes can be divided into four classes, i.e., (a) 2D Eulerian codes [1-7], (b) 2D inverse codes and variational moment methods [8-9], (c) 2D evolutionary codes, including 1D transport [10-14] and (d) 3D codes [15-16]. In the first group, we find three codes [1,2,5] which allow the calculation of coil currents for a predetermined plasma shape. On the other hand, the codes described in Ref. [3,4,6,11] assume fixed coil currents. This implies that a trial-and-error method must be used if one wants to compute

equilibria with a given shape. Reference 7 describes a method for computing coil currents, assuming the coils lie on a rectangular surface where the flux function is a constant. Inverse codes and moments methods [8,9,12,15] generally assume nested flux surfaces and a single magnetic axis, hence they cannot treat cases with arbitrary magnetic topology such as internal separatrices. In Refs. [5,10] it is assumed that the plasma shape can be varied by specifying flux values at the positions of the shaping coils. However, this can lead to bifurcations, i.e. several different shapes may be obtained for identical boundary conditions [5].

In this paper, we describe a new code, FBT, which has been developed with the aim of computing highly elongated and arbitrarily shaped equilibria, having external or internal separatrices. The code is especially suited to compute poloidal field coil currents during the startup phase of shaped tokamaks, when the plasma cross section undergoes drastic changes on a slow time scale. FBT is in many respects similar to the codes described in Refs. [1,2], but it has several new features, not found in other codes: (a) numerical feedback is used to control the plasma shape as well as the position of magnetic axes and saddle points, (b) the shape is determined by specifying either exact boundary points or approximate boundary points or both, (c) the linearized Grad-Shafranov equation is solved non-iteratively by cyclic reduction [17] coupled with tridiagonal matrix inversion and (d) the code includes a stability test [18] which gives a good approximation to the ideal MHD growth rate of the most unstable axisymmetric mode.

2. THE PHYSICAL PROBLEM

Let us consider an axisymmetric toroidal plasma, with boundary S, surrounded by vacuum. The poloidal field coils are distributed arbitrarily in the vacuum region. We assume that there is no iron in the system. For convenience, a computational boundary, C is introduced in such a way that the plasma is completely inside C.

The free-boundary equilibrium problem can be divided into four steps, (a) the calculation of coil currents, subject to a number of constraints and optimization criteria, (b) the calculation of flux values on the computational boundary, using known coil currents and a fixed plasma current distribution, (c) the solution of the Grad-Shafranov equation inside C, with given flux values on C and a fixed plasma current, and (d) the calculation of the plasma current distribution, using given source functions. Of course, all four steps have to be solved simultaneously, in a self-consistent way, and this is usually done by an iterative procedure as will be discussed in the next section. The physical issues involved in steps (b), (c) and (d) have been amply discussed in the literature and will not be presented here. The treatment of step (a), however, is not at all obvious and merits special attention.

2.1 Constraints

Let us first express each coil current, C_i, as a sum of a fixed current, $V_{i,1}$, and a number of current moments, $V_{i,n}$, multiplied by scaling factors X_n,

$$C_i = V_{i,1} + \sum_{n=2}^{N_V} V_{i,n} X_n \quad , \quad i = 1 \ldots N_C \qquad (1)$$

The current moments, $V_{i,n}$ can either represent physical connections between different coils, or they can be specified in such a way as to produce particular flux patterns (vertical field, radial field, etc).

Equation (1) implies that the coil positions are fixed and independent of the parameters X_n. Consequently, the fluxes and magnetic fields produced by the external coils are linear functions of the X_n's. This allows the use of standard optimization techniques and guarantees a unique solution. If, on the other hand, the coil positions are functions of the X_n's, the problem becomes highly non-linear, and there may not be a unique solution.

In computing the unknown scaling factors X_n, various constraints may be imposed. We consider the following three options:

2.1.1 Exact plasma boundary points

The condition that the plasma boundary, defined by $\Psi = \Psi_{lim}$, must contain a number of fixed points in the (R,Z) plane, can be expressed in the form

$$\sum_{n=1}^{N_V} G_{\nu,n} X_n + \sum_{k=1}^{N_P} G_{\nu,k} J_k = \Psi_{lim} \quad , \nu = 1 \ldots N_E \qquad (2)$$

where the first and second sums represent the contributions of the external coils and the plasma, respectively. The summation over coils

is hidden in the quantity $G_{\nu,n}$, i.e.

$$G_{\nu,n} = \sum_{i=1}^{N_c} V_{i,n} \, G'_{\nu,i}$$

where $G'_{\nu,i}$ is the Green's function connecting the i-th coil with the ν-the boundary point. ($G'_{\nu,i}$ is the flux at the ν-th boundary point, produced by a unit current in the i-th coil.) The second sum in eq. (2) extends over all plasma current elements, i.e. $N_p = N_R * N_Z$ where N_R and N_Z are the numbers of radial and axial mesh points, respectively. $G_{\nu,k}$ is the Green's function connecting the k-th plasma current element with the ν-th boundary point. These Green's functions are, of course, functions of four variables, i.e. the coordinates of the source point and the coordinates of the target point in the R,Z plane. It should be noted that, in, eq. (2), the contribution of the fixed current moment, $V_{i,1}$, has been included in the summation by setting $X_1 \equiv 1$. Furthermore, the limiter flux , ψ_{lim}, is considered as an unknown, and we write $\psi_{lim} \equiv X_{N_V+1}$.

2.1.2 Points with vanishing radial or axial field

The condition that the radial or axial component of the magnetic field vanish at particular points in space, is written as

$$B_R = \sum_{n=1}^{N_V} D_{\tau,n} X_n + \sum_{k=1}^{N_p} D_{\tau,k} J_k = 0 \, , \quad \tau = 1 \ldots N_S \qquad (3)$$

or

$$B_z = \sum_{n=1}^{N_v} E_{\lambda,n} X_n + \sum_{k=1}^{N_p} E_{\lambda,k} J_k = 0, \quad \lambda = 1 \ldots N_T \qquad (4)$$

respectively. The D's and E's are spatial derivatives of Green's functions or linear combinations of such derivatives. $E_{\lambda,k}$, for example, is the Z-component of the magnetic field at the point λ, produced by a unit current at the position k. Again, the summation over coils is hidden in the quantities $E_{\tau,n}$ and $E_{\lambda,n}$.

2.1.3 Volt-sec constraint

It is often useful to prescribe the vacuum flux produced by the sum of all currents in the poloidal field coils at a particular point within the plasma. This condition can be expressed as

$$\sum_{n=1}^{N_v} G_{F,n} X_n = \Psi_F \qquad (5)$$

where $G_{F,n}$ is again a linear combination of Green's functions and Ψ_F is a fixed constant. The point F may be chosen anywhere in the (R,Z) plane, for example $R_F = R_0$, $Z_F = 0$, where R_0 is the major radius.

The constraints listed above are all optional in the sense that they may be introduced if their total number does not exceed the number of free parameters (X_n). However, the constraints are not necessary, as we shall see below.

2.2 Optimization

It is well known that if one fixes the plasma shape very precisely, e.g. by specifying a large number of exact boundary points (eq. 2), the coil currents tend to assume very large values. These large currents often appear in the form of dipoles, i.e. positive and negative currents in adjacent coils. In practice, one is looking for a solution which offers the best possible shape accuracy without excessively large coil currents. To achieve this, we first introduce approximate boundary points, with flux errors δ_μ,

$$\sum_{n=1}^{N_V} G_{\mu,n} X_n + \sum_{k=1}^{N_P} G_{\mu,k} J_k - \psi_{lim} = \delta_\mu \;, \quad \mu = 1 \ldots N_A \qquad (6)$$

The number of approximate boundary points, N_A, is arbitrary and is in no way related to the number of exact boundary points, N_E.

We then calculate the total resistive power dissipation in the poloidal field coils,

$$P = \sum_{i=1}^{N_C} d_i C_i^2 = \sum_{i=1}^{N_C} d_i \left(V_{i,1} + \sum_{n=2}^{N_V} V_{i,n} X_n \right)^2 \qquad (7)$$

where the d's are proportional to the coil radius and inversely proportional to the coil cross section.

In addition, we express the sum of the squares of the current differences in adjacent coils as

$$T = \sum_{i=1}^{N_c} (C_i - C_{i-1})^2$$

$$= \sum_{i=1}^{N_c} \left[V_{i,1} - V_{i-1,1} + \sum_{n=2}^{N_v} (V_{i,n} - V_{i-1,n}) X_n \right]^2 g_i \qquad (8)$$

where the weighting coefficients g_i are assumed as

$$g_i = \left[(R_i - R_{i-1})^2 + (Z_i - Z_{i-1})^2 \right]^{-1} \qquad (9)$$

and R_i, Z_i are the coordinates of the i-th coil. The choice of g_i reflects the fact current differences in coils which lie close together are much more severe than current differences in coils which lie far apart.

We now combine the three optimization criteria into a single function,

$$Q = \sum_{\mu=1}^{N_A} W_\mu \delta_\mu^2 + \sigma P + \gamma T \qquad (10)$$

which is to be minimized. The weighting coefficients, W_μ, σ and γ, can be chosen arbitrarily. They reflect the degree of shape accuracy one wishes to achieve in a particular application. For example, it may be useful to specify the coefficients W_μ as being inversely proportional to the square of the poloidal magnetic field. This leads to a uniform shape error for all boundary points. The poloidal field distribution on the plasma boundary is, of course, not known at this stage of the calculation. Approximate values, obtained from a previously calculated equilibrium, would have to be used. If one chooses

all W_μ's identical, then the flux errors will be uniform, and this can lead to large shape errors in the vicinity of saddle points, i.e. in areas of low poloidal field.

3. SOLUTION PROCEDURE

The physical problem, as outlined in the previous section, can be solved in many different ways. In FBT, we are using a Eulerian coordinate system, because our aim is to compute equilibria with arbitrary topology, having internal saddle points and multiple magnetic axes. Inverse methods [8,12,15], using the flux surfaces as coordinates, as well as moments methods [9], are rather inconvenient in such cases.

Since the free-boundary problem is intrinsically non-linear, it is solved by an iterative procedure [1,2]. The method used in FBT consists of the following steps.

3.1 Plasma current initialization

An initial estimate for the plasma current distribution is necessary in order to start the iteration procedure. If this initial estimate is very different from the desired final solution, the iteration may converge to an unwanted or unphysical solution. In FBT, we construct the initial current estimate by using the coordinates of the specified boundary points (both exact and approximate), assuming a parabolic current profile inside the plasma region. The current distribution is then normalized such that the total plasma current agrees with the prescribed value.

3.2 Calculation of coil currents

If the plasma current, J_k, is known, the constraints, eqs. (2)-(5), can be written in the form

$$\sum_{n=1}^{N_V} A_{\alpha,n} X_{n+1} + H_\alpha = 0 , \quad \alpha = 1 \ldots N_L \qquad (11)$$

where $A_{\alpha,n}$ is a known matrix, H_α a known vector, and N_L is the total number of constraints, $N_L = N_E + N_S + N_T + N_F$, N_F being either 1 or 0, depending on whether a Volt-sec constraint is applied or not. Remember that $X_1 \equiv 1$ and $X_{N_V+1} \equiv \psi_{lim}$.

The minimization of the function Q (eq. 10), subject to the constraints (11) is then achieved by introducing a function U,

$$U = Q + \sum_{\alpha=1}^{N_L} \lambda_\alpha \left[\sum_{n=1}^{N_V} A_{\alpha,n} X_{n+1} + H_\alpha \right] \qquad (12)$$

and by requiring that

$$\left.\begin{aligned} \frac{\partial U}{\partial X_{n+1}} &= 0 , \quad n = 1 \ldots N_V \\ \frac{\partial U}{\partial \lambda_\alpha} &= 0 , \quad \alpha = 1 \ldots N_L \end{aligned}\right\} \qquad (13)$$

λ_α being the Lagrange multipliers. (13) is a system of $N_V + N_L$ linear equations which can readily be solved for the unknowns X_n and λ_α. The coil currents then follow from (1). The system of equations

(13) always has a unique solution. Whether this solution is a physically reasonable one depends on the choice of the constraints (eqs. (2)-(5)), the current moments ($V_{i,n}$) and the weighting coefficients (W_μ, σ and γ). This will be discussed in section 7.

3.3 Flux values on the computational boundary

The boundary fluxes, Ψ_m, are given by

$$\Psi_m = \sum_{n=1}^{N_v} G_{m,n} X_n + \sum_{k=1}^{N_p} G_{m,k} J_k - \Psi_{lim} \qquad (14)$$

where m labels the position on the computational boundary and the G's are the usual Green's functions. The evaluation of the second sum in eq. (14) can be extremely time-consuming if it is done in a straightforward manner, using Green's functions. The use of Lackner's method[1] allows to speed up this calculation considerably. The method consists of projecting the plasma current onto the boundary C by solving the Grad-Shafranov equation with $\Psi_m = 0$, and then taking the normal derivative of Ψ on the boundary. The boundary fluxes Ψ_m are then evaluated from the fictitious boundary currents, where care must be taken in the treatment of the logarithmic divergence of the Green's function.

3.4 Solution of the Grad-Shafranov equation

The Grad-Shafranov equation,

$$\frac{\partial^2 \psi}{\partial z^2} + R \frac{\partial}{\partial R}\left[\frac{1}{R}\frac{\partial \psi}{\partial R}\right] = \mu_0 R p \qquad (15)$$

where ρ is the toroidal current density, is written in finite-difference form as

$$\psi_{i,j+1} + \psi_{i,j-1} + a_i \psi_{i+1,j} + b_i \psi_{i-1,j} + e_i \psi_{i,j} = f_{i,j} \qquad (16)$$

where $a_i = \left(\frac{\Delta Z}{\Delta R}\right)^2 \frac{R_i}{R_{i+1/2}}$

$b_i = \left(\frac{\Delta Z}{\Delta R}\right)^2 \frac{R_i}{R_{i-1/2}}$

$e_i = -2 - a_i - b_i$

$f_{i,j} = -\mu_0 R_i \rho_{i,j} (\Delta Z)^2 = -\mu_0 R_i J_{i,j} \frac{\Delta Z}{\Delta R}$

Here, the index k, used in previous sections, has been replaced by the indices i and j, where i labels the radial and j the axial position in the computational grid.

Equation (16) is solved by a fast, non-iterative algorithm, using cyclic reduction coupled with tridiagonal matrix inversion. This method is as fast as the conventional double cyclic reduction algorithm [17] and it has the advantage that the number of grid points in the radial direction does not have to be a power of 2.

3.5 Calculation of plasma current density

The toroidal plasma current density is given by

$$\frac{J_k^u}{\Delta R \, \Delta Z} = R p'(\psi_k) + \frac{1}{R \mu_0} T T'(\psi_k) \qquad (17)$$

where ΔR, ΔZ are the mesh sizes, $p(\Psi)$ and $T(\Psi)$ are the usual source functions, the prime denotes derivation with respect to Ψ and the superscript u stands for "unnormalized". The normalized current, J_k, is then obtained from

$$J_k = J_k^u \left(I_p \Big/ \sum_k J_k^u \right) \tag{18}$$

where I_p is the total plasma current.

At this point, the calculation switches back to step 3.2 and the steps 3.2 through 3.5 are repeated until a suitable convergence criterion is satisfied. We normally require that the maximum change between successive iterations in any flux value must be less than 10^{-7} times the difference between the flux on the magnetic axis and the limiter flux, i.e. $\varepsilon = 10^{-7}$.

4. CODE VALIDATION

4.1 Comparison with an analytic solution

The accuracy of the finite difference approximation (16) can be tested by comparing the numerical solution with an analytic solution. Since the discretization of the second derivative with respect to Z is obvious, $\partial^2\Psi/\partial Z^2 = (\Psi_{j+1} - 2\Psi_j + \Psi_{j-1})/\Delta Z^2$, we consider a case which contains only radial derivatives, i.e. an equilibrium which is infinitely long in the Z direction.

Let us assume that the plasma is bounded by the cylindrical surfaces R = 1 and R = 2, and that the toroidal current distribution is

given by

$$p(R) = (-2 + 3R - R^2) / \mu_o \tag{19}$$

An analytic solution of the Grad-Shafranov equation (15), assuming $\partial^2\Psi/\partial Z^2 = 0$, $\Psi(1) = 0$ and $\Psi(2) = 0$, is then readily obtained as

$$\psi(R) = \frac{R^5}{15} - \frac{3}{8}R^4 + \frac{2}{3}R^3 - \frac{133}{360}R^2 + \frac{1}{90} \tag{20}$$

We now solve eq. (15) numerically, using two different discretization schemes. The first scheme is the one given in eq. (16). The second scheme [2] uses partial derivatives, which leads to slightly different coefficients, i.e.,

$$a_i^* = \left(\frac{\Delta Z}{\Delta R}\right)^2 \frac{R_{i-1/2}}{R_i}$$

$$b_i^* = \left(\frac{\Delta Z}{\Delta R}\right)^2 \frac{R_{i+1/2}}{R_i} \tag{21}$$

$$e_i^* = -2 - 2\left(\frac{\Delta Z}{\Delta R}\right)^2$$

We compare analytical and numerical solutions and evaluate the maximum error as a function of the number of grid points (Fig. 1). We note that the convergence is strictly quadratic and that the finite difference scheme (16) is slightly more accurate than the one using the coefficients (21).

4.2 Comparison with Lackner's code

In order to test the FBT code as a whole, we compare its results with those obtained from Lackner's code [1]. For this purpose, we compute identical equilibria with both codes. Lackner's code was run at the Max-Planck-Institut für Plasmaphysik, Garching, in collaboration with Dr. K. Lackner [19]. The FBT code is run at EPFL, Lausanne. Identical source functions are used in both codes,

$$\left.\begin{aligned} p' &= c_p\left[1-(1-\varphi)^2\right] \\ TT' &= c_T\left[1-(1-\varphi)^2\right] \\ \varphi &= (\psi-\psi_{lim})/(\psi_{axis}-\psi_{lim}) \end{aligned}\right\} \qquad (22)$$

where c_p and c_T are constants. We also use the same coil positions and mesh size (65×65) in the two calculations. The FBT equilibria are computed with three exact boundary points and two current moments. The first moment produces a predominantly vertical field. It consists of eight identical currents (I_V), four positive ones in the inner coils and four negative ones in the outer coils. The second moment produces a radial field which is necessary for adjusting the vertical position of the plasma. In the converged solution, this second moment has a negligibly small coefficient, since we have chosen configurations which are exactly up-down symmetric.

In Table 1, we compare the results of the two codes. All currents have been normalized to the plasma current. The first equilibrium is roughly circular and has a high beta value (β_p = 1.5). The

second equilibrium is characterized by a double-null divertor, with divertor currents equal to 70% of the plasma current. We note that the results of the two calculations differ by less than 0.1%. In Figs 2 and 3, we compare the corresponding flux surface plots. They are practically indistinguishable.

5. BIFURCATIONS

The free-boundary equilibrium problem, as described in Section 2, is, of course, highly non-linear. Several solutions may exist for given initial conditions. We show here three examples of bifurcations which arise frequently in plasma shaping studies.

First, we consider a plasma with elliptical cross-section (Fig. 4). We define two fixed boundary points on the mid-plane ($Z = 0$). The fluxes at these two points are equalized by an adjustable vertical field. The elongation (b/a) is given by a fixed quadrupole field. We compute a series of equilibria with increasing quadrupole field. It is seen that, as the elongation grows, the convergence becomes slower. Figure 5 shows the number of iteration cycles necessary to reduce the maximum error in Ψ to 10^{-7}, as a function of the elongation, κ. For $\kappa \geq 1.5$, there is a slight vertical drift of the plasma as the iteration proceeds. Figure 6 shows the displacement of the magnetic axis as a function of the cycle number, for two values of κ. For $\kappa > 1.75$, a converged solution can no longer be obtained. This phenomenon can be avoided in several ways. One method consists of fixing the z-position of the magnetic axis by specifying $B_R = 0$ somewhere on the mid-plane (e.g. at $R = R_0$). Another method relies on intro-

ducing additional boundary points on top and bottom of the plasma. In both cases we need additional free parameters (i.e. variable current moments) to satisfy the new constraints. It is then possible to obtain converged solutions for elliptical equilibria with $\kappa > 1.75$.

However, when the elongation becomes very large, another instability appears. Let us consider a racetrack-shaped equilibrium with $\kappa = 3.6$ (Fig. 7). If we try to compute this equilibrium with four fixed boundary points (two on the mid-plane and one each on top and bottom), we observe that the magnetic axis drifts slowly in the vertical direction and that the displacement grows exponentially with the number of iteration cycles. At the same time, the plasma shape is gradually distorted from the original racetrack into a pear-shape. The instability can again be suppressed by using additional boundary points (Fig. 8).

A third instability can be observed when the boundary points are placed diagnonally. We again consider the racetrack equilibrium with $\kappa = 3.6$ (Fig. 7). The boundary points are now placed in the four corners where the tangent to the flux surface forms an angle of 60° with the horizontal axis. In this case, the iteration is highly unstable, leading first to a doublet shaped plasma with an internal separatrix and two magnetic axes, and finally to a complete breakup of the plasma into two droplet-shaped configurations (Fig. 9). This mode is easily stabilized by introducing two additional boundary points, lying on the mid-plane.

It is interesting to note that all three numerical instabilities discussed above have their analogue among the physical instabilities of elongated tokamaks [20,21].

6. GROWTH RATES OF AXISYMMETRIC MODES

A free-boundary equilibrium code, such as FBT, lends itself readily for computing global axisymmetric instabilities of tokamak plasmas. The method has been described in detail in Ref. [18]. The basic concept is as follows: We first compute an initial equilibrium and a number of neighbouring equilibria with slight changes in plasma position, shape and source function parameters. We then construct linear combinations of these neighbouring equilibria in order to find displacement vectors which are compatible with ideal MHD constraints. The growth rate of the axisymmetric vertical instability can then be obtained from the radial magnetic field which is necessary to keep the displaced plasma in equilibrium. Results obtained for up-down symmetric configurations have been shown to agree with ERATO calculations [18].

As an example, we show here the results of a stability study of elongated racetrack and D-shaped plasmas in a rectangular conducting shell. Figure 10 shows the values of q_a/q_o for marginal stability, as a function of elongation, for two different values of the plasma-wall distance, Δ. When the plasma elongation is varied, the size of the shell remains fixed, and the plasma is always placed within the shell in such a way that the minimum plasma-wall distance on the top is the same as that on the two sides. For each of the four curves shown in Fig. 10, the stable domain is below and the unstabale domain above the curve. We note that, from the point of view of maximum allowable (q_a/q_o) values, the D-shaped plasmas are considerably more stable against axisymmetric modes than the racetracks.

7. APPLICATIONS

The FBT code has been applied to the computation of a wide variety of tokamak equilibria. Typical examples are presented in the following paragraphs.

7.1 Shape accuracy

In order to illustrate the conflicting requirements of shape accuracy and low power dissipation in the poloidal field coils, we consider a triangular equilibrium (Fig. 11). The plasma shape is defined by twenty approximate boundary points. Three of these points lie on the corners of the triangle, whereas the others are distributed evenly along the sides. Sixteen poloidal field coils are used to generate these equilibria, and the moments matrix $(V_{i,n})$ is assumed strictly diagonal. The weighting coefficients (Eq. 10) are defined as follows: The boundary points all have the same weight, $W_\mu = 1$. The weight of the power dissipation, P, is varied by assuming three different values of σ (0.1, 0.001, 0.0). The dipole term is not used ($\gamma = 0$).

The results (Fig. 11) show that, as σ becomes smaller and smaller, the plasma shape approaches the prescribed triangular shape, but the coil currents increase dramatically. Of course, even with $\sigma = 0$, the shape cannot be an exact triangle because the number of poloidal field coils is relatively small.

7.2 Startup Evolution

Startup scenarios for non-circular tokamaks usually begin with a circular plasma which is then stretched and deformed until it reaches the desired final shape. The plasma current is ramped up simultaneously, such that the q-value remains approximately constant during the evolution. In order to implement such a scenario in a tokamak, one needs to know the shaping currents as functions of time. These can be obtained by computing a number of free-boundary equilibria, simulating the desired shape evolution. An example of such a calculation is shown in Fig. 12. Thirty equilibria were computed to describe the transformation of a circular plasma into a bean-shaped one (Fig. 12 shows only every third equilibrium). Sixteen approximate boundary points were used to define the shape of each equilibrium. In the first equilibrium, the sixteen boundary points lie on a perfect circle, whereas in the last one, they are evenly distributed around the circumference of a bean shape. The intermediate equilibria are then defined by assuming that each boundary point follows a linear trajectory and that this trajectory is divided into thirty equal steps. Twenty-four shaping coils and sixteen independent current moments were used to generate the equilibria shown in Fig. 12. The plasma current is assumed to scale with elongation as $I_p \sim (\kappa^2+1)\kappa^{0.34}$ [22]. The weighting coefficients (Eq. 10) are taken as $W_\mu = 1$, $\sigma = 0.001$, $\gamma = 0$. Equilibrium source functions are identical with the ones used in Ref. [21]. The resulting coil currents are plotted as functions of the step number in Fig. 13.

In contrast to previous studies of shape programming [23], the plasma evolution considered here is assumed to take place on a very

slow time scale. Therefore, magnetic fluxes are not conserved in going from one equilibrium to the next. We also note that the evolution is asymmetric with respect to the mid-plane (Fig. 12). This is done in order to take advantage of the stabilizing effect of the top wall. At the end of the evolution, however, the equilibrium becomes exactly up-down symmetric and, hence, the coil currents which lie symmetric with respect to the mid-plane become identical (Fig. 13).

When we compare the actual and the prescribed plasma shapes, we find that they are practically the same, yet the coil currents are not excessively large. This is possible because the equilibria considered here do not contain very high order moments, in contrast to the triangular equilibria discussed previously.

7.3 Saddle Point Control

In many tokamaks, the plasma boundary is defined by a saddle point of the poloidal flux function $\Psi(R,Z)$. The position of this saddle point must be actively controlled throughout the discharge in order to avoid excessive heat loads on the divertor plates [24]. Using the FBT code, we can easily compute the coil currents which are necessary to produce a saddle point anywhere on the plasma boundary. For this purpose, we impose the conditions (2), (3) and (4) at the desired point in the (R,Z) plane. Figure 14 shows examples of divertor equilibria obtained in this way. In this sequence, the Z-coordinate of the saddle point is left constant, whereas the R-coordinate is increased by 0.04m between successive equilibria (R_0 = 0.88m). It should be noted that none of the poloidal field coils can easily be identified as the "divertor coil". The saddle point is produced by the combined

action of several coils.

An example of an internal saddle point is shown in Fig. 15. Here, the change in shape produces a separatrix inside the plasma region, which leads to a configuration with two magnetic axes.

8. DISCUSSION

Free-boundary MHD equilibrium calculations for circular or near-circular plasmas are relatively easy to perform and pose no particular convergence problems. However, when the plasma shapes become more complicated, various numerical instabilities appear. We have shown that these instabilities can be suppressed by feedback control of the plasma shape. The algorithm used in FBT continuously adjusts the coil currents in order to maintain a certain number of flux values on the plasma boundary equal to the limiter flux. FBT does not need complicated current initialization procedures or fictitious coils inside the plasma [25] to obtain a particular solution branch.

The combination of approximate and exact boundary points, which remain fixed throughout the iteration, has proven to be a very useful tool for creating free-boundary equilibria with a predetermined shape. In addition, magnetic axes and saddle points can be arbitrarily specified by imposing a vanishing poloidal magnetic field at a particular point. The code uses a fast, non-iterative Grad-Shafranov solver. Typical computation times are approximately 2 sec on a CRAY-1S machine, using 65 × 65 mesh points. Finally, an extended version of the FBT code [18] allows the computation of vertical instability growth rates of elongated tokamak plasmas.

9. ACKNOWLEDGEMENTS

Fruitful discussions with Professor F. Troyon, Drs. K. Lackner, S.C. Jardin, R. Gruber and F.B. Marcus are gratefully acknowledged. This work was partly supported by the Swiss National Science Foundation.

REFERENCES

[1] K. Von Hagenow and K. Lackner, in Numerical Simulation of Plasmas (Proc. 7th Conf. New York, 1975) p. 140; K. Lackner, Comput. Phys. Commun. 12 (1976) 33.

[2] J.L. Johnson et al., J. Comput. Phys. 32 (1979) 212.

[3] J. Blum, J. le Foll and B. Thooris, Comput. Phys. Commun. 24 (1981) 235.

[4] W. Feneberg, K. Lackner, Nucl. Fusion 13 (1973) 549.

[5] M.S. Chu et al., Phys. Fluids 17 (1974) 1183; F.J. Helton and T.S. Wang, Nucl. Fusion 18 (1978) 1523.

[6] Y. Suzuki, Nucl. Fusion 14 (1974) 345.

[7] G. Cenacchi, R. Galvao, A. Taroni, Nucl. Fusion 16 (1976) 457.

[8] J. DeLucia, S.J. Jardin and A.M.M. Todd, J. Comput. Phys. 37 (1980) 183.

[9] L.L. Lao, Comput. Phys. Commun. 31 (1984) 201.

[10] R.L. Miller, Nucl. Fusion 20 (1980) 133.

[11] S.P. Hirshman and S.C. Jardin, Phys. Fluids 22 (1979) 731.

[12] S.J. Jardin, J. Comput. Phys. 43 (1981) 31.

[13] J.T. Hogan, Nucl. Fusion 19 (1979) 753.

[14] S.C. Jardin, N. Pomphrey and J. DeLucia, J. Comput. Phys. 66 (1986) 481.

[15] R. Chodura and A. Schlüter, J. Comput. Phys. 41 (1981) 68.

[16] A. Schlüter and U. Schwenn, Comput. Phys. Commun. 24 (1981) 263.

[17] O. Buneman, SUIPR Report No 294 (Stanford, 1969)

[18] F. Hofmann, F.B. Marcus and A.D. Turnbull, Plasma Phys. and Controlled Fusion 28 (1986) 705.

[19] K. Lackner, private communication (1982)

[20] J.K. Lee, Nuclear Fusion 26 (1986) 955.

[21] F. Hofmann, A.D. Turnbull, F.B. Marcus, Nuclear Fusion 27 (1987) 743.

[22] F. Hofmann, S.C. Jardin, F.B. Marcus, A. Perez and A.D. Turnbull in Fusion Technology 1986 (Proc. 14th Symposium on Fusion Technology, Avignon, 1986) Vol. I, p. 687 (Pergamon Press 1986).

[23] J.A. Holmes, Y.-K.M. Peng and S.J. Lynch, J. Comput. Phys. 36 (1980) 35.

[24] R. Parker et al. in Controlled Fusion and Plasma Physics (Proc. 14th Europ. Conf., Madrid 1987) Vol. 11D, Part I, European Physical Society (1987) 301.

[25] T.S. Wang and F.J. Helton, Comput. Phys. Comm. 24 (1981) 255.

Table 1

Comparison between results of Lackner's code and FBT code

	Equilibrium 1 high beta, circular		Equilibrium 2 double-null divertor	
	Lackner's code KAL 326-4	FBT C131A	Lackner's code KAL 331-1	FBT C138
I_v/I_p	0.16598	0.1661	0.12492	0.1250
Ψ_{ax}(Vs)	0.9268×10^{-2}	0.9265×10^{-2}	0.7770×10^{-2}	0.7776×10^{-2}
R_{ax}(m)	0.6700	0.6701	0.6250	0.6249

FIGURE CAPTIONS

Fig. 1 Maximum relative error of numerical solutions for a one-dimensional equilibrium, as a function of the number of radial mesh points (NR). A: eq(16), B: eq(21).

Fig. 2 Flux surface plots for a near-circular, high-beta equilibrium. (A): Lackner's code, (B): FBT code. Poloidal field coils are indicated by open circles or crosses. Plasma boundary is shown as solid line (A) or dotted line (B).

Fig. 3 Flux surface plots for a double-null divertor equilibrium. (A): Lackner's code, (B): FBT code.

Fig. 4 Elliptical equilibrium with two fixed boundary points (open circles).

Fig. 5 Number of iteration cycles for convergence ($\varepsilon = 10^{-7}$), as a function of elongation, κ, for elliptical equilibria of the type shown in Fig. 4.

Fig. 6 Displacement of the magnetic axis as a function of the iteration cycle number, for two values of elongation.

Fig. 7 Racetrack-shaped equilibrium with elongation $\kappa = 3.6$.

Fig. 8 Displacement of magnetic axis of racetrack equilibrium ($\kappa = 3.6$), as a function of iteration cycle number, using four (open circles) or sixteen (crosses) fixed boundary points.

Fig. 9 Numerical instability in racetrack equilibrium calculation, using four boundary points placed diagonally (open circles).

Fig. 10 Ratio of q-values on the surface (q_a) and on axis (q_0) for marginal axisymmetric stability, as a function of elongation (κ), for various D-shaped (D) and racetrack (RT) equilibria. Δ is the minimum plasma-wall distance. Major radius, $R_0 = 0.88$ m.

Fig. 11 Triangular equilibria with increasing precision in shape. $\sigma = 0.1$, 0.001, and 0.0 in cases A, B, and C, respectively. Sum of absolute values of coil currents, divided by plasma current, increases from 2.92 (A), to 6.44 (B), to 42.53 (C).

Fig. 12 Startup evolution of an elongated tokamak. Profiles of pressure (dotted line) and current density (solid line), taken at the height of the magnetic axis, are shown on top of each equilibrium.

Fig. 13 Evolution of plasma current (I_p) and coil currents (1-16) for shape transformation of Fig. 12.

Fig. 14 Equilibria with saddle point at predetermined position.

Fig. 15 Transformation of a racetrack equilibrium into a doublet.

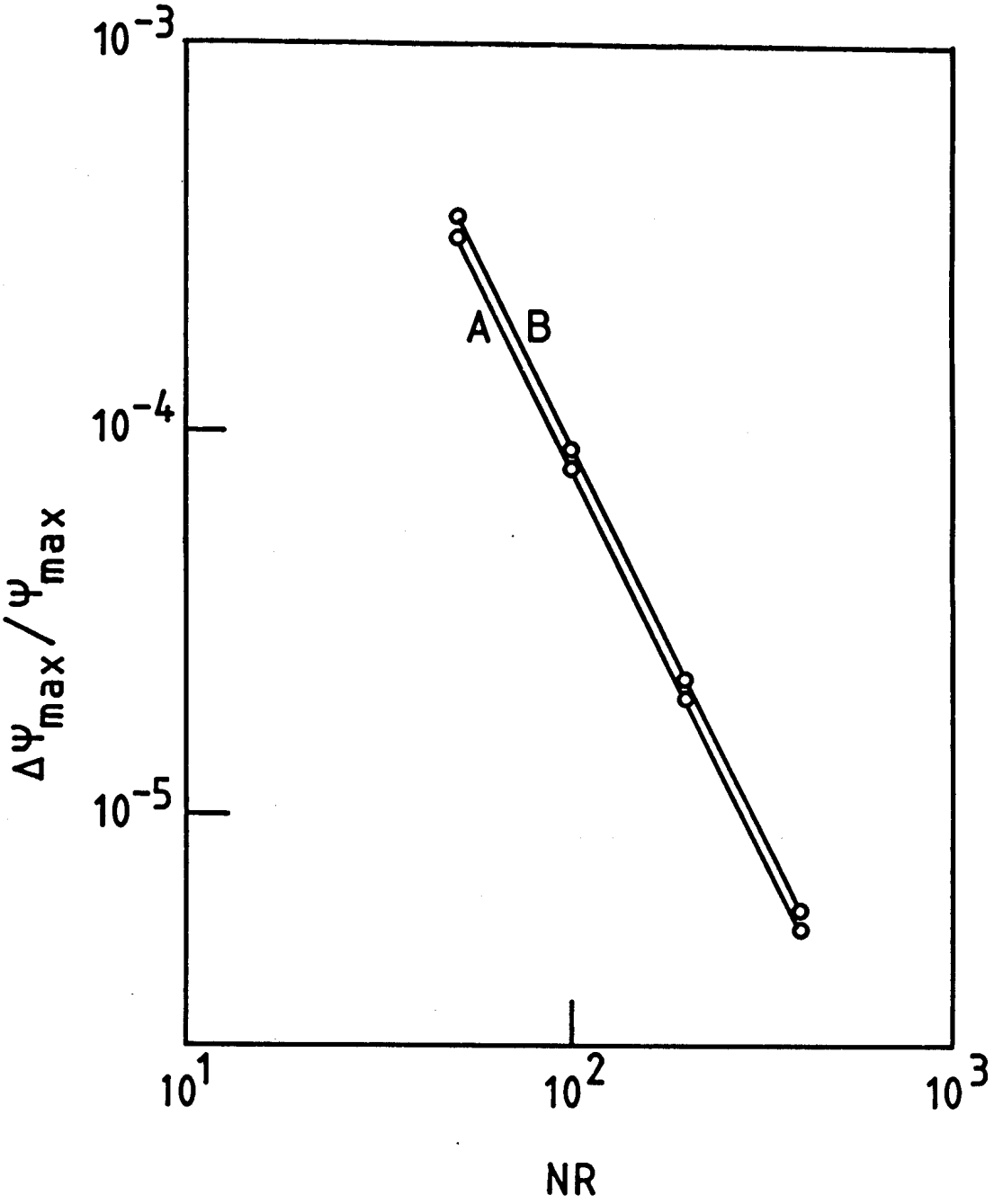

FIG.1

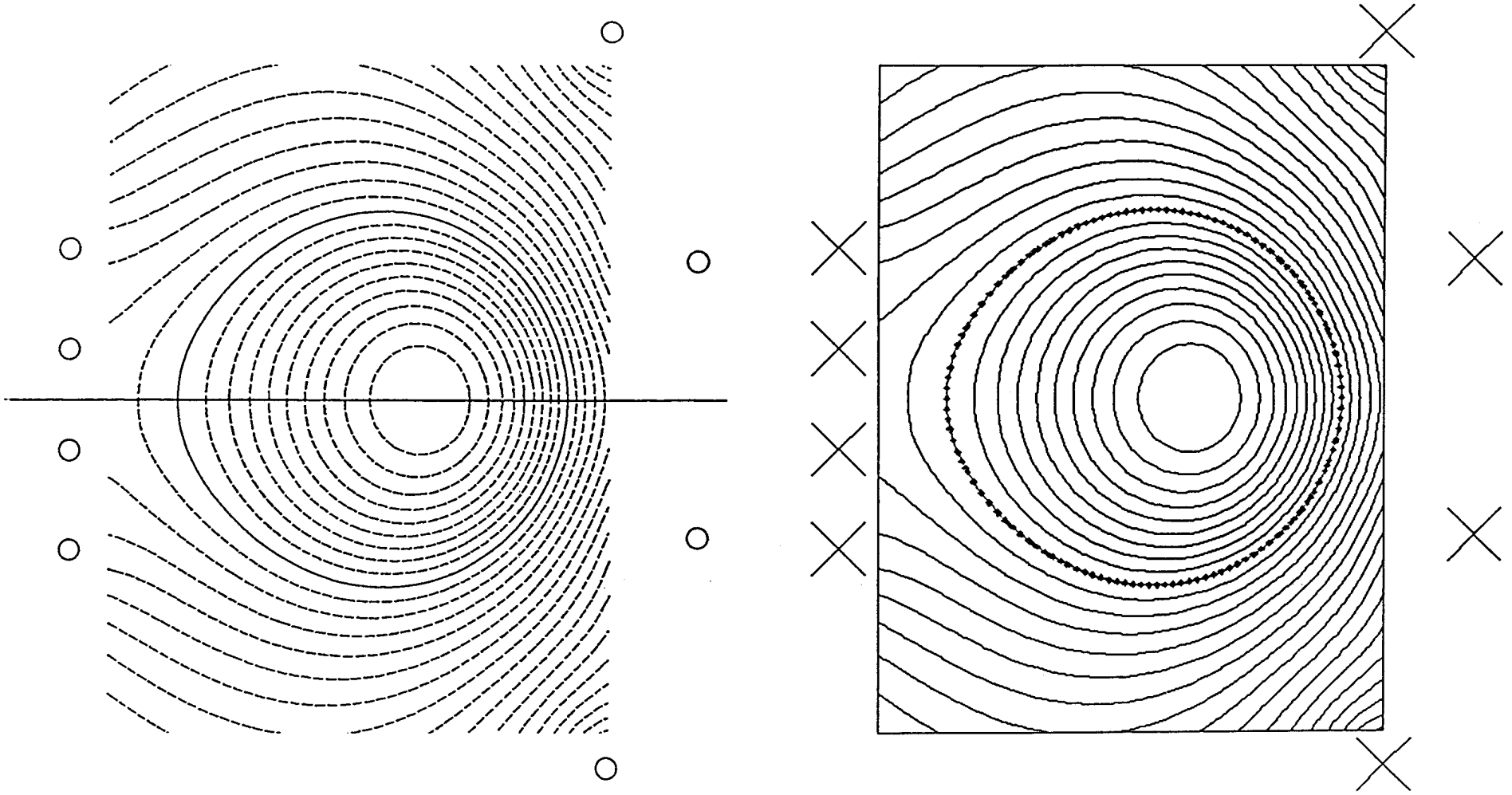

FIG. 2

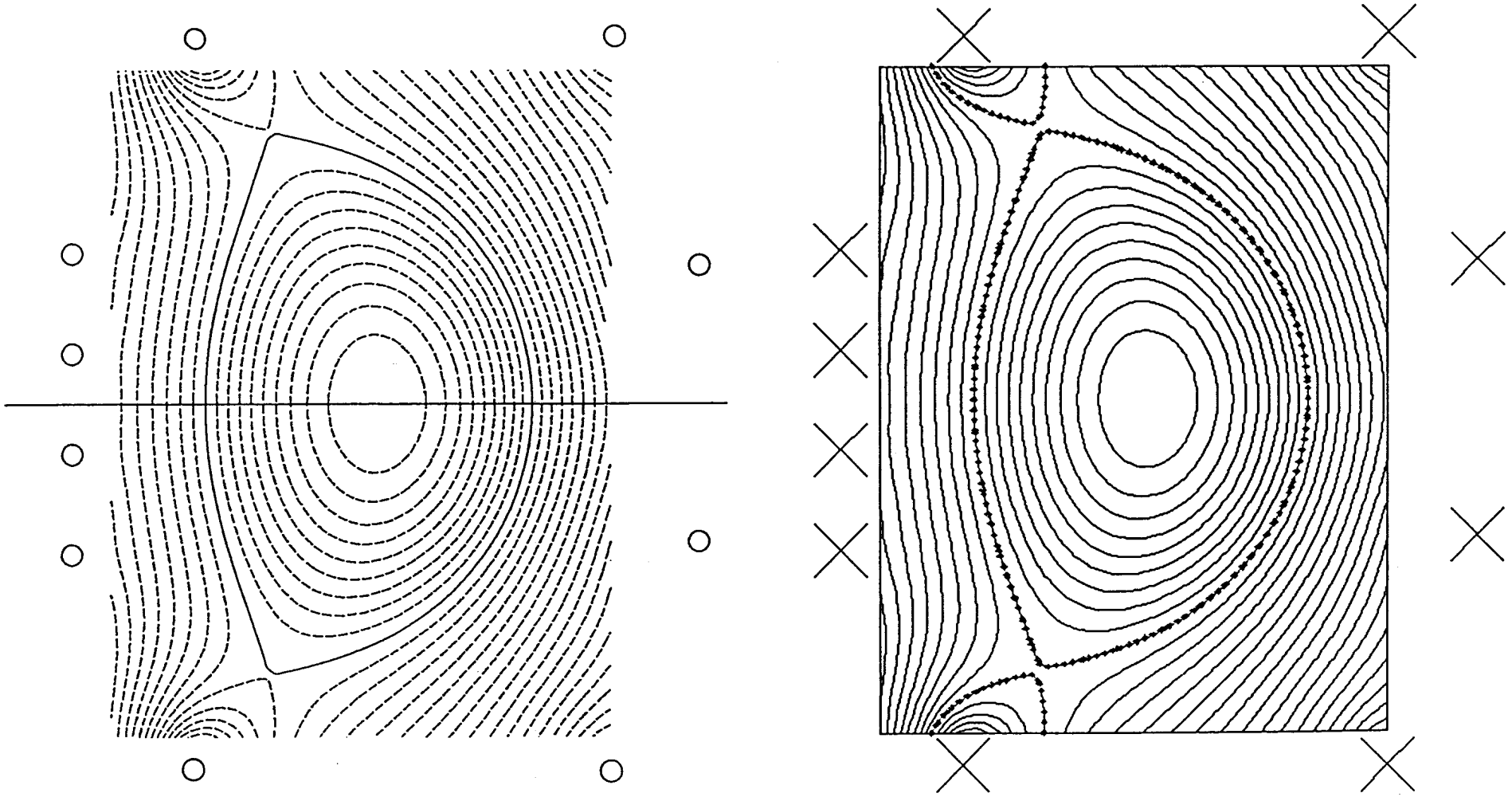

FIG. 3

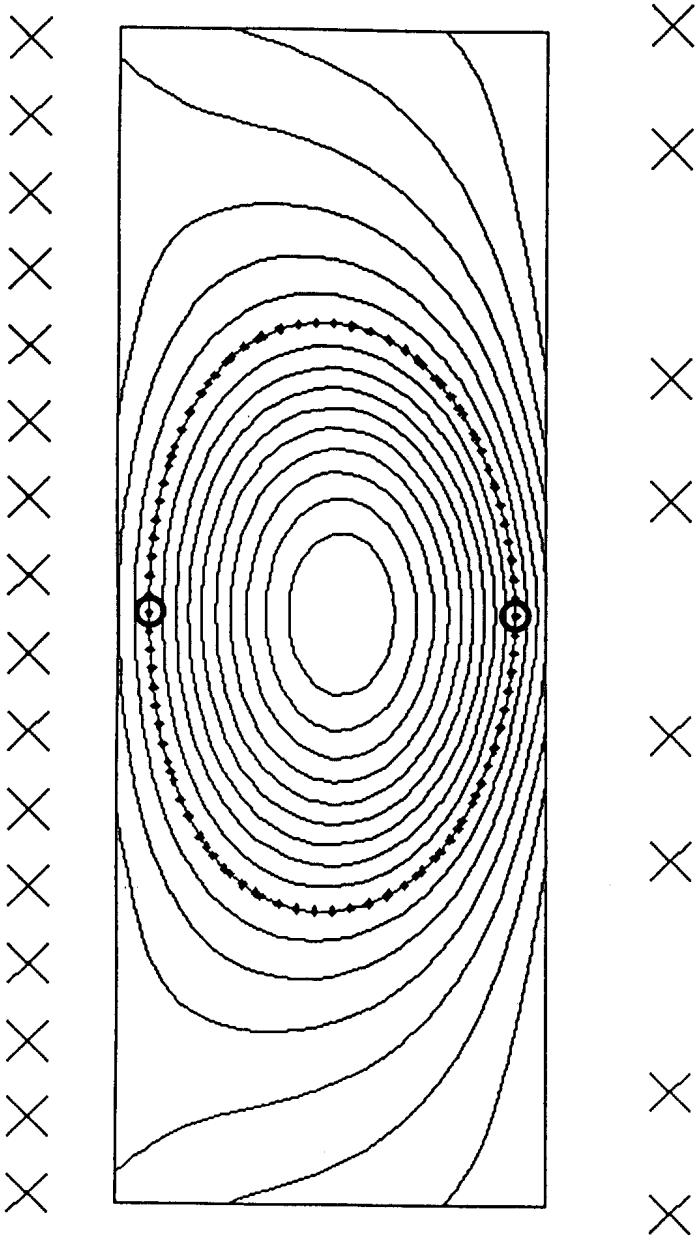

FIG. 4

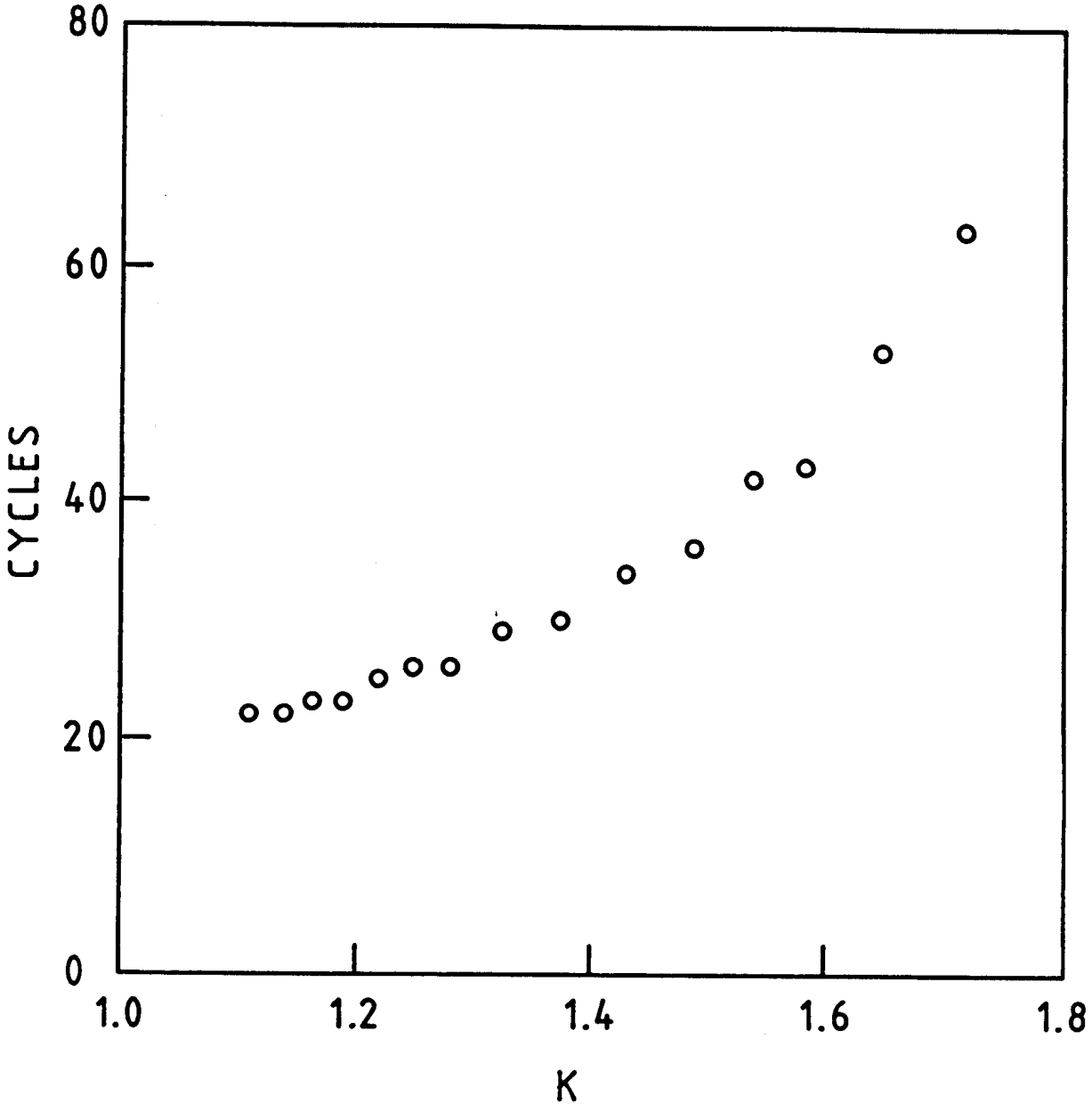

FIG.5

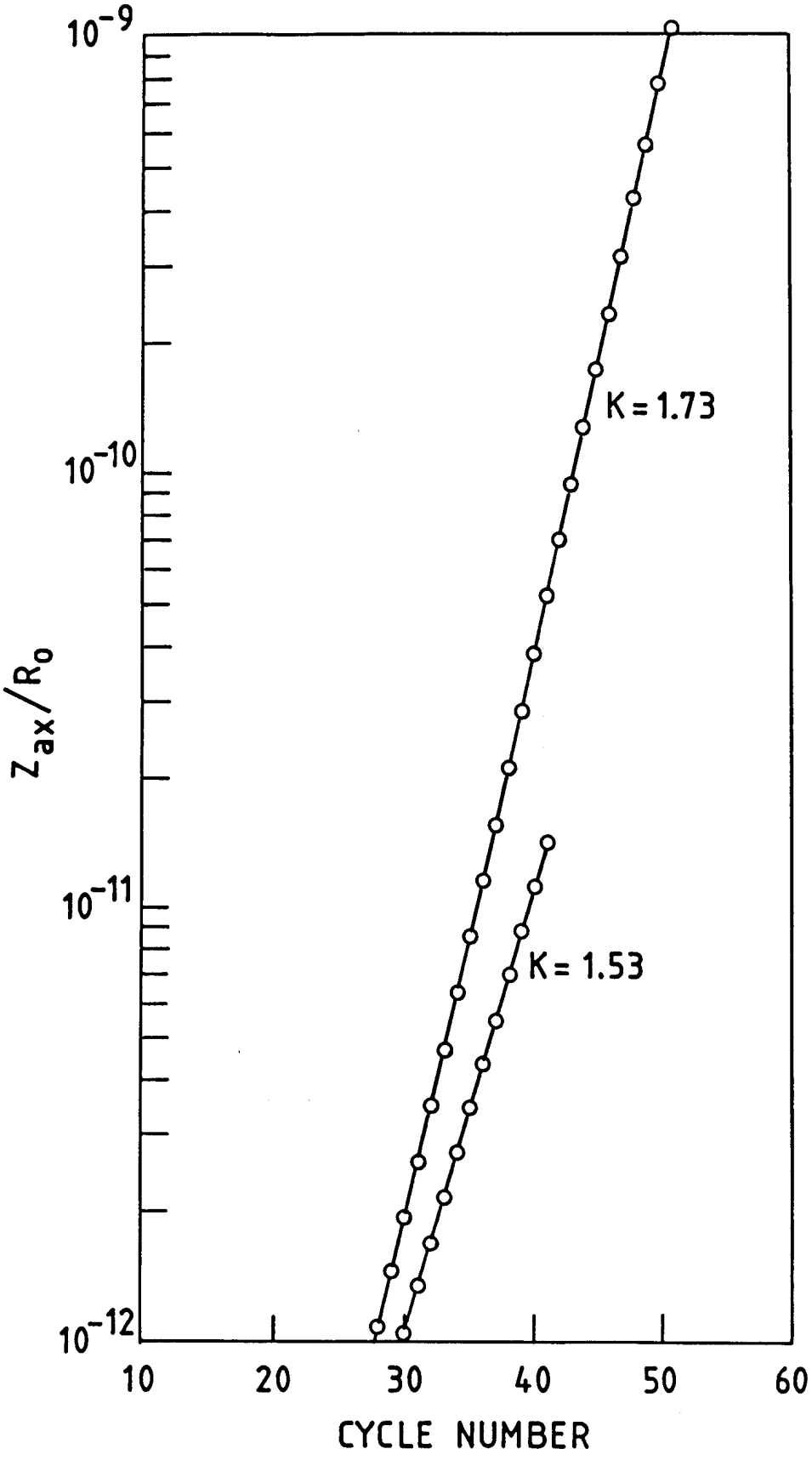
10^{-9}
10^{-10}
10^{-11}
10^{-12}
Z_{ax}/R_o
K = 1.73
K = 1.53
10
20
30
40
50
60
CYCLE NUMBER

FIG. 6

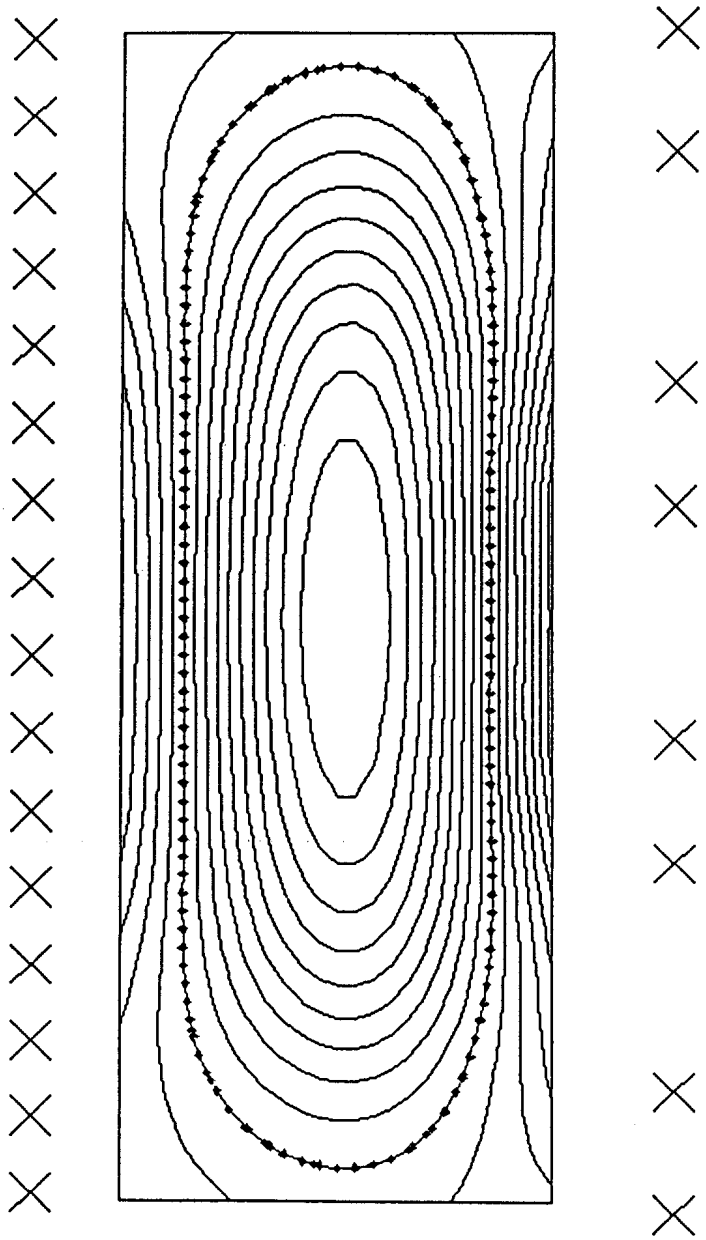

FIG. 7

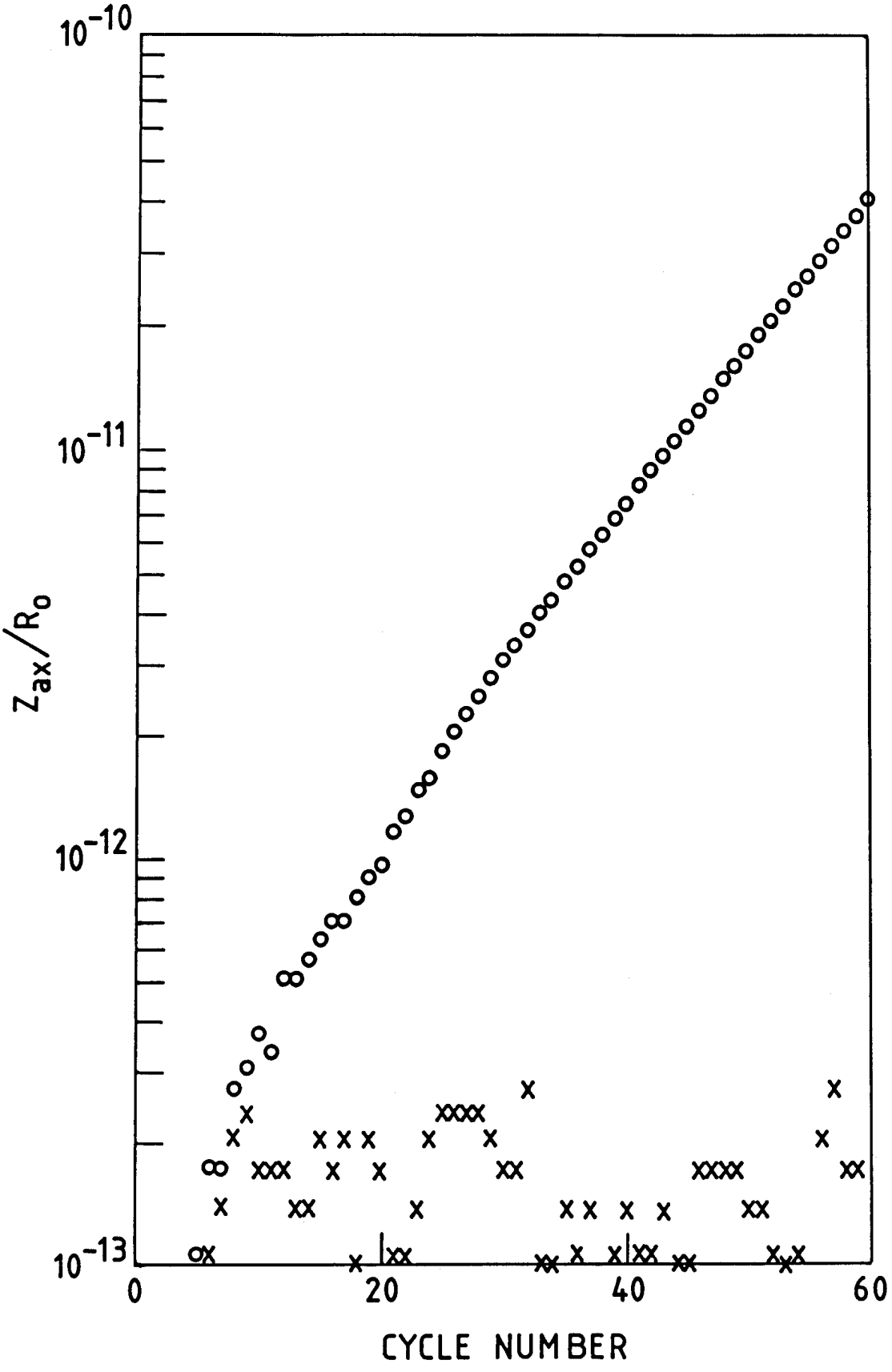

FIG. 8

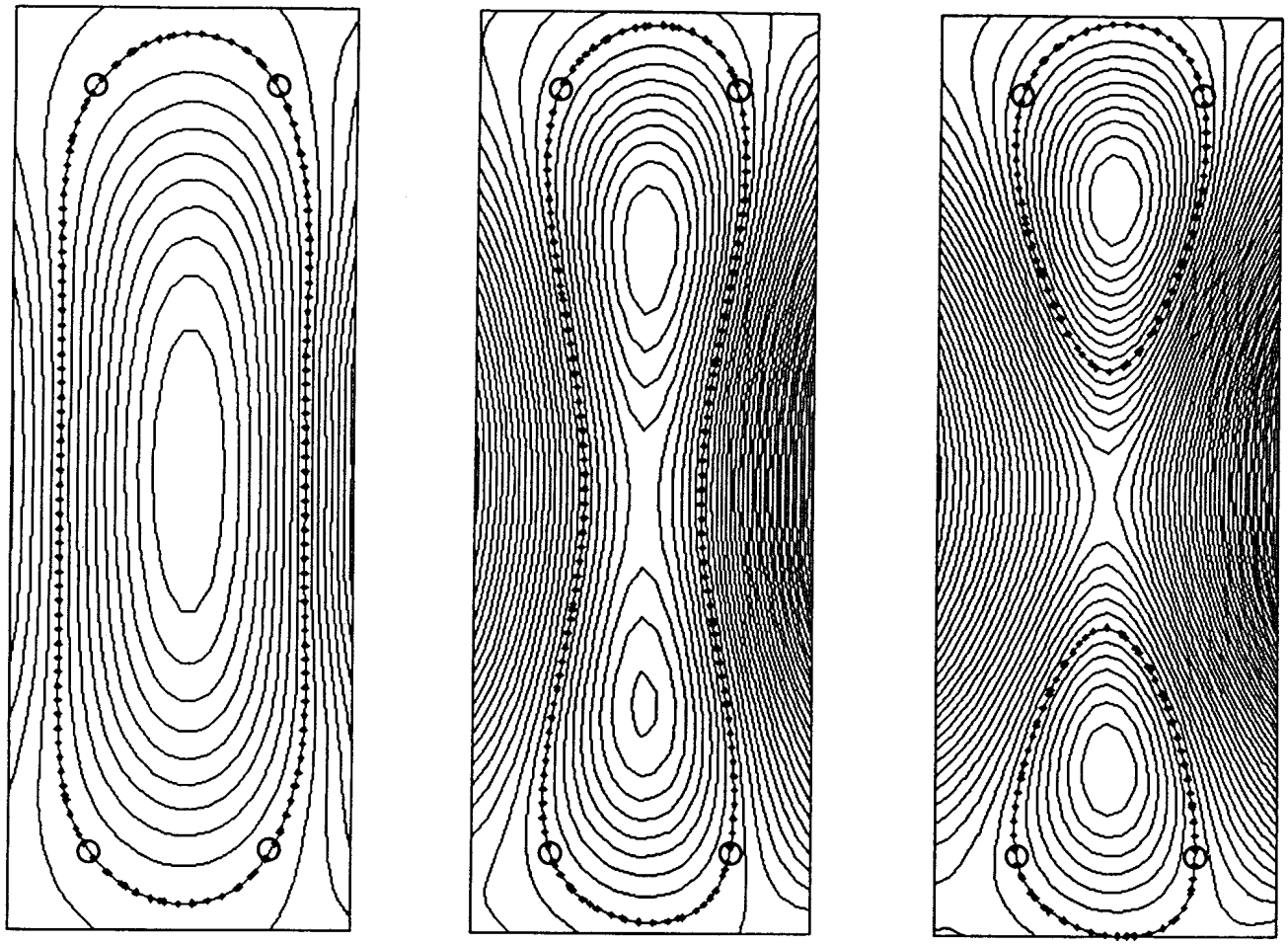

FIG. 9

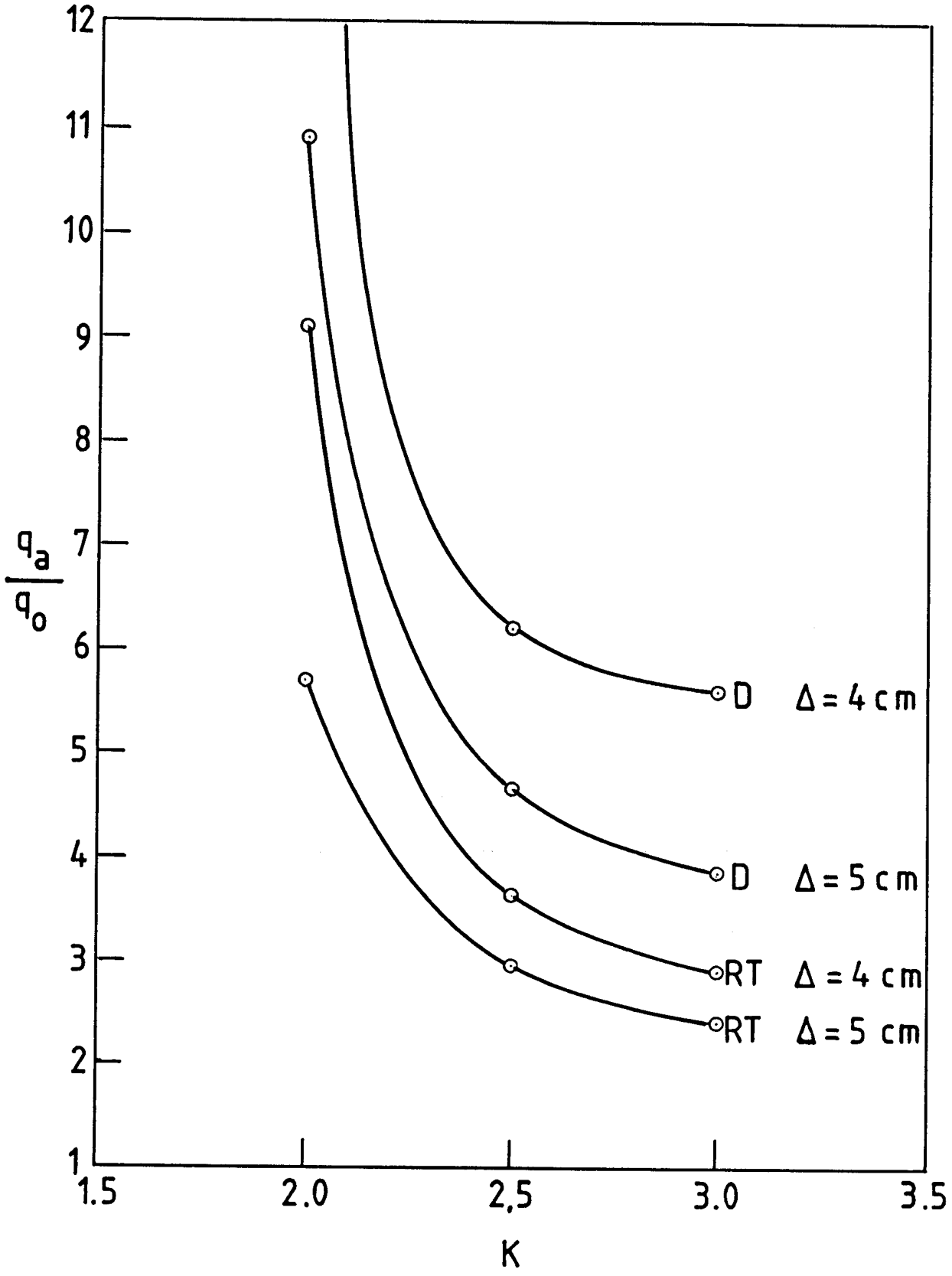
12
11
10
9
8
7
6
5
4
3
2
1
$\frac{q_a}{q_o}$
1.5
2.0
2,5
3.0
3.5
K
D Δ = 4 cm
D Δ = 5 cm
RT Δ = 4 cm
RT Δ = 5 cm

FIG.10

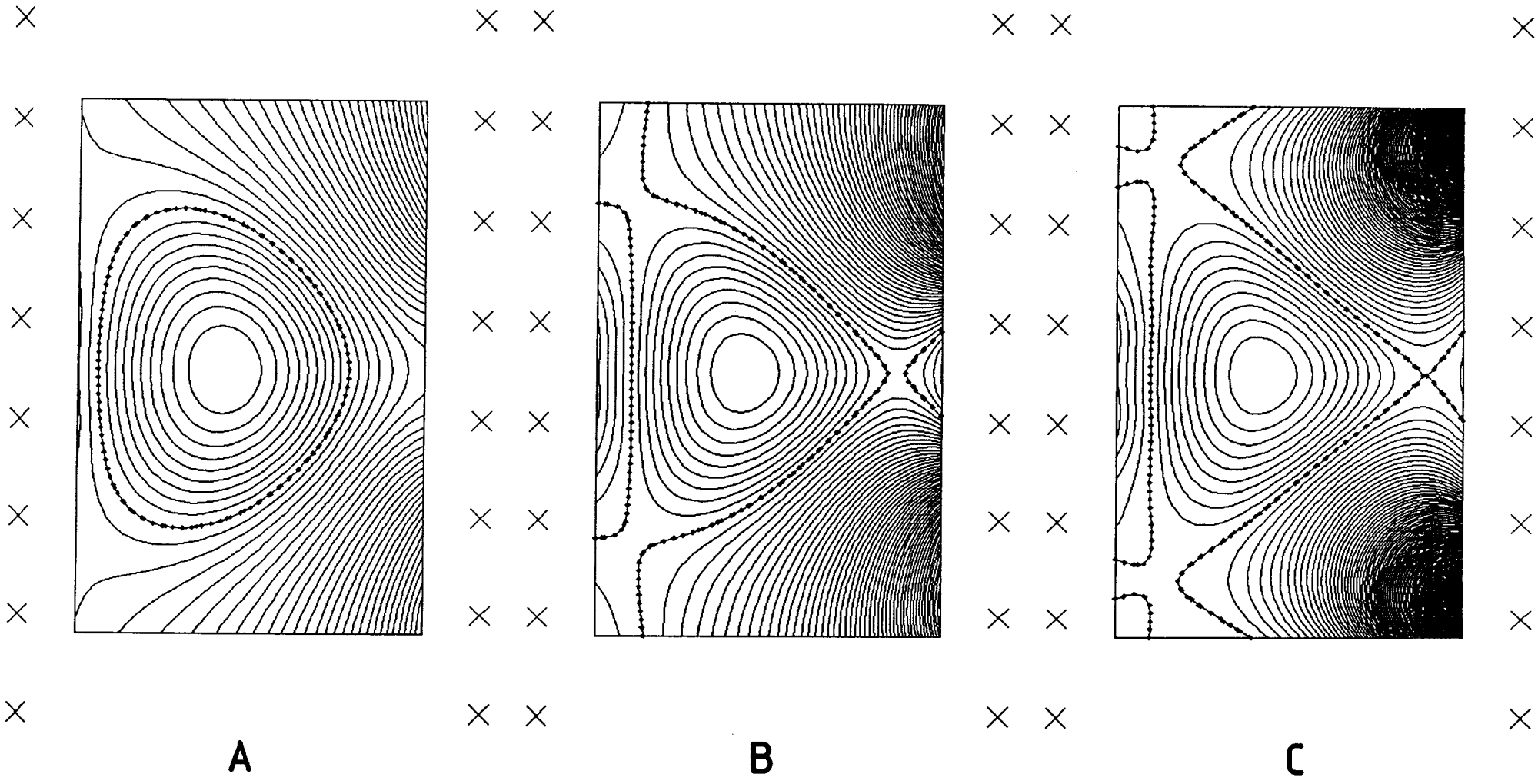

FIG.11

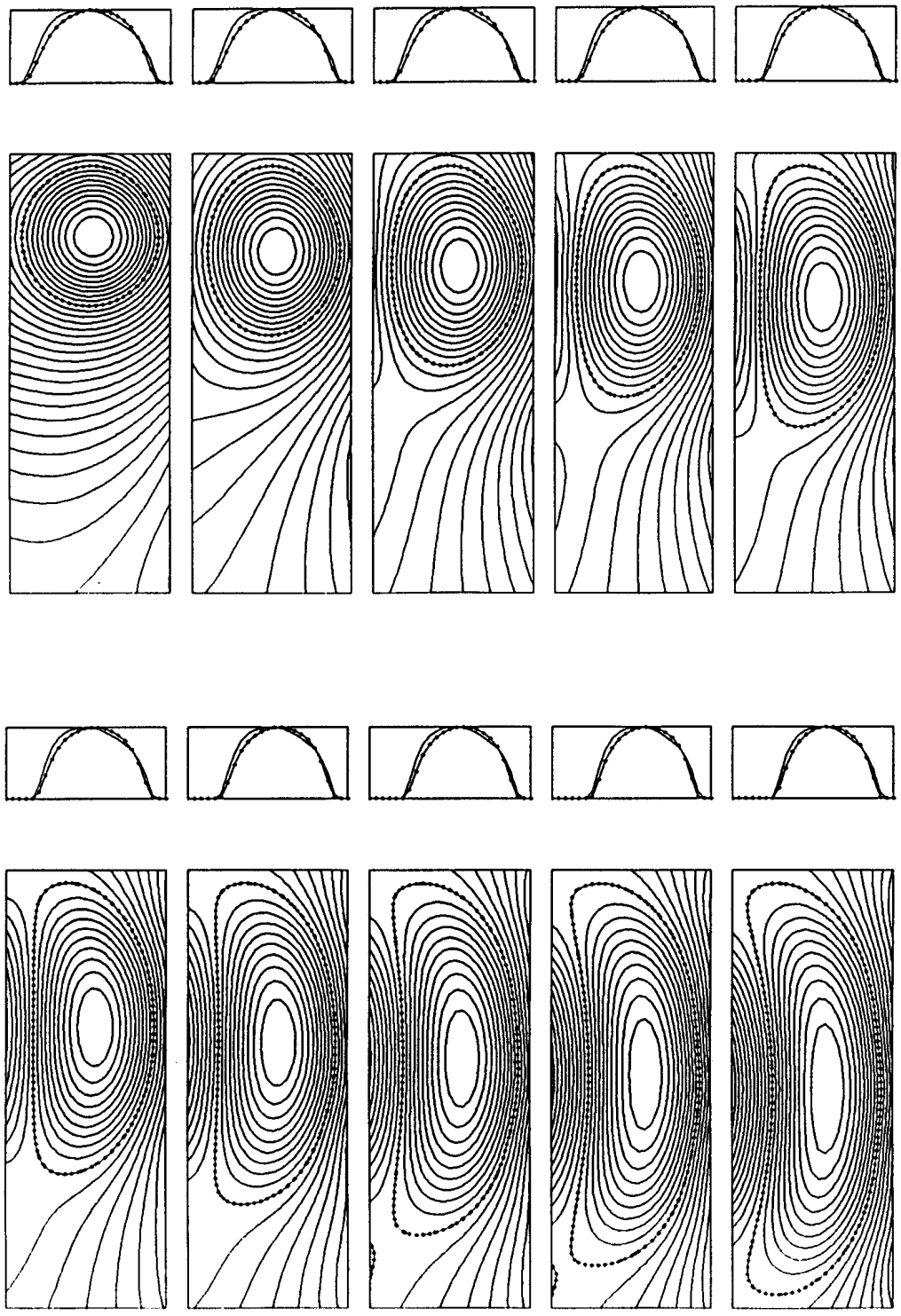

FIG.12

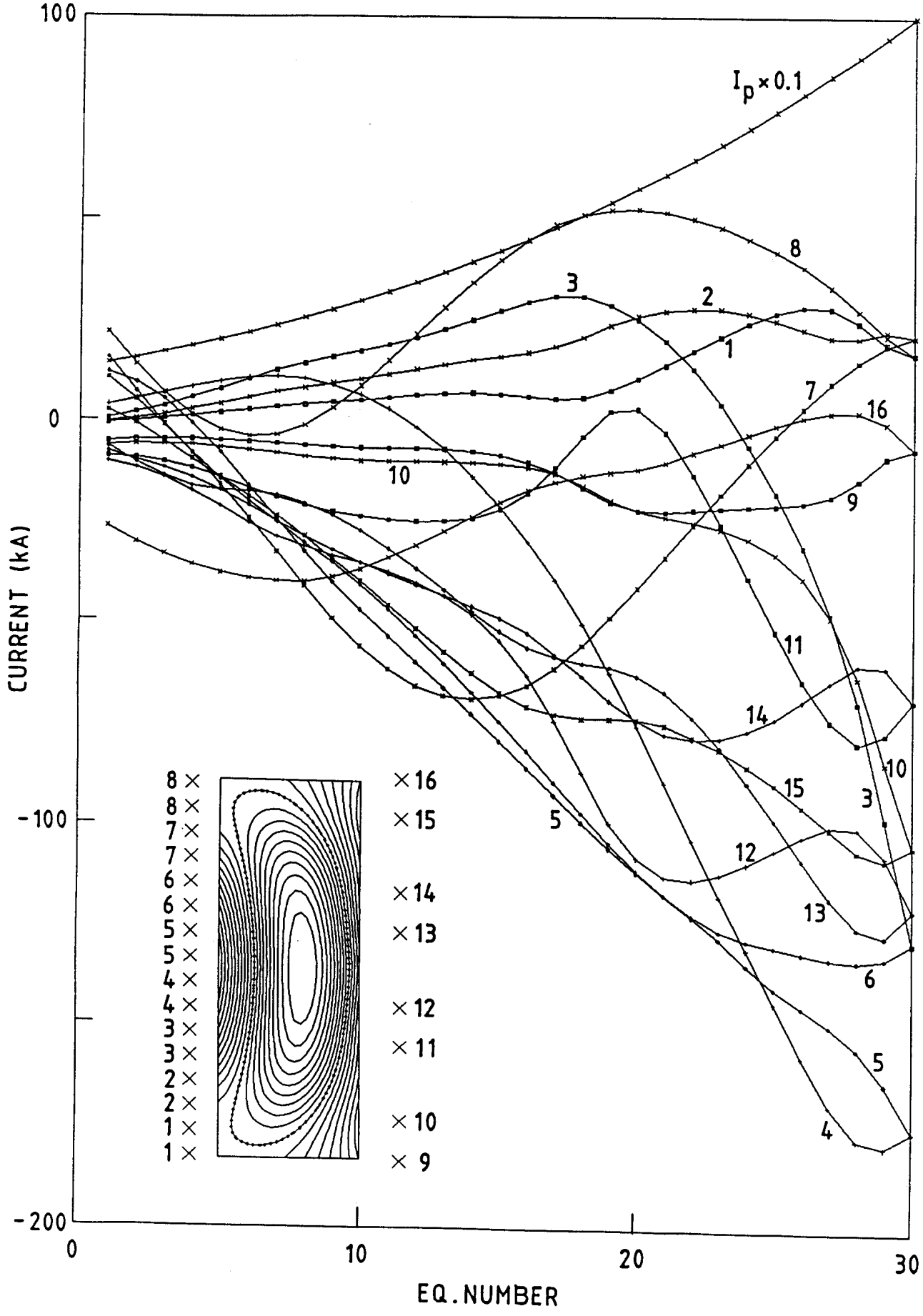
100
0
-100
-200
0
10
20
30
CURRENT (kA)
EQ. NUMBER
$I_p \times 0.1$
8 × 8 × 7 × 7 × 6 × 6 × 5 × 5 × 4 × 4 × 3 × 3 × 2 × 2 × 1 × 1 ×
× 16 × 15 × 14 × 13 × 12 × 11 × 10 × 9

FIG.13

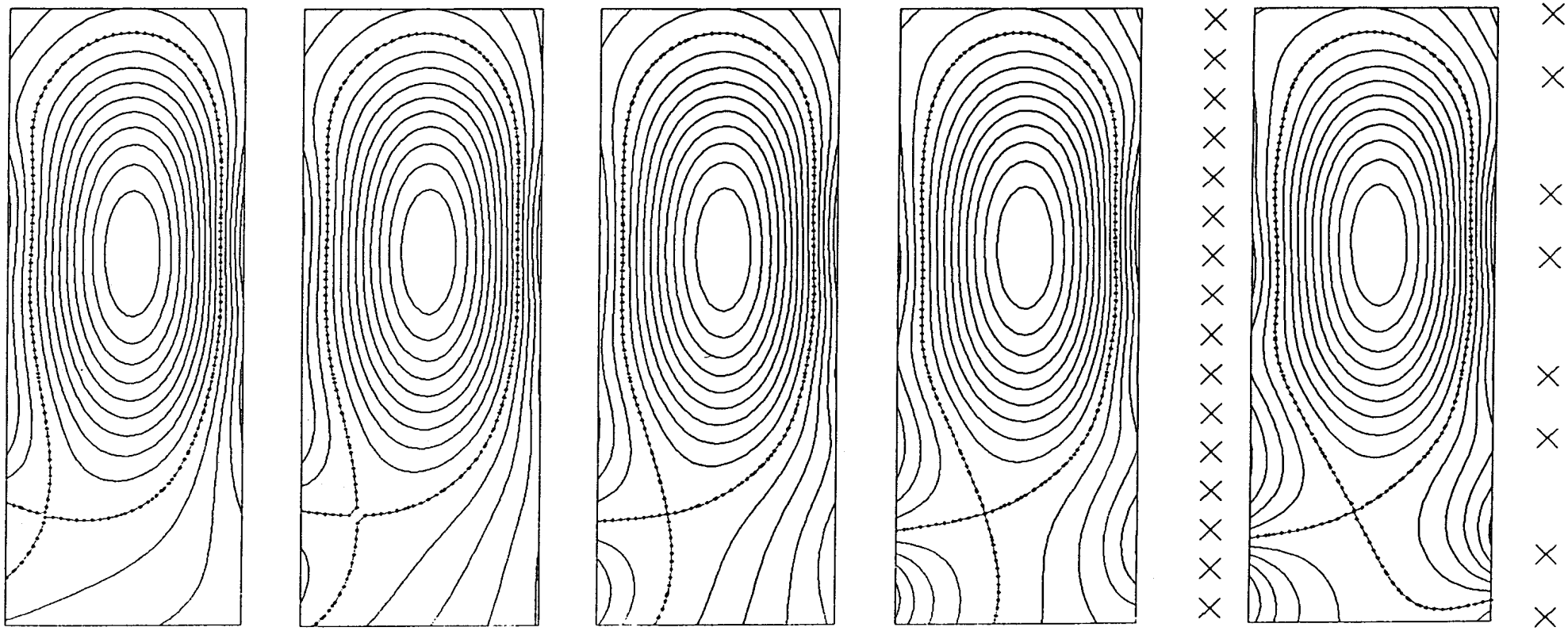

FIG.14

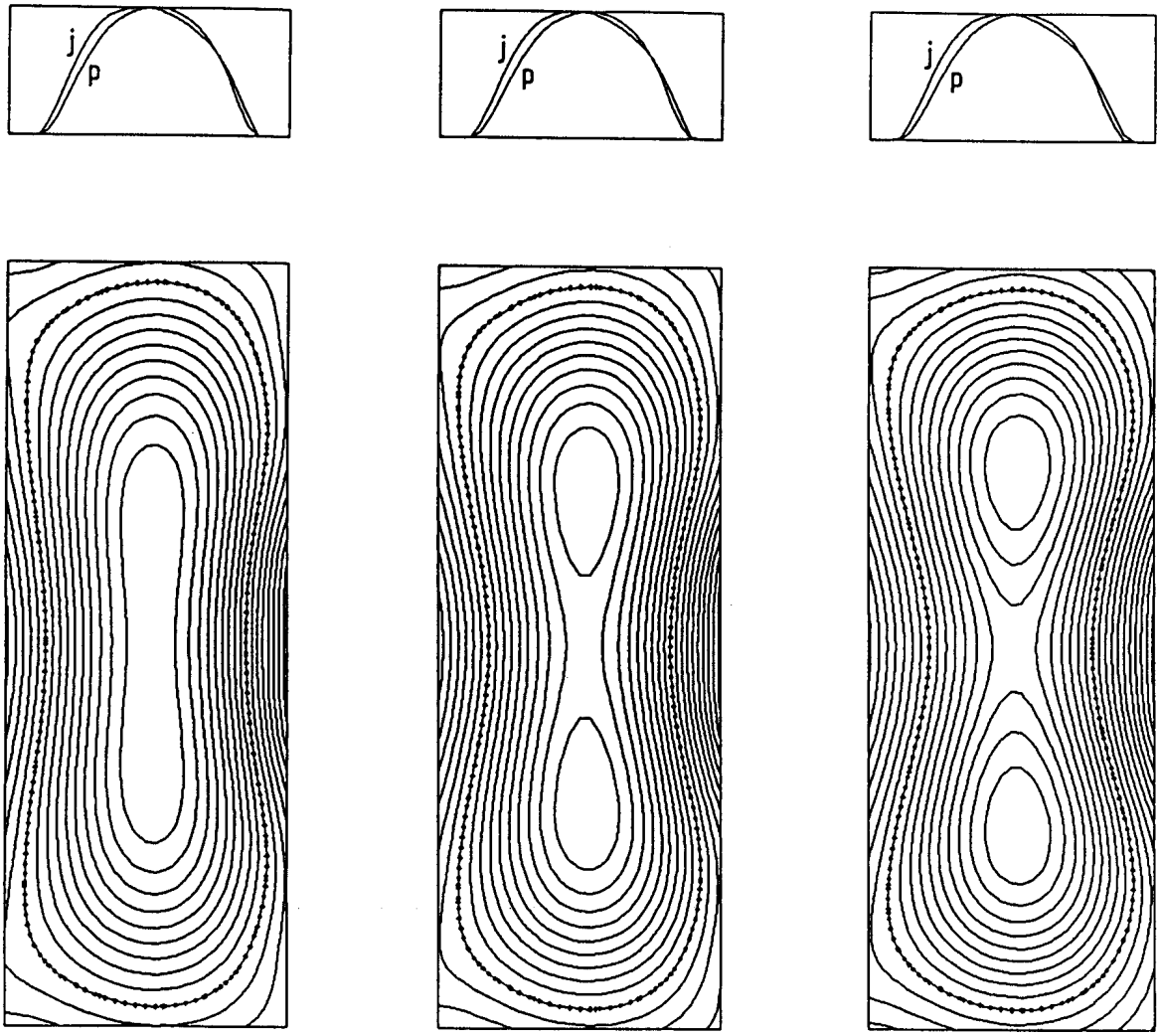

FIG.15