JOURNAL OF COMPUTATIONAL PHYSICS 73, 125–130 (1987)

Orthogonal Grid Construction for Modeling of Transport in Tokamaks

M. PETRAVIC

Plasma Physics Laboratory, Princeton University, Princeton, New Jersey 08544

Received February 11, 1986

A simple method has been developed for numerically constructing orthogonal grids based on the tokamak poloidal flux surfaces. The poloidal flux surfaces form a natural set of coordinate lines for the study of transport in the tokamak scrape-off region, since the energy transport there is mostly along the field lines contained within the flux surfaces. For a study of both the poloidal and perpendicular (radial) transport, a two-dimensional, preferably orthogonal, mesh is required. The need for a new mesh generating code arose from the requirements of the particular topology produced by the zeros in the poloidal field (x-points) and the consequent problems with the numbering of the mesh. © 1987 Academic Press, Inc.

INTRODUCTION

A number of methods for constructing orthogonal meshes exist in the literature [1], but the available codes required a considerable amount of adaptation to our problem of constructing a correctly numbered mesh which encompasses the regions both inside and outside the separatrix (Fig. 1). In order to map simply the topology of Fig. 1 onto a two-dimensional FORTRAN array, we make a cut along the dashed line in the figure. In this way, we end up with a single computational region with four new artificial boundaries. In order to have the physics properly connected, it is now necessary to ensure that the physical variables be continuous across these new boundaries. This can usually be accomplished with little difficulty.

Using this approach, the mesh lines coinciding with the poloidal flux surfaces are numbered from left to right as indicated in Fig. 1. The cuts in the mesh are used not only to make the addressing of the mesh variables simple, but it actually makes it easier to apply the sheath boundary conditions commonly used in the scrape-off modeling [2].

THE MESH GENERATION

The first set of coordinate lines. The starting point for mesh generation is a set of poloidal flux values on a rectangular mesh, usually provided by an equilibrium

0021-9991/87 $3.00

581/73/1-9

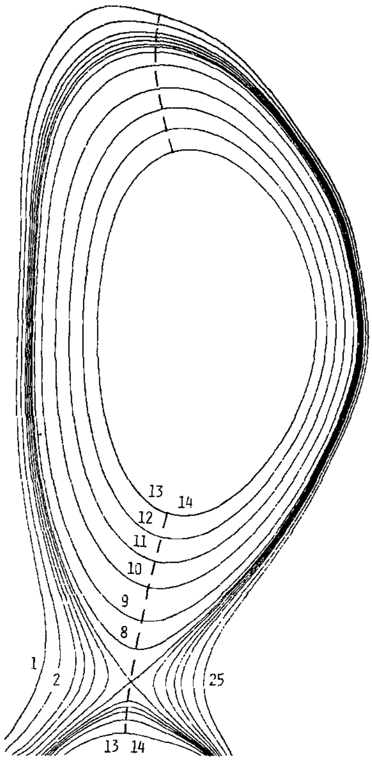

FIG. 1. The poloidal magnetic flux contours, in a poloidal plane, for a tokamak with a single null divertor. The dashed line shows how the area has been cut into two regions.

solver. The next step consists of numerically determining contours of constant flux. For this purpose the rectangular mesh is subdivided into triangles with a set of diagonal lines going through every point on the mesh. The contours are then obtained by interpolation between the corners of the triangles. An important practical part of the procedure is that the mesh is separated into two regions by cutting and rejoining the flux contours as shown in Fig. 2. The line numbering also occurs here. A part of a contour which lies inside the separatrix is joined to a section of another contour outside the separatrix, i.e., contours 8 to 13. The straight-line connection between the two contours is later ignored by the mesh generator.

The contour program defines a contour by means of an ordered discrete set of points lying on that contour. Next, these points are spline-fitted in a standard way; cubic splines are used. A small but practically important point is that in order to have single-valued functions, the contours have to be subdivided into segments, and each segment fitted separately in a different Cartesian coordinate system. Four systems are used in all with a common origin but rotated by 90° with respect to its neighboring systems. The spline fitting completes the definition.

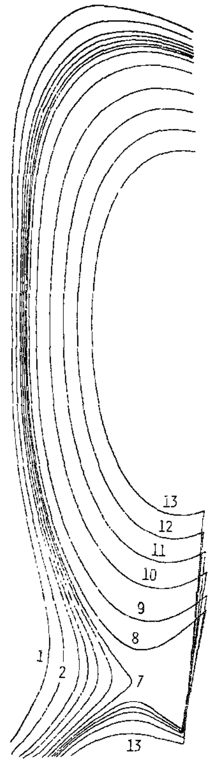

FIG. 2. The first family of coordinate curves in the first region. Segments of the curves inside the separatrix have been joined to the segments at the bottom of the picture to simplify the labeling, which is also indicated.

The second set of coordinate lines. The second set, orthogonal to the first, is then constructed using a piecewise continuous set of circle segments which join at the points of intersection with the splines. The method of finding a circle segment orthogonal to two adjoining splines is illustrated in Fig. 3. An auxiliary coordinate system is placed with its origin at the already determined point, (u_0, v_0), of the mesh, and with its x-axis oriented along the tangent to the spline $\Sigma_0(u)$. The spline $\Sigma(u)$ is defined in the system u, v. The same spline $\Sigma(u)$ when viewed from the x, y system will be named $S(x)$.

Next, we have to choose that circle, from all the circles orthogonal to Σ_0 at $x=0$ and $y=0$, which intersects $S(x)$ at the right angle. The requirement can be expressed through the three equations

$$y^2+(x-R)^2=R^2, \tag{1}$$

$$y'=\frac{1}{S'}, \tag{2}$$

$$y=S, \tag{3}$$

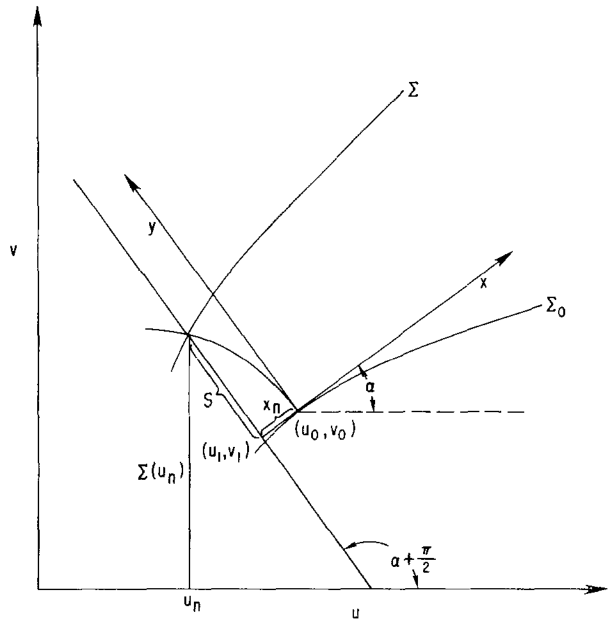

FIG. 3. The coordinate systems used in the construction of orthogonal trajectories, which are here arcs of circles. The curves labeled Σ belong to the first family of coordinates. A circle is drawn from the origin of the moving system n_0, v_0 to the curve Σ.

where (1) is the equation for the family of circles orthogonal to the x-axis at $x=0$, (2) is the orthogonal condition for the circle $y(x)$, and the spline $S(x)$ which must be satisfied at $y(x)=S(x)$ is given by (3).

Differentiating (1), we find

$$yy' = R - x, \tag{4}$$

and upon eliminating R by combining (1) and (4), we finally arrive at the condition

$$(S^2 - x^2)\, S' + 2Sx = 0, \tag{5}$$

where we have also substituted S for y. Equation (5) can now be solved iteratively using Newton's method

$$x_{n+1} = x_n - \frac{2S(x_n)\, x_n + [S^2(x_n) - x_n^2]\, S'(x_n)}{2S(x_n)\{1 + [S'(x_n)]^2\} + [S^2(x_n) - x_n^2]\, S''(x_n)}, \tag{6}$$

where S, S' and S'' are evaluated at $x = x_n$. The only problem is that S is not defined in the x, y system; it is only known as $\Sigma(u)$ in the u, v system. Therefore, in order to find $S(x_n)$, we again use Newton's rule to find the intersection of a straight line through (u_1, v_1), and parallel to the y-axis, with $\Sigma(u)$ in the u, v system. This straight line (Fig. 3) is given by

$$v - v_1 = -\frac{1}{\tan \alpha}\,(u - u_1), \tag{7}$$

where $u_1 = u_0 + x_n \cos \alpha$ and $v_1 = v_0 + x_n \sin \alpha$. Having found the value of $\Sigma(u_n)$ at the point of intersection, $S(x_n)$ can be easily obtained by transforming $[u_n, \Sigma(u_n)]$ into the x, y system.

This simple method has proven very effective for constructing useful orthogonal meshes in the diverted tokamak magnetic geometry. The resulting system is, of course, not orthogonal at the x-point, but this presents no problem since the problem variables are not evaluated at the corners of the computational volumes. A mesh constructed in this way for the purpose of modeling the D–III–D experiment at GA Technologies (San Diego, CA) is shown in Fig. 4. For clarity, a mesh with a reduced number of mesh points is shown. The actual mesh is finer, and is also trimmed to conform to the walls which, in general, cut across the mesh at an angle. However, the greater part of the vacuum wall conforms closely to the flux surfaces, or the outermost contour. It should be obvious that for our problem it was not possible to also use the wall as one coordinate of the system.

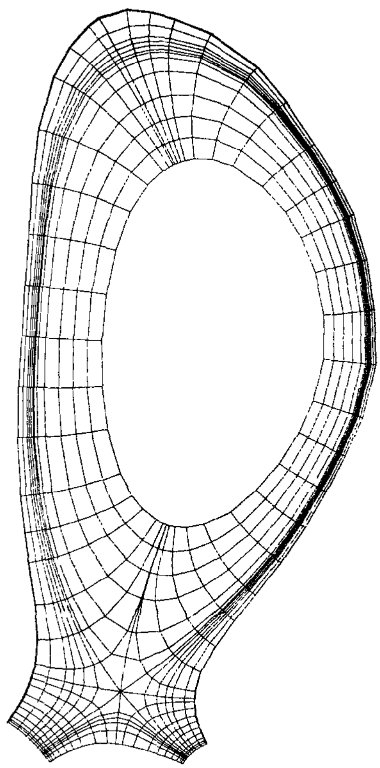

FIG. 4. A subset of the orthogonal system constructed for the simulation of the D–III–D experiment. A much finer work tailored to fit the vacuum walls, which is actually used in the simulation, could not be shown because of the lack of resolution.

SUMMARY

Our method falls in the category of orthogonal trajectory methods [3]. These methods in general seek to find pairs of points on two adjoining coordinate curves which are in orthogonal correspondence. With such an approach there is no unique answer and assumptions have to be made about the missing continuum of curves [4] in order to make the solution unique. We use instead what appears to be a more straightforward method which yields a unique answer without additional conditions and uses circle segments with a continuous first derivative to construct the second family of coordinate curves. The implementation of the method is particularly simple if a Cartesian coordinate system can be found in which the first family of coordinate curves can be represented by single-valued functions. Though this method was derived for a particular purpose, it should prove useful in a more general range of applications.

ACKNOWLEDGMENTS

I thank D. Heifetz and S. Lieberson–Heifetz for a number of helpful suggestions. This work was supported by the U.S. Department of Energy under Contract DE-AC02-76-CHO-3073.

REFERENCES

1. J. F. THOMPSON (Ed.), *Numerical Grid Generation* (North–Holland, Amsterdam, 1982).
2. M. PETRAVIC *et al.*, *J. Nucl. Mater.* **128/129**, 91 (1984).
3. P. R. EISEMAN, in *Numerical Grid Generation*, edited by J. F. Thompson (North–Holland, Amsterdam, 1982).
4. D. E. POTTER AND G. H. TUTTLE, *J. Comput. Phys.* **13**, 483 (1973).