*33rd EPS Conference on Plasma Phys. Rome, 19 - 23 June 2006 ECA Vol.**30I**, P-5.165 (2006)*

Plasma transport modelling with multiple-mapping magnetic surfaces

I.S. Landman, G. Janeschitz

Forschungszentrum Karlsruhe, IHM, FUSION, P.O. Box 3640, 76021 Karlsruhe, Germany

Abstract.

A simulation approach with arbitrary behaviour of magnetic flux coordinates uniformly covering the whole vessel is described that was developed for the two-dimensional plasma equilibrium code TOKES aiming at tokamak modelling. The connections between neighbour regions of monotonic flux variation are described by a graph structure that is automatically generated in the code. It is explained how plasma diffusion throughout the graph can be calculated, across closed field lines as well as such ending at the wall.

The plasma confinement in ITER near the edge will probably be far from a stationary process, being governed substantially by the edge localized modes. Fast plasma transport may drastically expand the width of the scrape-off layer (SOL) up to the walls. Lost plasma can propagate as filaments in which the currents are localized causing the poloidal magnetic flux to have several local minimums and maximums in the vessel.

For computer modelling of the complex plasma behaviour in the whole vessel, including the core as well as the SOL, a new code TOKES is being developed. Some features of the code were described elsewhere, such as implementation of plasma-wall interactions [1] and simulation of carbon impurity penetration into the core [2].

This work describes the models of TOKES for the multi-mapping magnetic frames

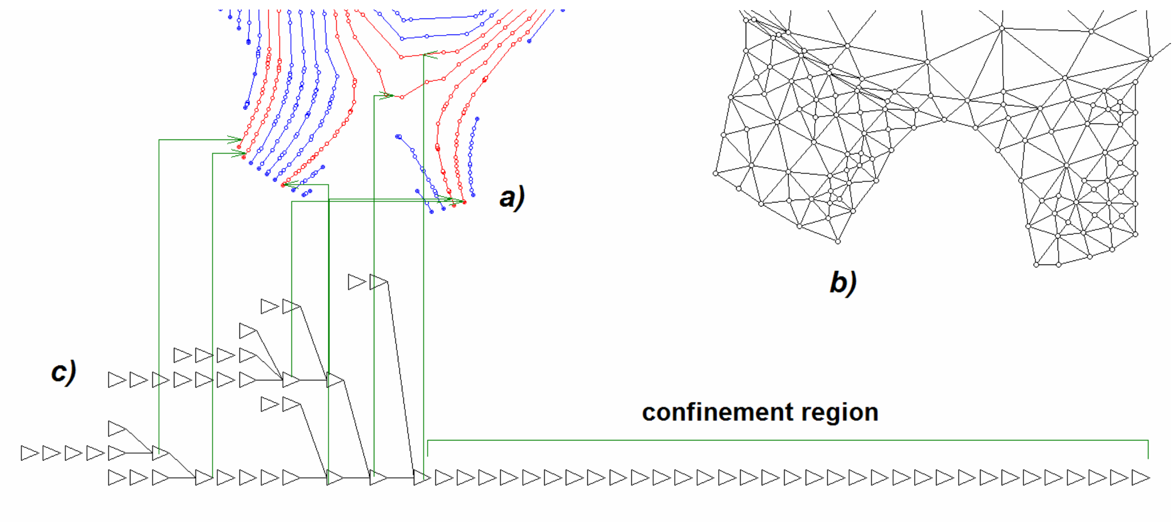

Fig. 1 A fragment of ITER vessel is shown, with the magnetic flux contours (a), the respective graph (c) and a fragment of the background triangular meshes (b)

and the plasma diffusion across them. The code calculates at each time step the plasma currents and the coil currents assuming toroidal symmetry about the axis z. Then, new magnetic flux contours representing magnetic layers that contain plasma are generated (Fig 1a), and the pressure profile resulting from plasma transport and beam fuelling during the time step is updated. Special triangular meshes were produced that cover the vessel's poloidal cross-section (see Fig. 1b). The magnetic layers, approximated in the poloidal plane with chains of line segments, are attached to the triangles' sides.

The poloidal magnetic flux $w(r,z)$ is a continuous function of the polar coordinates r and z calculated at the triangles' corners and linearly approximated inside the triangles. Going away from the magnetic axis of the core, w changes monotonically up to the separatrix, outside of which the contours are unclosed. There the poloidal field coils produce several regions of monotonic decrease or increase of $w(r,z)$ near the wall surfaces.

Fig. 1c shows a graph structure representing the connections between the neighbour regions of monotonic $w(r,z)$. In the graph, one (turned) symbol Δ stands for one magnetic contour line. The code automatically builds up such a graph for an arbitrary function $w(r,z)$, with one original layer Δ being followed by chains of its descendants ("dads" and "sons" in TOKES terminology). Each dad but the last one in a chain has only one son, and the last Δ either has several sons or none. In such a graph structure, each separate region of monotonic behaviour of w is represented with one Δ-chain, for instance the whole confinement region.

Each segment of a magnetic layer has some surrounding area s in the poloidal plane, which is part of the area S of its host triangle. Area s is proportional to the length of the triangle's segments $\mathbf{l}_i$: $s = s_i = k|\mathbf{l}_i|$, with coefficient $k = S/\sum_i|\mathbf{l}_i|$ providing the correct metric. However, not all triangles in Fig. 1b are crossed by the magnetic layers in Fig. 1a. Such triangles are presently ignored in the computations, which restricts the approach so far.

Fig. 2 Rectangular meshes of magnetic layers in TOKES

The segment $\mathbf{l}_i$ is interpreted as a rectangle oriented along the segment, with some sizes a_i and b_i ($a_ib_i = s_i$). If $|\mathbf{l}_i|^2 > s_i$, the side of a_i is assumed to be equal to $|\mathbf{l}_i|$, otherwise $a_i = b_i = \sqrt{s_i}$ is assumed. The layers become split into such rectangles (see Fig. 2).

The confined plasma of the core or the filaments diffuses into periphery regions, however only via those rectangles. At each layer the densities n and temperatures T of ions and electrons are assumed constant over the layer's rectangles (which for unclosed layers is a rather rough but acceptable estimation). The code calculates also the interface areas between magnetic layers necessary for considering cross fluxes. The approximation suggested here, with rectangles as numerical plasma cells, provides convenient finite-difference schemes for TOKES for involved physical processes.

The feature that the magnetic contour graph allows uniform algorithms for plasma transport across arbitrary magnetic flux frame is demonstrated using as an example the diffusion process. With a coordinate x that acquires sequential or the same integer values at the interfaces between numerical magnetic layers, the diffusion equation for $n(t,x)$ reads

$$V\frac{\partial n}{\partial t}=\frac{\partial}{\partial x}\left(\mathrm{K}\frac{\partial n}{\partial x}\right)-V\frac{n}{\tau_0} \tag{1}$$

Here $V(x)$ is the volume enclosed between x–½ and x+½, $\mathrm{K}(x)$ the magnetic surface integral of the diffusion coefficient divided by the local distance between x–½ and x+½, and $\tau_0(x)$ the longitudinal loss time of plasma ($\tau_0 = \infty$ for closed surfaces). The code calculates V and τ_0 for each layer, and K for each layer's interface.

The corresponding finite-difference scheme for a layer reads

$$V\frac{n-\hat{n}}{\tau}=\mathrm{K}_d\left(n_d-n\right)+\sum_s \mathrm{K}_s\left(n_s-n\right)-V\frac{n}{\tau_0} \tag{2}$$

The $\hat{n}$ and n correspond to the beginning and the end of the time step τ, respectively. The integrals K_d and K_s describe the connections of the layer with its dad and the sons if they are available. Eq.(2) is rewritten in terms of convenient coefficients A, B and S as

$$\left(B+A_d+\sum_s A_s\right)n-A_d n_d-\sum_s A_s n_s=S \tag{3}$$

$$B=V\left(1+\frac{\tau}{\tau_0}\right),\qquad A_d=\tau\mathrm{K}_d,\qquad A_s=\tau\mathrm{K}_s,\qquad S=V\hat{n} \tag{4}$$

The solution to Eq.(3) is obtained as follows. Initially a childless layer ("end layer") is considered (anyone at the left ends of the graph Fig. 1c). For it Eq.(3) takes the simple form:

$$\left(B+A_d\right)n-A_d n_d=S \tag{5}$$

The n of the end layer is thus expressed in terms of its dad's plasma density n_d:

$$n = \frac{S + A_d n_d}{B + A_d} \tag{6}$$

In Eq.(3) for the layer which is the dad of the end layer, the end layer is one of sons and thus its n is one of n_s in the sum. We eliminate that n_s using Eq.(6) with the corresponding re-designations: $n \to n_s$, $B \to B_s$, $A_d \to A_s$, $n_d \to n$, $S \to S_s$. Eq.(3) for the dad takes the form:

$$\left(B + \tilde{A}_d + \sum_{s' \neq s} A_{s'} \right) n - A_d n_d - \sum_{s' \neq s} A_{s'} n_{s'} = \tilde{S} \tag{7}$$

The modified coefficients are given by

$$\tilde{A}_d = A_d + A_s \left(1 - \frac{A_s}{B_s + A_s} \right), \qquad \tilde{S} = S + A_s \frac{S_s}{B_s + A_s} \tag{8}$$

If the sums in Eq.(7) are empty (the dad has only one son) the end layer's function n is eliminated and the dad becomes playing the role of the end layer, however with the modified A_d and S. Thus one step of calculations for a layer chain is done.

The code repeats those steps until reaching a layer with several sons. As for it the sums on s' in Eq.(7) are not empty, the same calculations are made for the other sons. At each step the equations of type Eq.(6) expressing the sons' n in terms of its dads' n are obtained. Finally the sums of Eq.(7) are replaced by the corresponding modified coefficients and thus the layer which has several sons transforms into the end layer. The procedure is then repeated further until reaching the original layer. Eq.(7) for the original layer, with the sums replaced, reads

$$\left(B + \tilde{A}_d \right) n = \tilde{S} \;\Rightarrow \qquad n = \frac{\tilde{S}}{B + \tilde{A}_d} \tag{9}$$

Having obtained the last value of n, the process is reversed in order to obtain all previous n. It is done using the available expressions Eq.(6) for sons' n in terms of their dads' n.

The described approach and the diffusion algorithm were successfully tested. Similar algorithms for the thermal conductivity and convective contributions to the transport equations have also been implemented in TOKES.

References: 1. I.S. Landman, G. Janeschitz, "Numerical simulations of plasma wall interactions for ITER", 17th Intern. Conf. on Plasma Surface Interactions, Hefei Anhui, China, May 22-26, 2006. **2**. I.S. Landman, G. Janeschitz, "Contamination and radiation losses in post-ELM tokamak plasma", 32nd EPS Conf. on Plasma Physics, Tarragona, Spain, June 27 – July 1, P-1.003 (2005)