JOURNAL OF COMPUTATIONAL PHYSICS **138,** 881–906 (1997)
ARTICLE NO. CP975848

Advanced Finite Element Modeling of the Tokamak Plasma Edge

R. Zanino

Dipartimento di Energetica, Politecnico, 10129 Torino, Italy and Max-Planck-Institut für Plasmaphysik, 85748 Garching, Germany
E-mail: zanino@alphaz.polito.it

Received October 30, 1996

A finite element fluid model of the two-dimensional axisymmetric plasma edge region in a tokamak is presented. A pure plasma with different electron and ion temperatures is considered, where its evolution is driven by sources. The sources are due to the interaction between plasma and neutrals recycling at the walls, which are described by a Monte-Carlo code. A realistic curvilinear (poloidal divertor) geometry is treated and can be discretized both with an unstructured and with a structured flux-surface-fitted mesh generator. The convergence of the code is demonstrated numerically and the results are compared with those of a reference finite volume (conservative) code. © 1997 Academic Press

Key Words: finite elements; plasma; tokamak; scrape-off layer; fluid model.

1. MOTIVATION

Over the last decade the edge plasma region has risen from (not only topographically) marginal to a rather central role. At present it is clear that, together with the problem of confinement, plasma–wall interactions present the major constraints toward a magnetic fusion reactor: heat loads on plasma facing components must be reduced to technologically acceptable levels, impurities coming from those components must be kept to a sufficiently low concentration in the plasma to avoid excessive cooling and maintain ignited conditions, etc. [1]. Therefore, in parallel with new experimental diagnostics of the plasma edge, more and more sophisticated tools for the computational modeling of this region are being developed.

In a tokamak, the main plasma region is essentially one-dimensional, because fast equipartition along magnetic field lines is allowed by the existence of closed magnetic surfaces. On the contrary, the edge plasma region and, in particular, the

0021-9991/97 $25.00

scrape-off layer, i.e. the region with magnetic surfaces intersecting the plasma facing components, is intrinsically (at least) two-dimensional. In fact, there is no principal reason for having, e.g., the same plasma temperature at points on the same magnetic surface, if they are at different distances from, e.g., the divertor target. The modeling of the plasma edge is therefore much more complex than the modeling of the main plasma.

From the point of view of the physics, due to the complexity and cost of a fully kinetic description of the plasma, fluid models [2] are often privileged notwithstanding their limited applicability [3], and the more general kinetic approach is mainly used to derive suitable boundary conditions [4] and (long mean free path) corrections to the constitutive relations.

From the point of view of the computational approach, finite volume codes have been developed first [5] and have now reached a high degree of sophistication in the physics they can treat [6]. During the last years, however, the problem of a more detailed and accurate treatment of several geometrically complicated features of the edge, e.g., solid targets extremely inclined to the poloidal magnetic field, X-point region, etc., has led to considering finite elements [7–10] as an alternative approach, able in principle to deal more naturally with the mentioned issues (let us stress at once, however, that no existing finite element code contains at present so much physics as the major finite volume codes). Very recently, then, and to the same purpose, some finite volume codes [11, 12] have been upgraded to 9-point computational stencils in order to allow their use on nonorthogonal grids.

The first attempt with finite elements—the FELS code—worked originally in a model rectangular geometry [7] with only a simplified treatment of the neutral particles [8], and similar limitations were also present in subsequent independent work by other authors [10]. In those papers the computational emphasis was on the stabilization of the finite element scheme by means of a suitable mesh choice and/or a Petrov–Galerkin formulation [13]; however, both the essential physical aspects of a realistic curvilinear domain and of an accurate model for the neutrals, and the computational issues related to the energy conservation properties of the scheme in realistic geometries, were not addressed; the adjective ''advanced'' used in the title of the present paper refers to the inclusion of these aspects.

Curvilinear geometries both of the toroidal limiter type [14] (with vanishing poloidal extension) and poloidal divertor type [15] were later treated with FELS using as discretization of the spatial domain an unstructured mesh of triangles stretched in the direction of the poloidal magnetic field.

However, it was subsequently realized [16] that the best alignment to the poloidal field (see below), attainable with a (fixed) unstructured mesh, could still be insufficient to get a good global energy conservation; therefore, FELS was coupled to a structured flux-surface-fitted mesh generator, showing a good comparison (in the absence of sources acting on the plasma) with the reference finite volume code in the actual poloidal divertor geometry of the ASDEX Upgrade tokamak. (Good alignment has also been obtained with a novel unstructured mesh generator [17], which is currently being used in conjunction with a newly developed finite element code [18].) More recently, also the case of a model toroidal limiter (with finite poloidal extension tangent to the plasma) was treated with FELS [19].

Here the results of [16] will be extended to the case with either externally imposed or self-consistent Monte-Carlo sources, demonstrating numerically the convergence of the FELS code in both cases, and a detailed comparison with the finite volume approach to plasma edge modeling will be presented. The paper is organized as follows: in Section 2 and 3 respectively the plasma and neutral models are briefly reviewed. The geometries of interest and the mesh generators available to FELS are discussed in Section 4. Results are presented in Section 5.

2. PLASMA MODEL

The plasma in the edge is modeled as the mixture of two coupled fluids—electrons, subscript "e," and a single, singly charged ion species, subscript "i"—in a given magnetic field **B**; since the typical lengths are much larger than the Debye scale, the mixture can be considered to be quasineutral (i.e., for the particle densities, $n_e = n_i = n$); different temperatures $T_e \neq T_i$, but the same flow velocity $\mathbf{V}_e = \mathbf{V}_i = \mathbf{V}$ for both components are assumed.

The latter assumption, i.e. the absence of electric currents, is made for the sake of simplicity and is rather customary in edge plasma transport studies (see, e.g., [5, 10]). A second important simplifying assumption in the model concerns the number of nonvanishing components of the fluxes considered here; again similarly to what was done in [5, 10], we include the parallel (to **B**) component and treat the component perpendicular to the magnetic surfaces (radial component) with a diffusive ansatz, but we neglect the so-called "drifts," i.e., the (diamagnetic) component of the fluxes perpendicular to **B** *on* the magnetic surfaces. It is still not clear whether electric currents and drifts are essential in scrape-off layer modeling; however, they have recently been included in some models specifically devoted to the study of their effects [20].

Two coordinate systems will be of use here: (a) a cylindrical global one, (R, φ, Z), with R the distance from, φ the angle around, and Z the distance along, the symmetry axis; (b) a local one, defined by the unit vectors e_φ in the toroidal direction, i.e. along φ, e_θ in the poloidal direction, i.e. along the projection of **B** on the R, Z plane, and $\mathrm{e}_\psi = \mathrm{e}_\theta \times \mathrm{e}_\varphi$ in the radial direction. Axisymmetry (also called toroidal symmetry), i.e. $\partial a/\partial\varphi = 0$ for any scalar a, is assumed throughout, thus reducing the problem to 2D. Although some 3D effects in the edge plasma of a tokamak can exist, because of, e.g., magnetic and/or first wall asymmetries, none of the present edge plasma codes is three-dimensional.

Transport coefficients in the parallel direction are taken as classical, and the corresponding constitutive relations are simplified versions [7] of those given in [2]. On the contrary, transport coefficients in the perpendicular (φ) direction are anomalous and can be prescribed as arbitrary functions by the user. In general, because of the anisotropy introduced by **B**, parallel transport coefficients can be several orders of magnitude larger than the corresponding perpendicular ones [21]; this fact leads to a scrape-off layer which is radially very thin—$O(10^{-2}$ m)—compared to its parallel length—$O(10^2$ m) and will also have an important influence on the numerics in the following.

2.1. *Weak Form of the Plasma Equations*

The set of plasma model equations [2, 8] is brought into so-called weak (or weighted residual) form by multiplying each equation with an arbitrary test function ϕ of space and integrating over the whole (3D) spatial domain Ω. As usual, multidimensional integration by parts (Green's formula) is then used, leading to surface integral contributions from the boundary $\partial\Omega$ of Ω and reducing the minimal required degree of the polynomials to be used later in the finite element formulation.

The resulting set of equations is given by:

the continuity equation,

$$(\rho_t \phi) = -\langle \mathbf{\Gamma} \cdot \mathbf{n} \phi \rangle_{\text{in}} - \langle \mathbf{\Gamma} \cdot \mathbf{n} \phi \rangle_{\text{out}} + (\mathbf{\Gamma} \cdot \nabla \phi) + (S^m \phi), \tag{1}$$

where $\rho \equiv m_i n$ is the mass density, the subscript "t" indicates partial derivation in time $\partial/\partial t$, $\mathbf{\Gamma} \equiv \rho \mathbf{V}$ is the mass flux, S^m is the mass source; $(\beta) \equiv \int_\Omega \beta \, d\Omega$, and $\langle \gamma \rangle$ is the surface integral of γ on $\partial\Omega$ (the subscripts "in" and "out" refer to the subsets of $\partial\Omega$, where $\mathbf{\Gamma} \cdot \mathbf{n} < 0$ or >0, respectively, and $\mathbf{n}$ is the outward-going unit normal vector to $\partial\Omega$);

the total parallel momentum balance,

$$(\mathbf{b} \cdot \mathbf{\Gamma}_t \phi) = -\langle \mathbf{b} \cdot \mathscr{T} \cdot \mathbf{n} \phi \rangle + (\mathbf{b} \cdot \mathscr{T} \cdot \nabla \phi) - (\nabla \mathbf{b} : \mathscr{T} \phi) + (\mathbf{b} \cdot \mathbf{S}^P \phi), \tag{2}$$

where $\mathbf{b} \equiv \mathbf{B}/|\mathbf{B}|$, $\mathscr{T} \equiv \mathbf{\Gamma}\mathbf{\Gamma}/\rho + (p_e + p_i)\mathbf{I} + \mathbf{\Pi}_i$ is the momentum flux tensor ($p_e = nT_e$ and $p_i = nT_i$ are the pressures, $\mathbf{I}$ is the unit dyad, and $\mathbf{\Pi}_i$ is the ion viscous stress tensor, the electron viscosity being neglected here as usual), and $\mathbf{S}^P$ is the total momentum source;

a radial diffusion ansatz,

$$(\Gamma_\psi \phi) = -\left(D \frac{\partial \rho}{\partial \psi} \phi \right), \tag{3}$$

where D is the (anomalous) particle diffusion coefficient across magnetic surfaces;

the electron internal energy balance,

$$(\tfrac{3}{2}[p_e]_t \phi) = -\langle \mathbf{\Gamma}_e^E \cdot \mathbf{n} \phi \rangle + (\mathbf{\Gamma}_e^E \cdot \nabla \phi) + (\{\mathbf{\Gamma} \cdot [\nabla p_e - \mathbf{S}_e^P]/\rho + Q_e + S_e^E\} \phi), \tag{4}$$

where $\mathbf{\Gamma}_e^E \equiv \frac{5}{2} p_e \mathbf{\Gamma}/\rho + \mathbf{q}_e$ is the electron energy flux, $\mathbf{q}_e$ is the electron conductive heat flux, $\mathbf{S}_e^P$ is the electron momentum source, Q_e is the energy interchange due to collisions between electrons and ions, and S_e^E is the electron energy source. Notice that the $\mathbf{\Gamma} \cdot [\nabla p_e - \mathbf{S}_e^P]/\rho$ contribution to the last term on the right-hand side represents the ohmic heating and comes from the elimination of the electric field $\mathbf{E}$ by using the electron momentum balance (while this strategy is generally accepted for the parallel component of $\mathbf{E}$, it is used here also for the radial component, only for the sake of keeping the model self-consistent and as simple as possible—refer to [22] for a more accurate treatment of the electric field in the edge);

the total energy balance,

$$(\mathscr{E}_t\phi) = -\langle \mathbf{\Gamma}^E \cdot \mathbf{n}\phi\rangle + (\mathbf{\Gamma}^E \cdot \nabla\phi) + (S^E\phi), \tag{5}$$

where $\mathscr{E} \equiv \frac{3}{2}(p_e + p_i) + \frac{1}{2}\Gamma^2/\rho$ is the total energy density, $\mathbf{\Gamma}^E \equiv \mathbf{\Gamma}_e^E + \frac{5}{2}\mathbf{\Gamma}p_i/\rho + \mathbf{q}_i + \mathbf{\Pi}_i \cdot \mathbf{\Gamma}/\rho$ is the total energy flux (including ion energy convection, conduction, and viscous dissipation), and S^E is the total energy source.

A few comments are due as to the choice of equations solved in the FELS code: (1), (2), and (4) are present in most of the existing codes, whereas instead of (5) the *ion* energy balance is often used (e.g., in [5, 10]); in B2 the total energy balance is then relaxed to ensure energy conservation. Notice that (3) is kept as a separate equation (as done in [5], but opposite to what done in [10]) in view of eventually implementing a more complex form of radial transport (e.g., including inward pinch, drifts, etc.); notice also that (4) cannot be written in conservative form.

As to the choice of dependent variables, we can observe that primitive variables (ρ, velocity components, and temperatures) have often been used in the literature (e.g., [5, 10]). Here we decided to use conservative variables (as far as possible; see above). This simplifies the discretization of time derivatives and flux boundary conditions and is obviously the most natural choice corresponding to equations in conservative form. One possible disadvantage of our choice of dependent variables can be related to the following, a priori not obvious, fact: in FELS T_e is a derived variable ($\propto p_e/\rho$); since the (huge) parallel electron heat conductivity is $\propto T_e^{5/2}$ [2], linearization (see Section 2.4) of the heat conduction terms in (4) and (5) leads to coefficients $\propto \rho^{-9/2}$; in (otherwise uninteresting) regions of very low density, these coefficients can become very large and compromise the accuracy of the linearization. Although it was not possible to prove it rigorously, we postulate that this is the major reason for occasionally having $T_e \to 0$ in the numerical solution at nodes where the density is extremely low (e.g., $O(10^{17})$ m^{-3}).

2.2. *Plasma Boundary Conditions*

The problem of the boundary conditions to be imposed on the plasma described by the set (1)–(5) is not yet a completely settled question from the point of view of the physics.

Difficulties arise in particular with respect to two issues: (1) ion momentum and energy transport to solid boundaries is complicated by the fact that the ion distribution function at the Debye sheath in front of a solid wall can be far from Maxwellian because of the accelerating nature of the sheath for the ions, as opposed to the electrons [4]; (2) in the practically relevant case of glancing (up to tangent!) incidence of the poloidal field lines to solid targets, the sheath physics can change completely (see, e.g., [23]) from the textbook situation of perpendicular incidence and, correspondingly, affect the correct boundary conditions for the fluid plasma. Additional difficulties can arise, as we shall see in the following sections, if artificial boundaries have to be introduced for the purpose of, e.g., reducing the number of nodes requested for a particular simulation.

From the computational point of view, two types of boundary conditions can be automatically implemented in the FELS code:

(1) Dirichlet-like boundary conditions. Here a combination of the components of the vector of unknowns

$$\mathbf{u} \equiv [\rho, \Gamma_{\parallel}, \Gamma_{\psi}, \mathrm{p_e}, \mathscr{E}] \tag{6}$$

is equated to some space and time-dependent function on $\partial\Omega_D \subseteq \partial\Omega$ ($\Gamma_{\parallel} \equiv \mathbf{b}\cdot\mathbf{\Gamma}$);

(2) Robin-like boundary conditions. Here a component of the vector of outward fluxes,

$$\mathrm{g_n} \equiv [\mathbf{\Gamma}\cdot\mathbf{n}, \mathbf{b}\cdot\mathscr{T}\cdot\mathbf{n}, -, \Gamma_{\mathrm{e}}^{\mathrm{E}}\cdot\mathbf{n}, \mathbf{\Gamma}^{\mathrm{E}}\cdot\mathbf{n}], \tag{7}$$

is equated to some combination of the components of $\mathbf{u}$ on $\partial\Omega_R = \partial\Omega - \partial\Omega_D$.

Since (1) is hyperbolic in character, at least in the poloidal direction, boundary conditions will have to be imposed only on the "in" portions of $\partial\Omega$, whereas the "out" integrals will be discretized as they are. Since (2), (4), (5) are parabolic, some boundary condition will be needed everywhere on $\partial\Omega$, whereas obviously (3) requires no boundary conditions. The specific boundary conditions used in the present paper will be discussed in Section 5.

2.3. *Space Discretization and Artificial Dissipation Mechanisms*

A poloidal (vertical) cross section of the domain Ω is triangulated by means of either an unstructured mesh generator (see Section 4.1) or a structured one (see Section 4.2).

Linear Galerkin finite elements ($P1$ test and trial functions) are then used to discretize the set (1)–(5) and the corresponding boundary conditions in space on the chosen mesh of triangles. The choice of triangular elements, as opposite to [10], where bilinear quadrilaterals were used, is motivated by the fact that we want to be as flexible as possible in the description of complicated edge geometries. On the other hand (see below), it could turn out that the optimal element choice is probably a combination of quadrangles away from solid boundaries and/or X-points and triangles in those regions. Since this would imply a significant increase of complexity of the code, such a possibility has not been implemented yet into FELS.

The choice of test and trial functions also deserves a comment. On the one hand, being the set (1)–(5) of second order in space, linear test and trial functions are sufficient to obtain a nontrivial discretization, and the Galerkin method (using the same set of test and trial functions) can be shown to be optimal in the case of a linear elliptic problem [24]. On the other hand, the sources in (1)–(5) can be extremely localized near the solid boundaries, and in the same regions the parallel flow is assumed to become sonic [4], so that a situation can arise with steep poloidal gradients in a region where convection dominates over diffusion/conduction (this effect is nonlinearly increased by the $T^{5/2}$ dependence of the parallel diffusivities, together with the significant drop of T near the solid boundaries). Therefore, unless

a sufficiently fine mesh is available to resolve the steep gradients, spatial oscillations will arise and extend even away from the convection-dominated region, making the numerical solution of no use [24].

In FELS we have thus implemented two forms of artificial dissipation, in order to provide some stabilizing mechanism to the pure Galerkin formulation:

(1) modified parallel artificial diffusion [24] in (1). I.e., we add to the physical $\mathbf{\Gamma}$ in (1) the numerical flux $-D_\parallel(\partial\rho/\partial\theta)\mathrm{e}_\theta$; $D_\parallel \propto (\Gamma_\parallel/\rho)h$ varies element by element, and h is the diameter of the circumference circumscribed to the given triangle;

(2) following an idea of Argyris, instead of implementing a full Petrov–Galerkin scheme for (1)–(5), which would become rather cumbersome [10], we weight, with the modified Petrov–Galerkin test function, only the convective portion of the residual in (2), (4), (5). Although not leading to a fully consistent scheme [13], this still gives the desired streamline upwind-like diffusion and has the advantage of being easily amenable, with straightforward algebra, to a simple modification of the parallel diffusivities. In practice, the parallel viscosity $\eta \to \eta[1 + f_1(Re)]$, the parallel electron heat conductivity $\kappa_e \to \kappa_e[1 + f_2(Pe)]$, and the parallel ion heat conductivity $\kappa_i \to \kappa_i[1 + f_3(Re)]$, where Re and Pe are the mesh Reynolds and Paclet numbers, respectively (recall that the classical parallel momentum and ion heat diffusivities are both $O(Re^{-1})$ [2]). The definition of Re and Pe can be heuristically derived from the convection/diffusion ratios in (2) and (4), respectively. Using h as the length scale one gets, element by element,

$$Re \equiv \frac{\mathrm{hnV}_\parallel}{\mathrm{b}_\theta \eta}, \quad Pe \equiv \frac{\mathrm{hnV}_\parallel}{\mathrm{b}_\theta \kappa_\mathrm{e}}. \tag{8}$$

Starting from the 1D recipe for convection–diffusion of a scalar as given in [13] one finally finds that $f_1 = \lambda_1[Re \times \coth(Re) - 1]$, $f_2 = \lambda_2[Pe \times \coth(Pe) - 1]$, and $f_3 = \lambda_3[Re \times \coth(Re) - 1]$. The parameters λ_1, λ_2, λ_3 are introduced here both as an external control on the numerical diffusivities (see below) and in view of the uncertainties in the application of a recipe developed for a simple 1D scalar problem to the complex 2D set (1)–(5).

For both of the previous artificial dissipation mechanisms, we check their effect on a numerical solution in the following way: Once a steady state (see next section) with $D_\parallel$ and λ_1, λ_2, $\lambda_3 \neq 0$ has been obtained, we restart with both $D_\parallel$ and λ_1, λ_2, λ_3 reduced by, say, an order of magnitude, and we repeat this until no significant effect can be seen in the solution. During the actual transient, however, vanishingly small values of $D_\parallel$ and λ_1, λ_2, λ_3 can be insufficient to obtain a converged solution.

In FELS the elemental matrix elements are computed in a local reference frame, with the axes aligned with e_θ and e_ψ at the center of mass of the element. With the exception of the third term on the right-hand side of (2) a constant element by element approximation is used for $\mathbf{b}_\theta$ [18].

2.4. *Time Discretization, Linearization, and Plasma System Solver*

At this point the set (1)–(5) and its boundary conditions have been reduced to a set of nonlinear ordinary differential equations in time; the unknowns are the values of $\mathbf{u}$ at each node of the mesh.

Although modeling of time-dependent phenomena in the edge (e.g., ELMs) has very recently received attention [25], in this paper we shall restrict ourselves to the more traditional study of steady states. Since time-dependency is used here only as a means to solve a complex nonlinear set of equations, stability more than accuracy becomes an issue. In FELS we chose to implement a ϑ-method for the time discretization, which includes fully explicit, Crank–Nicolson, and fully implicit schemes. It must be noticed, however, that the very large parallel electron heat conductivity essentially rules out the possibility of using explicit methods as in [9], because the stability restrictions would become practically unacceptable; here, only the fully implicit or the Crank–Nicolson options will be used.

The resulting set of nonlinear algebraic equations is linearized with a single Newton–Raphson step, using as a guess at each new time step the last-obtained value of **u**; the Jacobian is evaluated analytically (an alternative, numerical evaluation for the same problem was discussed in [40]). Some form of relaxation (roughly in the flavor of linesearch/backtracking techniques) is also applied, so that typically the value of **u** at the new time step is taken as a convex combination of the last computed value and the previously computed one.

In approaching a steady state with an unconditionally stable method, increasingly longer time steps Δt can be taken; furthermore, the nature of the coupling to the neutral particles as implemented in the code (see Sections 3.1 and 5.2) is such that very different time steps can be taken just after or long after a Monte-Carlo call. Therefore, a time adaptive algorithm has been implemented in FELS: depending on the maximum relative variation, $|\Delta u|/u$, of ρ, p_e, and $\mathscr{E}$ on Ω between two successive time steps, the old Δt is either decreased, left as it is, or else increased. We typically increase Δt by a factor 1.1 if $|\Delta u|/u < 3\%$, or we decrease it by a factor 1.5 if $|\Delta u|/u > 5\%$. In the latter case the previous step is rejected and repeated with the new Δt.

The set of linear equations obtained as described above is then solved at each time step by a public domain solver. The most important features of the assembled matrix are that it is sparse (as always with finite elements, because of the localization of test and trial functions) and nonsymmetric (because of convection and nonlinearities). Therefore we have chosen a well-known iterative method, GMRES, with incomplete LU preconditioning, implemented in the routine DSLUGM of the public domain library SLAP (sparse linear algebra package); block diagonal preconditioning was found to be less efficient in most instances. For the cases at hand a relatively large dimension of the Krylov space (from 20 to 40, say) was chosen; smaller dimensions led to very slow or even no convergence of the iterations to the prescribed tolerance (typically slightly above machine precision).

The FELS code is written in FORTRAN77 double precision.

3. MODEL OF THE NEUTRAL PARTICLES

For the description of neutral particles in the plasma edge, models at different levels of sophistication can be found in the literature: semi-analytical simplified ones (see, e.g., [8]), diffusive models (see, e.g., [10]), fully kinetic models (e.g., [26]), Monte-Carlo models (e.g., [27]), and fluid (Navier–Stokes) models [28].

Another way to compare the different models is by observing that not all of these apply to any possible plasma condition; for instance, both Monte-Carlo and kinetic models are typically linear, i.e., neutral–neutral collisions are neglected. Because of the recent interest for very high neutral density regimes, on the other hand, Navier–Stokes models have been developed, incorporating in the macroscopic neutral transport coefficients, also the effect of those collisions (the extension to nonlinear Monte Carlo has been very recently discussed in [29]).

For the time being, FELS is coupled both to a simplified model [8] and to the Monte-Carlo code EIRENE [27]. The simplified model is essentially used only to build initial plasma conditions not too far from the steady state for the FELS-EIRENE coupled runs, so that not too many EIRENE calls are needed. From here on, therefore, we shall restrict our discussion to the Monte-Carlo model.

EIRENE is a fully three-dimensional Monte-Carlo code developed with the aim of investigating neutral gas transport in tokamak plasmas. It solves, for each neutral (atoms, molecules) or trace (molecular ions) particle species of interest, the corresponding linear kinetic equation, by means of statistical methods based on so-called collision estimators or, alternatively, tracklength type estimators.

Essentially all the reaction rates of practical interest and the reflection properties at solid walls are collected in the form of functional dependences on the relevant parameters (e.g., plasma temperature) in two files of suitable form.

The user can define, mostly by means of a single input file, the domain geometry (which does not necessarily coincide with, but more typically includes, the plasma domain Ω), the types of neutral particles, and the corresponding reactions to be considered, the number of particles to be launched, the solid reflection model, etc. A user interface file is also present which does the I/O for the communication between the plasma and the neutral codes (see below) and where the neutral boundary conditions are defined.

3.1. *Plasma–Neutral Coupling Algorithm*

The plasma in the edge is very tightly and nonlinearly coupled to the neutrals. The sources in (1)–(5) come in our case from the plasma–neutral interactions [30], whereas, vice versa, the neutral histories strongly depend on the background plasma conditions. In particular, the higher the plasma density, the tighter the coupling. Furthermore, the sources are nonlocal in nature because their intensity and, to some extent, even their location depend on the outgoing plasma fluxes at recycling surfaces [8]. On the other hand, the evolution of the neutrals is typically much faster than the evolution of the plasma [30], so that it is possible to assume that during the evolution of the plasma the neutrals go through a succession of quasi-steady states.

From the computational point of view it is important to notice that the cost of a Monte-Carlo run is typically much larger than the cost of a plasma time step. Therefore, we call EIRENE only every several plasma time steps (see Section 5.2). After an EIRENE call we decide to call it again when the integrated plasma flux on a recycling surface has changed relatively by more than a given fraction (typically 1 to 5%). This criterion is purely empirical and tries to avoid too large plasma

variations between two EIRENE calls, which would essentially impede the convergence of the coupled system.

During an EIRENE run the execution of FELS is kept frozen, using the LIB$SPAWN runtime library routine of the VMS operating system; i.e., EIRENE is not treated as a subroutine of FELS. At the end of each EIRENE run the execution of FELS is then automatically resumed. The two codes communicate by means of files: EIRENE becomes from FELS one file with the finite element mesh, one file with the plasma density and temperatures at each node in the poloidal cross section of Ω, and another one with density, flow velocity components and temperatures at each node of the recycling boundaries. In turn, FELS becomes from EIRENE one file with the element by element distribution of the mass, momentum, and energy sources.

The strategy we have just discussed is similar to what was originally used for the B2/EIRENE coupling. When going toward high density regimes (above 0.5×10^{20} m^{-3} at the midplane, say) the coupling between plasma and neutrals becomes so strong that the procedure mentioned often leads to oscillations in time. In order to avoid this, a so-called time-dependent Monte-Carlo approach (with EIRENE treated now as a subroutine) has been developed recently [31], but it is not coupled yet to FELS.

4. EDGE PLASMA GEOMETRIES AND MESH GENERATION

The two principal edge plasma geometries of current interest are those of the poloidal divertor (in, e.g., ASDEX-Upgrade, JET, ALCATOR C-Mod, D-III D, the ITER project) and of the toroidal limiter (in, e.g., TFTR, Tore Supra, FTU, TEXTOR, the IGNITOR project).

The current status of divertor modeling is reviewed in [32]. Detailed limiter modeling by means of 2D plasma codes is much less developed; the TEXTOR geometry was modeled with B2/EIRENE [22] and with a finite element code [9] (the finite element simulations were not very accurate, however, and the partly explicit algorithm used in that code was too much CPU demanding [33]). More recently, an FTU-like model geometry was treated with FELS [19].

From the point of view of the modeling, the major difference between limiter and divertor is that in the limiter case a tangency line exists between the separatrix and the limiter surface, whereas in divertor designs the plates are never at angles below some degree, say, with respect to the poloidal magnetic field. This fact makes the limiter modeling more difficult from the point of view of automatic mesh generation and boundary conditions [19]. In this paper we shall concentrate on a divertor geometry.

In both the limiter and the divertor cases it turns out that the finite element mesh must obey certain requirements (in particular, of alignment to the poloidal magnetic field; see below) in order to guarantee an accurate numerical solution. In the following we shall briefly describe the major features of two mesh generators coupled to FELS, together with their relative simplicity and ability to satisfy the above-mentioned requirements.

We conclude this section by observing that very recently [34, 35] significant

progress has been made in the generation of *adaptive* anisotropic meshes for gasdynamics, and we are currently investigating the application of those ideas to edge plasma modeling [41].

4.1. *Unstructured Mesh Generator*

The problem of modeling a magnetized plasma in the edge of a fusion device presents a peculiarity with respect to many other problems where finite elements can be of interest; because of the huge differences between parallel and perpendicular transport coefficients, the solution is expected to be fairly regular along poloidal field lines (except near targets, or X-points, or wherever sources should be present) with typical scale lengths much larger than in the radial direction [21] (a qualitatively similar situation arises in gasdynamics, albeit for very different reasons, at shocks and/or viscous boundary layers). This anisotropy implies, on the one hand, that the elements could typically be very elongated and, on the other hand, that a very careful alignment of at least one side of each element with the local direction of the poloidal field should be guaranteed, in order to make the numerical errors in the poloidal gradients as small as possible. The importance of the latter point resides in the fact that the fluxes are obtained by multiplying those gradients by large coefficients, so that large conservation errors could arise even with relatively small errors in the poloidal gradients. In the context of finite elements, numerical experiments on simple model problems with anisotropic diffusion of a scalar were reported in [9, 10, 21], showing the widening of internal layers in the case of mesh misalignment; in the context of finite volumes similar results were given in [36]. A rigorous analysis of the effects of misalignment on a finite element scheme, however, does not seem to be available yet, not even for simple model problems.

With the previous points in mind, we have modified an existing unstructured mesh generator (see [14] and references therein) which produced an unstructured mesh of nearly equilateral triangles using the advancing front technique, by including the possibility of stretching the triangles along a locally prescribed direction, producing a mesh of triangles with locally prescribed aspect ratio.

The generator is rather general, in that it can deal in principle with any magnetic geometry and any shape of the domain. It only requires the definition of the vertices of the boundary of the domain (which can be connected by straight lines, arcs of circumference, or cubic splines), and of the desired local element size, aspect ratio, and stretching direction.

However, although meshes with average misalignments as small as a few degrees can be easily obtained (see Figs. 2a–c and Table I) even in realistic poloidal divertor geometries [15], this is not always sufficient to guarantee a good accuracy in a strongly curvilinear magnetic field [16] (see also Section 5.1); therefore, we recently included the possibility of generating a structured, flux-surface-fitted mesh, as discussed in the next section.

4.2. *Structured Flux-Surface Fitted Mesh Generator*

The structured mesh generator which is coupled to FELS essentially obtains a mesh of triangles by regularly cutting into two the flux-surface-fitted, structured,

but not necessarily orthogonal, mesh of quadrilaterals obtained by the Sonnet generator [37] developed at Garching by H. P. Zehrfeld. Sonnet is coupled to the Diva equilibrium code and can work in any measured or computed (by solving the Grad–Schlueter–Shafranov equation) magnetic geometry.

From the point of view of the user, Sonnet essentially requires: a definition of the solid boundaries (vertices connected by straight lines); a partition of the separatrix; a definition of the flux-surfaces (ψ values) on which the nodes will then be requested to lie. These nodes are obtained by mapping the partition of the separatrix on the other flux surfaces using a certain number of orthogonals (to be prescribed) and of nonorthogonals, separately in each of the various subdomains (e.g., private region, scrape-off layer, core plasma). The nonorthogonals are needed in order to be able to build a structured mesh in the relevant case when the solid boundaries are not orthogonal to the poloidal field lines.

With Sonnet the average misalignments even in actual poloidal divertor geometries fall down well below one degree, and this allows (see Section 5.1) a much better accuracy than with an unstructured mesh. On the other hand, all the disadvantages of a structured mesh appear; in particular, selective refinement near the X-points or near complex solid boundaries, if needed, is impossible. In this respect, it is likely that the optimal solution (not implemented yet in any finite element code for the edge plasma) would be a mixed mesh of unstructured triangles near X-points and complex solid boundaries, and structured quadrilaterals elsewhere. Of course, such a choice would also make the code more cumbersome.

5. RESULTS

The results presented here refer to the geometry of the single null poloidal divertor as shown schematically in Fig. 1. First of all a detailed comparison will be presented between the results of FELS and those of the reference finite volume code, B2, in the case of externally imposed sources (a first comparison in the case without sources was already presented in [16]). For these tests the model implemented in B2 was simplified such as to match the model given by (1)–(5). We shall then concentrate on the results obtained with FELS/EIRENE. Convergence studies for these cases will also be presented, with emphasis on the global energy conservation properties of the present finite element formulation.

Since we are interested in discussing numerical convergence in space, very many nodes are needed, so that in order to make the problem tractable from the point of view of CPU and memory, only a subset of the whole edge region will be considered, corresponding to the lower outer leg of the scrape-off layer. With reference to Fig. 1: AX is the upper separatrix, X is the X-point, XB is the lower separatrix, BC is the target, CD (on a flux surface) qualitatively represents the first wall, and DA represents a midplane boundary. The magnetic geometry refers to shot no. 4881 of the ASDEX-Upgrade tokamak, at time t = 3s.

In Fig. 2 we show details of a typical unstructured mesh (a–c) and a typical structured mesh (d–f), corresponding in Table I to the ''unstr'' and ''1179'' entries, respectively; the two meshes have a comparable total number of nodes, and the zooms refer to the proximity of the midplane (a,d), of the X-point (b,e), and of

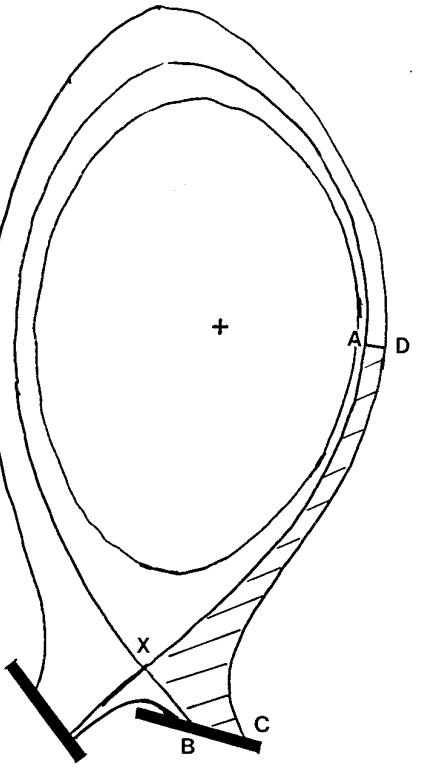

FIG. 1. Schematic view (poloidal cut) of the full axisymmetrical plasma edge region in a single null poloidal divertor tokamak. The results presented here refer to the subset identified by the cross section AXBCDA of the domain Ω, which is shown with a dashed pattern.

the target (c,f). A reference-structured mesh for the whole domain is shown in [16]. Notice that an apparently reasonable alignment can be obtained with the unstructured mesh generator, which can additionally provide a much finer (selective) resolution near the X-point.

Only steady state results will be presented here, the steady state being defined as the condition when the maximum relative variation $|\Delta u|/u$ of ρ, p_e, and $\mathscr{E}$ on Ω between two successive time steps is below a certain threshold (10^{-5} to 10^{-4}, say), and the time step Δt has reached a previously fixed maximum value (typically 10^{-2} to 1 times the sound time scale).

In all cases presented here the boundary conditions (b.c.) imposed on the plasma are as follows:

on AX Dirichlet b.c. on the density (1×10^{19} m^{-3}) and the temperatures ($T_e = T_i = 50$ eV), together with vanishing radial flux of parallel momentum;

on XB symmetry is assumed, i.e., a homogeneous Neumann b.c.; this boundary is an artificial one, because the domain should in principle include the private region;

on BC—an outflow boundary for (1)—no boundary condition is imposed on the density, whereas a Dirichlet b.c., isothermal sonic parallel flow, is imposed on the

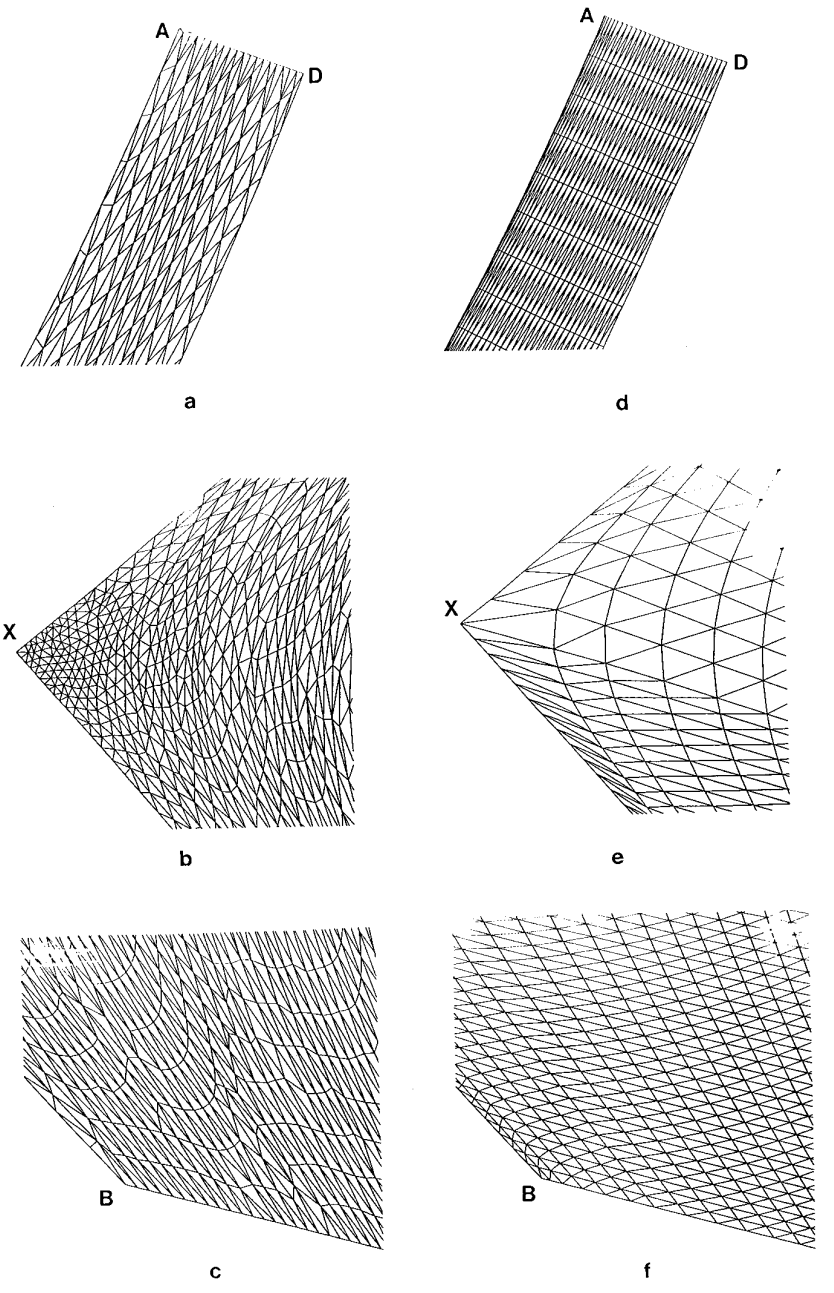

FIG. 2. Details of typical unstructured (a–c) and structured (d–f) finite element meshes with comparable number of nodes (entries ''unstr'' and ''1179'' in Table I): (a,d) near midplane, (b,e) near X-point, (c,f) near target.

TABLE I
Characteristics of Finite Element Meshes Used

MESH	1134	1136	1177	1178	1179	1180	1187	1188	1189	Unstr.
Elements	1344	2856	1798	3596	7192	7192	7440	6232	7488	7301
Nodes (total)	731	1530	960	1890	3750	3717	3984	3300	3971	3789
Nodes on upp. sep.	16	31	22	43	85	43	169	85	169	88
Nodes on separatrix (tot)	43	85	32	63	125	63	249	165	209	119
Nodes on midplane	17	18	31	31	31	61	16	20	20	19
Nodes on target	17	18	31	31	31	61	16	20	20	62
$\langle\alpha\rangle$ [°][a]	0.69	0.55	0.52	0.31	0.24	0.26	0.42	0.36	0.33	2.4
$\langle A\alpha\rangle/A_\Omega$ [°][b]	0.74	0.49	0.68	0.38	0.31	0.35	0.61	0.52	0.61	2.1
$\alpha > 1°$ [%][c]	10	6	10	2	2	2	8	6	6	68
$\alpha > 5°$ [%][d]	0.97	0.81	2.2	0.08	0.1	0.07	0.3	0.2	0.3	8

[a] Average misalignment (see text).
[b] Area weighted average misalignment.
[c] Fraction of elements with misalignment >1°.
[d] Fraction of elements with misalignment >5°.

momentum balance; as to the energy balances we impose Robin b.c. in the usual [4] form of sheath energy transmission factors $\delta_e = 4.5$ and $\delta_i = 2.5$ for ions and electrons separately (in other words we impose $\Gamma^E_{j\|} = \delta_j \Gamma_\| T_j$, with $j = e, i$);

on CD we impose a Robin b.c. on the continuity equation (radial particle flux fixed by a given radial scale length (= 0.01 m) for the density), vanishing radial shear of the parallel flow velocity, and Dirichlet temperature pedestals ($T_e = T_i =$ 2 eV); also this boundary is rather artificial, because the actual first wall hardly coincides with a flux surface;

on DA, finally, we assume symmetry/stagnation conditions to apply; also this boundary is artificial, because the domain should in principle extend around the main plasma up to the inner target.

As a general comment on the use of artificial boundaries we can say that XB and DA are considered here only because of the need to use a smaller domain for performing convergence studies with a still tractable number of nodes; for a treatment with FELS of the whole plasma edge, without need of introducing these artificial boundaries, see [21]. On the other hand, if CD should actually coincide with the first wall, from the physics point of view one would not really know at present what to impose there as a b.c., because it would then have portions at very small (but not vanishing) angles to the magnetic surfaces (see the discussion on this point in [19]).

As to the various coefficients present in (1)–(5) we used, for the results reported in this paper, the following values: uniform anomalous D in (3), and momentum diffusivity (for radial viscosity), equal to 1 m^2/s, uniform anomalous ion and electron heat diffusivities, equal to 2 m^2/s; classical [2] values for the parallel ion viscosity, the parallel electron, and ion heat conductivities, and the electron–ion energy interchange term Q_e in (4). The ion species is deuterium.

All of the computations were performed either on a DEC-Alpha 3000 workstation at Torino, or on an IBM-RISC 6000 workstation at Garching.

5.1 *Comparison against Finite Volumes with External Sources*

In this section we assume that on the plasma act only an external mass source S^m and an external electron energy sink $S_e^E = -E_i \times S^m$ (with $E_i = 30$ eV); both could be related, e.g., to ionization. The mass source is distributed as a Gaussian bell centered in B, with decay lengths of 0.05 m (0.04 m) in the direction perpendicular (parallel) to the target. The intensity of S^m is chosen such as to contribute a significant fraction (about 80%) of the total mass outflux through the target.

We shall now compare the steady state radial profiles of density and temperatures (along the Neumann/Robin boundaries AD and BC) obtained with FELS, against those obtained with the finite volume code B2. B2 is designed to run on a structured orthogonal mesh of quadrilaterals, and the meshes on which FELS and B2 are compared have the same location of the nodes.

The most important features of all of the meshes used in the present paper have been collected in Table I. B2 was run on two (quasi-orthogonal) meshes (1177 and 1179 entries of Table I) and FELS was run also on several other meshes, both structured (e.g., 1178 and 1189) and unstructured (''unstr'' entry). The comparison between the results obtained with the two codes on mesh 1179 is shown in Figs. 3a–f. One sees that very good agreement is obtained for all relevant quantities. Notice also that the weak nonorthogonality of meshes 1177 and 1179 appears to play a very little role in the B2 solution, although B2 strictly speaking assumes an orthogonal mesh (more precisely, ψ gradients are computed in B2, assuming that the centers of mass of two ''radially'' adjacent quadrilaterals are connected by a line which is orthogonal to the local flux surface, and this is exact only if the mesh is orthogonal).

Another very important point is the comparison of the spatial convergence speed of the two approaches: in Figs. 4a–f we report the profiles obtained running FELS on some meshes (1177, 1178, 1179, 1189) which are poloidally progressively finer, plus one unstructured mesh of comparable poloidal resolution as 1179. One notices first that the results on the unstructured mesh present significant errors, notwithstanding the relatively small misalignments of the elements (this was already noticed in a rectangular geometry; see [16]); second, it appears that the results of FELS converge spatially only after a mesh with significant poloidal resolution (compared with the relatively weak poloidal gradients in the solution) is used. Even more importantly, however, one can consider the comparison between the results obtained from B2 on the two meshes 1177 and 1179, as shown in Figs. 5a–f. It is clear that finite volumes converge spatially faster than finite elements in this example, particularly as to the temperature profiles.

Our understanding of the previous difference between finite elements and finite volumes is that the fact of satisfying exactly approximate global conservation laws (as a consequence of local conservation) gives the finite volume approach a greater robustness (in the sense of a weaker mesh sensitivity) than the finite elements. In the case of the continuity equation the fluxes are themselves unknowns, so that

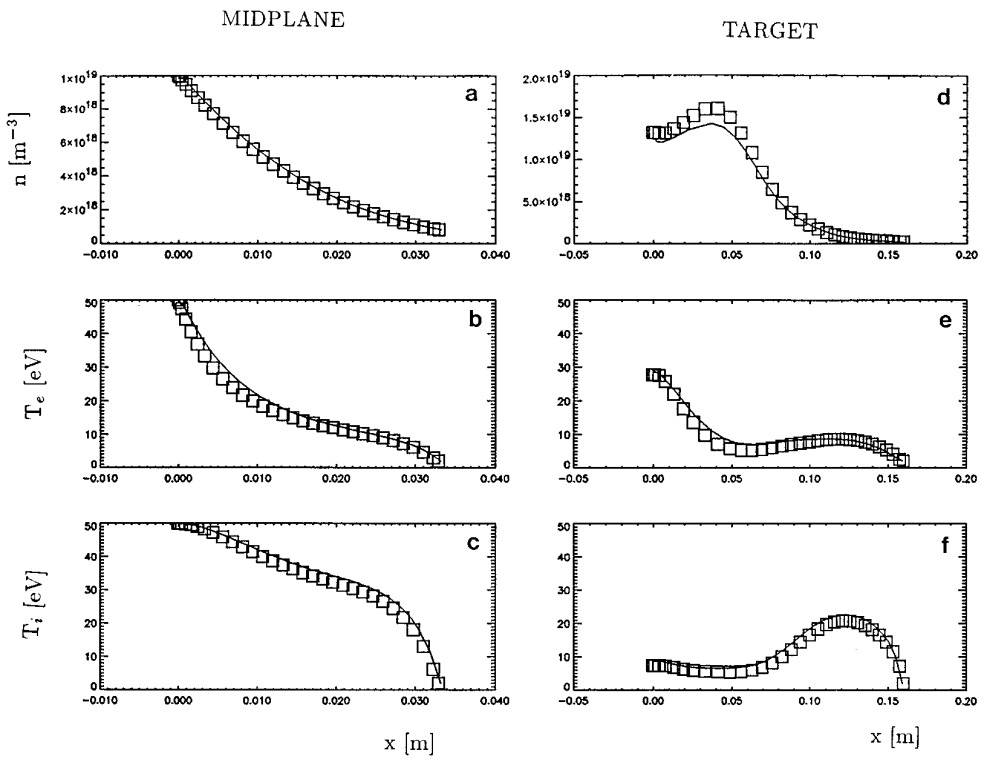

FIG. 3. Comparison between FELS (solid lines) and B2 (open squares) steady state results on mesh 1179, with external sources: (a–c) midplane profiles versus radial distance [m] from separatrix, (d–f) target profiles versus distance [m] from B along target; (a,d) particle density n [m^{-3}], (b,e) electron temperature T_e [eV]; (c,f) ion temperature T_i [eV].

particles are always very well conserved, also by finite elements. Typical relative particle conservation errors, defined as $|1 - (\text{net particle outflow through } \partial\Omega)/(\text{particle source integrated over } \Omega)|$, are $\leq O(10^{-3})$; for the same reason, the plasma density is the quantity showing the least mesh dependence. On the contrary, in the energy balances the conductive fluxes are proportional to gradients of unknowns, which are *not* continuous at element interfaces. Therefore, the local conservation concept is meaningless in the context of P1 Galerkin finite elements, and global energy conservation can be obtained with them only to the limit of the vanishing element size.

With P1 Galerkin finite elements, in an elliptic problem, the error in the unknowns should decrease quadratically with some element size, whereas the error in the gradients should decrease linearly. Since the energy flux is a combination of conductive and convective contributions we expect a behavior of the error in global energy conservation somewhere between linear and quadratic. This was verified by running the same case on several meshes (see Table I): we demonstrate numerically the convergence of FELS with external sources in Fig. 6, where the absolute conservation error in total energy is reported as a function of some parameters characterizing

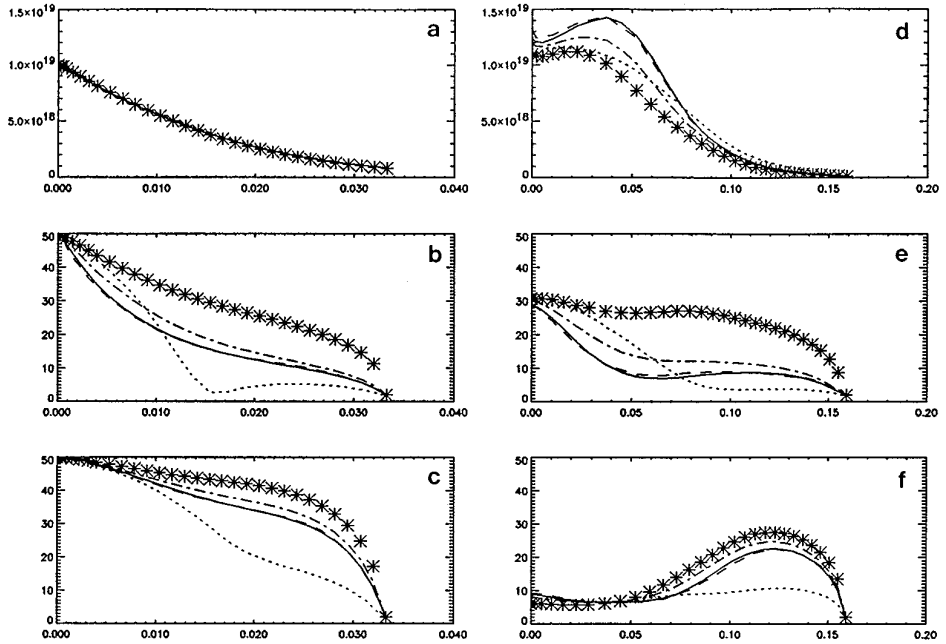

FIG. 4. Spatial convergence of FELS (same quantities of Figs. 3a–f): steady state profiles on "Unstr." mesh (dotted lines); on mesh 1177 (stars); on mesh 1178 (dash-dotted lines); on mesh 1179 (solid lines); on mesh 1189 (dashed lines).

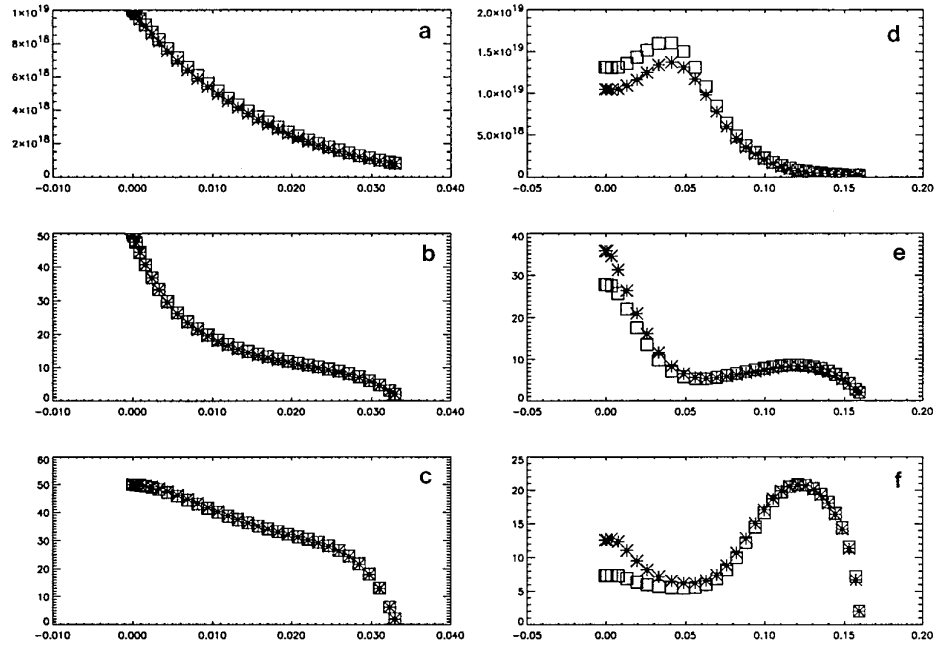

FIG. 5. Spatial convergence of B2 (same quantities of Figs. 3a–f): steady state profiles on mesh 1177 (stars); on mesh 1179 (open squares).

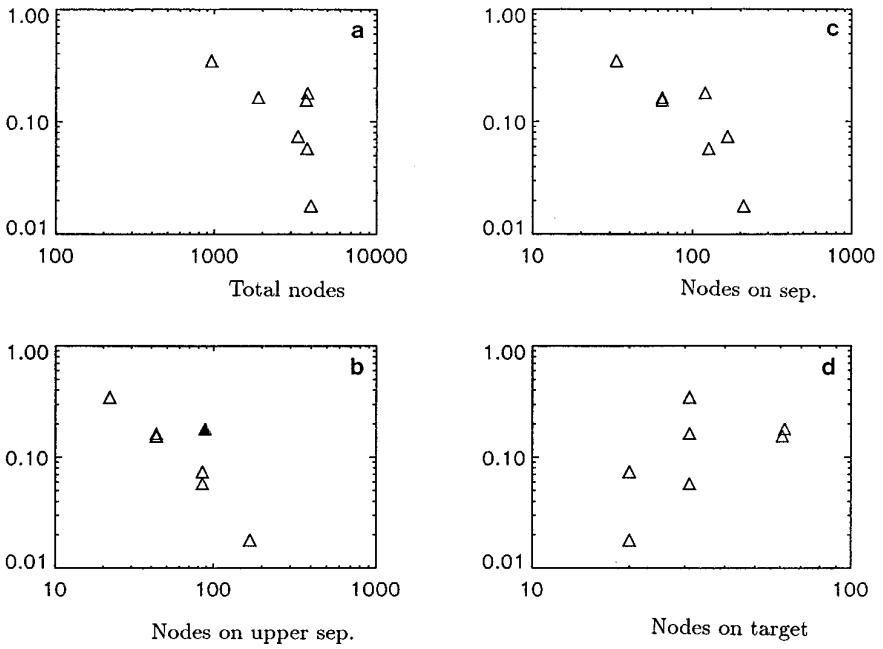

FIG. 6. Absolute error in total energy conservation [a.u.] for FELS with external sources versus total number of nodes (a), number of nodes on upper separatrix (b), number of nodes on separatrix (c), number of nodes on target (d). (The solid symbol in (b) refers to the solution computed on the unstructured mesh.)

different meshes. Notice that, as expected from the previous discussion, poloidal, as opposed to radial, resolution is crucial for convergence.

From the practical point of view it is important to assess if still reasonable element sizes (with respect to memory and CPU requirements) can lead to sufficiently small conservation errors, for the problem at hand, or vice versa, what order of conservation error implies a sufficiently correct solution (the latter point is obviously important in order to assess if a solution is "good," without having to compare with another code); this is essentially equivalent to determining numerically the constant in an error estimate. What turns out is that the relative error in global energy conservation, defined similarly as the relative particle conservation error, needs to be $\leq O(10\%)$, in order to have a solution correct to some percentage. Conservation errors, however, may also be important *per se*, because one has engineering constraints not only on the accuracy of the solution but also on the accuracy of the energy fluxes themselves, in particular for the determination of the power load on the plate BC.

It is clear that the relative inaccuracy of the finite element method for the problem at hand partly resides in the choice of P1 basis functions. It is not to be expected, however, that this problem can be easily solved by increasing the degree of interpolation, because in 2D only going up to P5, which is extremely impractical, could guarantee continuity of the unknowns and of all their first partial derivatives, i.e., of the fluxes, at the element interfaces [24]. An interesting alternative could be

given by mixed type finite element methods, where the fluxes are treated as dependent variables (unknowns). These methods reduce in certain cases to finite volume methods and, as such, guarantee local conservation; however, they have the problem that it is rather difficult to accomodate into their framework an upwinding recipe [38]. Possibly the most interesting and readily applicable development which could make finite elements more competitive against finite volumes lies in the adaptive techniques for anisotropic meshes; for gasdynamics problems these techniques already allow obtaining solutions of the same quality with both methods on optimized meshes [39].

5.2 *Plasma Coupled to Monte-Carlo Neutrals*

In this section we present results of a typical FELS/EIRENE coupled run. The plasma coefficients and boundary conditions have been chosen as discussed above. The sources acting on the plasma, on the contrary, are now self-consistently computed by the Monte-Carlo code EIRENE. Ten thousand particles are launched in EIRENE from the target BC. For the case at hand the following particle species are followed by EIRENE: D_0, D_2, D_2^+ (molecular ions are considered as a trace population).

The boundary conditions for the neutral particles are as follows: on the target BC a particle recycling coefficient equal to 1 is assumed; for all other boundaries we assume that the fast component of each species described by EIRENE is reflected, whereas 2% of the thermal component is pumped, so that steady state conditions for the neutrals can be obtained. Notice that these boundary conditions are purely demonstrative, which can be justified within the mainly computational framework of the present paper; if a comparison with experiments or a detailed physics discussion would be aimed at, different boundary conditions and even a different (larger) domain, e.g., extending up to the actual vessel walls, should be used for the neutrals.

In order to give an idea of the computational nature of the coupling between plasma and neutrals we show in Fig. 7 the behavior of the relative variation $\Delta\Gamma/\Gamma$ (with respect to the ''initial'' one; see below) of the plasma particle flux integrated over the target surface ($\Gamma \equiv \int_{BC} \mathbf{\Gamma}\cdot\mathbf{n}\, dS$), as a function of the time step number. A threshold of 1% in $\Delta\Gamma/\Gamma$ was used as a criterion to call again EIRENE (the ''initial'' above refers to the conditions either before the first call of EIRENE, or right after the temporarily last call). Notice that also the time step Δt is brought back to its minimum value ($O(10^{-6})$ times the sound time in this case) right after a call of EIRENE.

Initially the plasma is set up as a steady state obtained with a simplified model for the neutrals, so that EIRENE produces significant changes in the plasma rather fast and must be called relatively frequently. Eventually the calls to EIRENE become less and less frequent, until a steady state is reached ($|\Delta u|/u < 10^{-4}$, together with $\Delta t = \Delta t_{\max} = 10^{-2}$ times the sound time, is used as steady state definition here). This required four runs of FELS/EIRENE, about 7000 plasma steps, and 47 EIRENE calls; each run following the first one is obviously restarted from the last obtained solution.

An example of contour plots of the steady state plasma profiles obtained in such

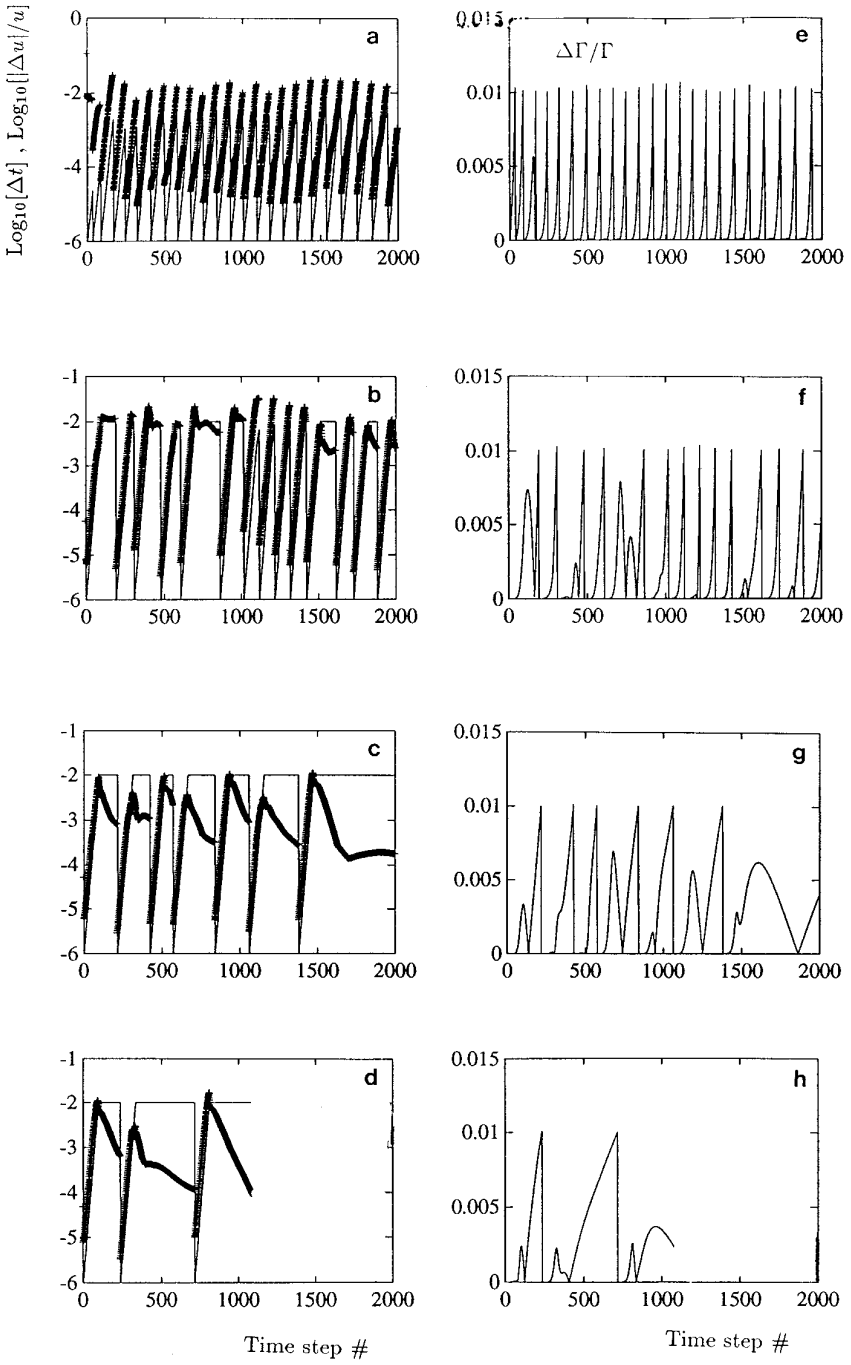

FIG. 7. Convergence in time to steady state of a FELS/EIRENE coupled run versus time step number: initial run (a,e), first restart (b,f), second restart (c,g), third restart (d,h); Figs. a–d, time step size Δt normalized to sound time (solid lines), maximum relative variation $|\Delta u|/u$ of ρ, p_e, $\mathscr{E}$ over Ω between two successive time steps ("+" symbols); Figs. e–h, maximum relative variation $\Delta\Gamma/\Gamma$ (with respect to the "initial" one; see text) of the particle flux integrated on the target.

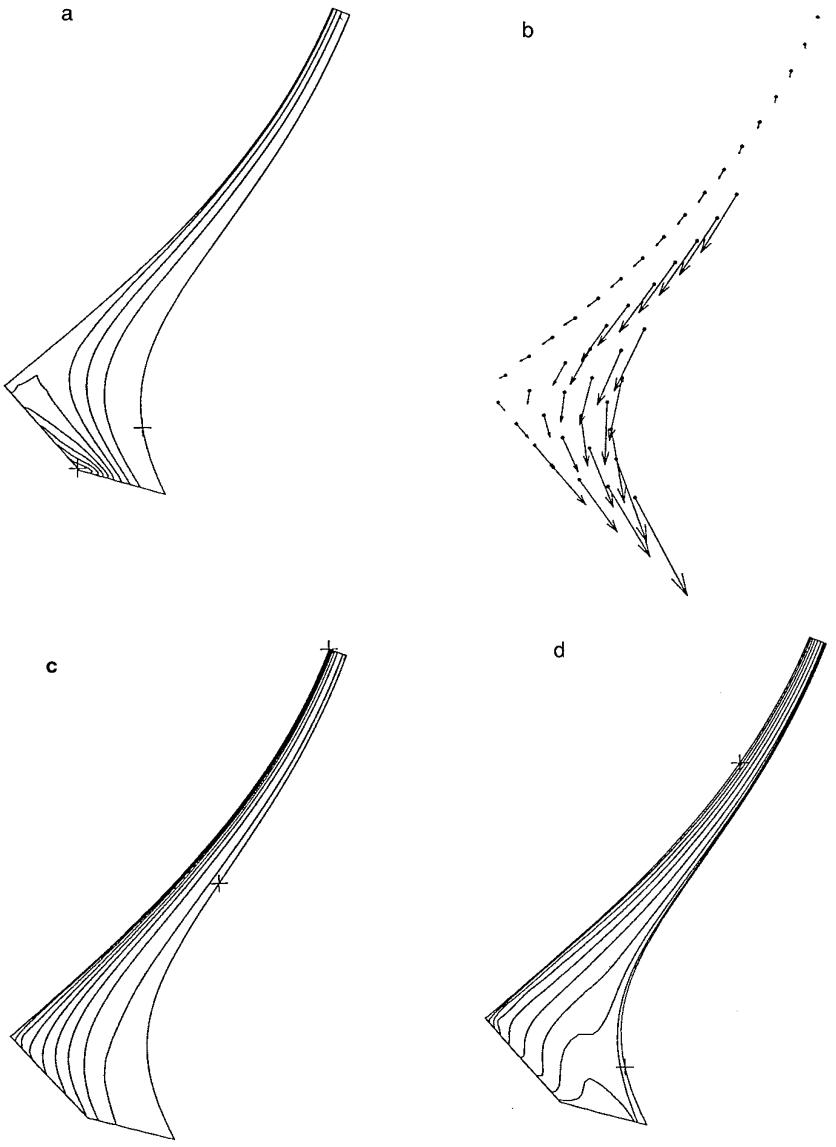

FIG. 8. Contour plots of steady state results of a FELS/EIRENE coupled run on mesh 1189: particle density (a); poloidal flow velocity field (b); electron temperature (c); ion temperature (d).

a way is shown for the sake of completeness in Figs. 8a–d. The essential standard physical features of the problem at hand are well reproduced by our numerical solution, namely the poloidal uniformity of density and temperatures far from the target where the sources are acting, and the significant increase of the density, acceleration of the poloidal flow, and drop of the temperatures near the target.

Of course, it is expected that the considerations on global energy conservation made in the previous section become even more important here, because of the stronger localization of the Monte-Carlo sources with respect to the externally

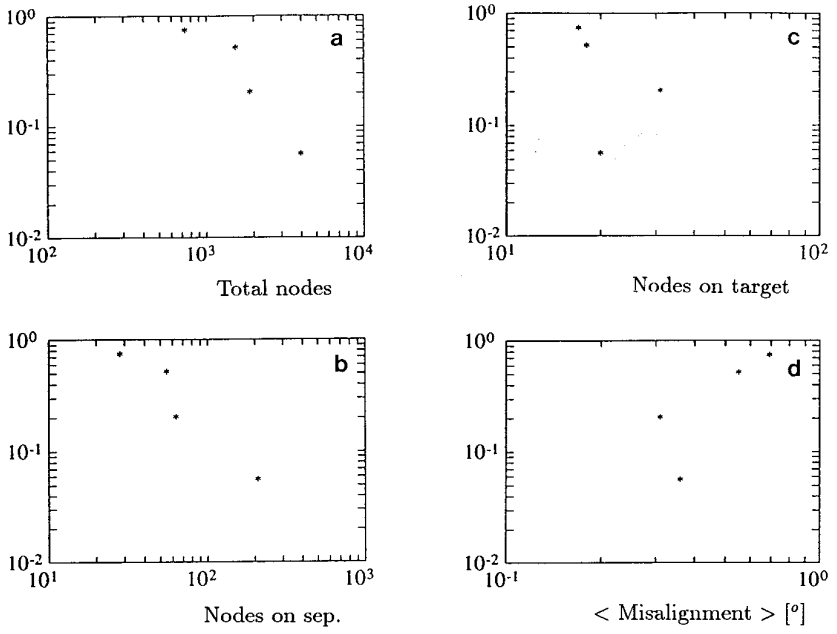

FIG. 9. Absolute error in total energy conservation [a.u.] for FELS with self-consistent EIRENE sources versus total number of nodes (a), number of nodes on separatrix (b), number of nodes on target (c), average misalignment (degrees) (d). (The average misalignment is computed by taking the arithmetical average over Ω of the minimum angle formed by the sides of each triangle with the local direction of the poloidal magnetic field at the center of gravity of that triangle.)

imposed ones, and the consequently stronger gradients, which will worsen the accuracy (the particle source contributes now about 98% of the total particle outflux through the target). In order to better understand which of the mesh features most influences the quality of the solution we show in Figs. 9a–d the results of a convergence study of the FELS/EIRENE coupled system, reporting the absolute error in global energy conservation versus some of the mesh features which could be thought of as important in principle. Clearly, the extreme anisotropy of the problem at hand must play a major role in determining which parameter is actually important and which is not.

The same case has been run on the structured meshes 1134, 1136, 1178, and 1189 (see Table I). As relevant mesh parameters we consider here the dependence of the energy conservation error on: the total number of nodes, Fig. 9a; the number of nodes on the separatrix (poloidal resolution), Fig. 9b; the number of nodes on the target (radial resolution), Fig. 9c; and the average misalignment of the mesh, Fig. 9d. The convergence of the coupled code system FELS/EIRENE is demonstrated numerically by Figs. 9a,b. It is clear from Fig. 9b that a sufficiently large poloidal resolution, everywhere and not only in the region where the sources are active, is essential. As in the case of external sources, no significant correlation between radial resolution and convergence appears to exist (Fig. 9c) in the case at hand. Finally, one sees from Fig. 9d that a sufficiently small misalignment is required for minimizing the error in energy conservation, but reducing the misalignment by

increasing the radial resolution can be counterproductive, unless a sufficiently good poloidal resolution is also guaranteed. The latter point can be compared with that found in rectangular geometry with straight field lines [16], where reducing the misalignment always reduces the error; therefore it can be concluded that poloidal resolution is essential when FELS is applied to realistic curvilinear magnetic geometries.

6. CONCLUSION AND PERSPECTIVE

A finite element code, FELS, has been developed for 2D 2-fluid modeling of the tokamak plasma edge. FELS is advanced with respect to other finite element codes, e.g., [10], because it can deal with realistic curvilinear geometries, and because it is coupled to the Monte-Carlo code for neutrals, EIRENE, which allows a detailed and realistic description of the interactions between plasma and recycling atoms and molecules. These extensions also involve significant new computational issues, as was shown here.

Several aspects make the edge problem particularly challenging and difficult, among which the extreme anisotropy in the transport coefficients, the significant nonlinearities, the number of different particle species involved, and, last but not least, the basic uncertainities concerning all plasma transport phenomena in the direction perpendicular to the magnetic surfaces and some of the boundary conditions.

In the present paper we have compared the results of FELS with those of the finite volume code B2, showing very good agreement between the two codes, in the case of external sources. Also, a slower spatial convergence of FELS with respect to B2 was noticed.

Typical steady state plasma profiles obtained with FELS/EIRENE in a subset of the ASDEX-Upgrade poloidal divertor geometry were shown, and the numerical convergence of the coupled code system was demonstrated. Results pertaining to the full divertor geometry, including inner and outer plate, private region and a portion of the core plasma have been discussed elsewhere [21].

In the perspective of future developments of FELS one can (somewhat artificially) consider separately the computational issues and the physics issues.

From the computational point of view the *practical* need for finite elements (as opposite to the more conventional finite volumes) has to be demonstrated. This is, in the opinion of the author, the next top priority and will come from a comparison (which is beyond the scope of the present work) against finite volume codes, e.g., [11, 12], recently upgraded to a 9-point computational stencil—the least prerequisite to work on a nonorthogonal grid in the finite volume framework. Modern adaptive meshing techniques could also help in bridging the gap between the two methods [41].

From the physics point of view, the major required extensions of FELS are: going from 2-fluid to multifluid, i.e., the inclusion of impurities in the plasma model, and (possibly) the inclusion of drifts and electric currents. This would bring FELS to the level of physics currently implemented in the most advanced finite volume codes and allow a meaningful comparison with experiments.

ACKNOWLEDGMENTS

Many people have contributed to this project over the years. In particular I thank K. Lackner for continuing encouragement and support, D. Coster and R. Schneider for modifying and running the B2 code for the benchmarks presented here, P. Boerner, B. Braams, F. Brezzi, C. Canuto, R. Chodura, D. Coster, J. Neuhauser, D. Reiter, A. Russo, R. Schneider, and H. P. Zehrfeld for collaboration, discussions, and help on both physics and computational issues. I finally acknowledge the extremely kind and fruitful hospitality of the division Tokamakphysik (formerly Theorie 3) of the Max-Planck-Institut für Plasmaphysik at Garching, Germany, of which I came to feel myself more than a guest. This work was partially financially supported by EURATOM through several mobility contracts, and by CNR.

REFERENCES

1. P. C. Stangeby and G. M. McCracken, *Nucl. Fusion* **30,** 1225 (1990).
2. S. I. Braginskii, in *Reviews of Plasma Physics*, Vol. 1, edited by M. A. Leontovich (Consultants Bureau, New York, 1965), p. 205.
3. S. I. Krasheninnikov, *Contrib. Plasma. Phys.* **34,** 151 (1994).
4. R. Chodura, in *Physics of Plasma–Wall Interactions in Controlled Fusion*, edited by D. E. Post and R. Behrisch (Plenum, New York, 1986, p. 99).
5. B. J. Braams, NET Report EUR-FU/XII-80/87/68 (1987).
6. R. Schneider *et al.*, paper F1-CN-64/D3-1, in *Proceedings of the IAEA Conference, Montreal, Canada, 1996.*
7. R. Zanino, Max-Planck-Institut Report IPP 5/43 (1991).
8. R. Zanino, *J. Nucl. Mater.* **196–198,** 326 (1992).
9. Th. Puetz, *et al.*, *Contrib. Plasma Phys.* **34,** 404 (1994).
10. R. A. Vesey and D. Steiner, *J. Comput. Phys.* **116,** 300 (1995).
11. R. Simonini, *et al.*, *Contrib. Plasma Phys.* **34,** 368 (1994).
12. G. R. Smith *et al.*, *J. Nucl. Mater.* **220–222,** 1024 (1995)
13. A. N. Brooks and T. J. R. Hughes, *Comput. Methods Appl. Mech. Engrg.* **32,** 199 (1982).
14. R. Zanino *et al.*, *Contrib. Plasma Phys.* **32,** 432 (1992). [R. Zanino, *Contrib. Plasma Phys.* **33,** 149 (1993)].
15. R. Zanino and A. Russo, *Contrib. Plasma Phys.* **34,** 410 (1994).
16. R. Zanino, *Contrib. Plasma Phys.* **36,** 407 (1996).
17. R. Marchand, private communication, 1996.
18. M. Simard and R. Marchand, *Contrib. Plasma Phys.* **36,** 401 (1996).
19. R. Zanino, Finite element fluid modeling of heat loads on a toroidal limiter, in *12th International Conference on Plasma Surface Interactions in Controlled Fusion Devices, 1996.*
20. M. Baelmans *et al.*, *Contrib. Plasma Phys.* **36,** 117 (1996).
21. R. Zanino, in *Numerical Methods in Engineering '96*, edited by J.-A. Desideri *et al.* (Wiley, New York, 1996), p. 1035.
22. M. Baelmans, KFA Report Jül-2891 (1994).
23. K. Theilhaber and C. K. Birdsall, *Phys. Fluids B* **1,** 2260 (1989).
24. C. Johnson, *Numerical Solutions of PDEs by the Finite Element Method* (University Press, Cambridge, 1987).
25. D. P. Coster *et al.*, *Contrib. Plasma Phys.* **36,** 150 (1996).
26. E. Sabbadin and R. Zanino, *Contrib. Plasma Phys.* **36,** 333 (1996).
27. D. Reiter, KFA Report Jül-2599 (1992).
28. D. A. Knoll *et al.*, *Contrib. Plasma Phys.* **36,** 328 (1996).

29. D. Reiter *et al.*, Monte Carlo simulation of neutral gas transport including neutral-neutral interactions, in *12th International Conference on Plasma Surface Interactions in Controlled Fusion Devices, 1996.*
30. D. Reiter, in *Atomic and Plasma-Material Interaction Processes in Controlled Thermonuclear Fusion,* R. K. Janev and H. W. Drawin (editors) (Elsevier, Amsterdam, 1993) p. 243.
31. D. Reiter *et al.*, *J. Nucl. Mater.* **220–222,** 987 (1995).
32. A. Loarte, "Understanding the edge physics of divertor experiments by comparison of 2-D edge code calculations and experimental measurements," presented at the *12th International Conference on Plasma Surface Interactions in Controlled Fusion Devices 1996.*
33. Th. Puetz, private communication, 1995.
34. M. Fortin *et al.*, in *Computational Fluid Dynamics '96*, edited by J.-A. Desideri *et al.* (Wiley, New York, 1996), p. 174.
35. M. J. Castro-Diaz *et al.*, in *Computational Fluid Dynamics '96*, edited by J.-A. Desideri *et al.* (Wiley, New York, 1996), p. 181.
36. M. Day, Ph.D. thesis, University of California Los Angeles, 1995.
37. R. Schneider *et al.*, *J. Nucl. Mater.* **196–198,** 725 (1992).
38. F. Brezzi and A. Russo, private communication, 1995.
39. M. Fortin *et al.*, Mesh adaptation for structured and unstructured grids: Towards a solver-independent solution, in *3rd ECCOMAS Computational Fluid Dynamics Conference 1996.*
40. D. A. Knoll and P. R. McHugh, *J. Comput. Phys.* **116,** 281 (1995).
41. R. Zanino and F. Subba, *Contrib. Plasma Phys.* (in press).