Edge equilibrium code for tokamaks

Xujing Li, Leonid E. Zakharov, and Vladimir V. Drozdov

Edge equilibrium code for tokamaks

Xujing Li, Leonid E. Zakharov, and Vladimir V. Drozdov

Edge equilibrium code for tokamaks

Xujing Li,[1] Leonid E. Zakharov,[2] and Vladimir V. Drozdov[3]

[1]*Institute of Computational Mathematics and Scientific/Engineering Computing, Academy of Mathematics and Systems Science, Chinese Academy of Sciences, P.O. Box 2719, Beijing 100190, China*
[2]*Princeton Plasma Physics Laboratory Princeton, MS-27 P.O. Box 451, New Jersey, USA*
[3]*Euratom/CCFE Fusion Association, Culham Science Centre, Abingdon OX14 3DB, United Kingdom*

The edge equilibrium code (EEC) described in this paper is developed for simulations of the near edge plasma using the finite element method. It solves the Grad-Shafranov equation in toroidal coordinate and uses adaptive grids aligned with magnetic field lines. Hermite finite elements are chosen for the numerical scheme. A fast Newton scheme which is the same as implemented in the equilibrium and stability code (ESC) is applied here to adjust the grids. © 2014 AIP Publishing LLC. [http://dx.doi.org/10.1063/1.4861369]

I. INTRODUCTION

With time the demand of equilibrium calculations of tokamak plasma is growing. In addition to the well established use of equilibrium codes[1,2] for controlling the shape of tokamak plasmas and for magneto-hydrodynamic (MHD) stability studies, new applications include the sophisticated control of vertical stability with realistic models for plasma external structures,[3,4] the analysis of variances in equilibrium reconstruction,[5–7] special equilibrium solvers in transport simulations (e.g., Refs. 8–10), and refined calculations of the plasma core and of the plasma edge. The literature on equilibrium calculations is extensive. The review article[11] covers well the numerous codes and approaches before 1991, an additional brief review of numerical codes is given in Ref. 12. The recent article[13] reviews also the recent codes.

Despite extensive efforts in addressing the demands, there is still a gap in coverage of equilibrium problems by the existing codes. Simulations of the high temperature tokamak plasma, which outstanding anisotropy increases with the increase in the plasma temperature and the sizes of machines, require the use of flux coordinate based equilibrium calculations. At present, most of experiment relevant simulations are done initially with so-called r–z equilibrium codes.

The needs in adaptive solvers were understood long time ago[14–21] and led to creation of POLAR (2-dimensional) and VMEC (2- and 3-dimensional). The first one uses a finite difference scheme in both radial and poloidal directions and is uniquely capable of reproducing extreme plasma configurations, e.g., with a high beta. The second one uses Fourier transformations in both the toroidal and poloidal directions and became the leading code for 3-D stellarator calculations. Another example is a very fast equilibrium and stability code (ESC)[12] which uses perturbed equilibrium equations as a mean for solving the Grad-Shafranov (GSh) equation.

At the same time, configurations with a separatrix as the plasma boundary, typical in tokamak experiments, are not covered by the existing flux coordinate based codes. Because of this deficiency, this kind of codes are not used for free-boundary plasma equilibria and equilibrium reconstructions, which are highly requested.

In this paper, we make a decisive step in extending the applicability of the flux coordinate equilibrium codes. The edge equilibrium code (EEC), described here, uses finite elements[22,23] for representing the magnetic configuration in a boundary layer. We intentionally did not extend the scope of EEC to the entire cross-section. For the plasma core, Fourier decomposition is the best and fastest representation, implemented, e.g., in the above mentioned ESC. EEC uses Hermite finite elements and compensates for the loss of accuracy in ESC when the configuration is extended toward the separatrix: both codes are interfaced through a virtual boundary, thus making full plasma coverage consistent with the accuracy and speed of calculations.

Earlier, same the Hermite elements were used in the CHEASE code[24] with no grid adaptation. We also found that our approach in EEC is very close to formulation of the solver for the HELENA code in Ref. 25, which is based on Hermite (bicubic splines) finite elements and mentions the grid adaptation. Still the implementation of finite elements in EEC is different, there is no restrictions on the plasma shape. A special fast solver of matrix equations and the grid advancing scheme were developed for EEC, and its link with the core simulation code ESC makes the speed of the code system uniquely fast.

Section II of the paper explains the variational principle, use of Hermite finite elements, boundary conditions and the fast solver of matrix equations representing the GSh equation in curvilinear toroidal coordinates. Section III describes the fast Newton scheme for advancing the coordinate system. Section IV gives examples of equilibria generated by the ESC-EEC code system, and the Sec. V outlines the near term extensions of EEC.

II. ENERGY FUNCTIONAL AND FINITE ELEMENTS

Throughout the paper, m, T, MA, MPa, V s are adopted as units for lengths, magnetic field strength, currents, pressure, and magnetic fluxes. The notations follow the earlier paper on ESC,[12] with cylindrical coordinates r, ϕ, z, poloidal and toroidal fluxes Ψ and Φ, poloidal and toroidal magnetic fields B_p and B_ϕ, plasma pressure p, current density $\mathbf{j}$, and

$$\mu_0 \equiv 0.4\pi, \quad \bar{\Psi} \equiv \frac{\Psi}{2\pi}, \quad \bar{p} \equiv \mu_0 p, \quad \bar{\Phi} \equiv \frac{\Phi}{2\pi},$$

$$\bar{F} \equiv rB_\phi, \quad \bar{\mathbf{j}} \equiv \mu_0 \mathbf{j}, \tag{1}$$

$$\mathbf{B} = \mathbf{B}_p + \mathbf{B}_\phi = \frac{1}{r}(\nabla\bar{\Psi} \times \mathbf{e}_\phi) + \frac{1}{r}\bar{F}(\bar{\Psi})\mathbf{e}_\phi. \tag{2}$$

The notations a, θ, ϕ are used for curvilinear toroidal coordinates, specified by

$$r = r(a, \theta), \quad z = z(a, \theta) \tag{3}$$

with the metric tensor coefficients[12]

$$K \equiv \frac{g_{\theta\theta}}{rD}, \quad N \equiv \frac{g_{a\theta}}{rD}, \quad M \equiv \frac{g_{aa}}{rD}, \quad D \equiv r'_\theta z'_a - z'_\theta r'_a,$$
$$g_{aa} = r'_a r'_a + z'_z z'_a, \quad g_{\theta\theta} = r'_\theta r'_\theta + z'_\theta z'_\theta, \quad g_{a\theta} = r'_a r'_\theta + z'_a z'_\theta. \tag{4}$$

It is well known[26] that the Grad-Shafranov equation, which describes the axisymmetrical toroidal equilibria,

$$\Delta^*\bar{\Psi} \equiv \frac{\partial^2\bar{\Psi}}{\partial r^2} - \frac{1}{r}\frac{\partial\bar{\Psi}}{\partial r} + \frac{\partial^2\bar{\Psi}}{\partial z^2} = -r\bar{\mathbf{j}}_\phi = -r^2 P(\bar{\Psi}) - T(\bar{\Psi}), \tag{5}$$

$$P(\bar{\Psi}) \equiv \mu_0 \frac{dp}{d\bar{\Psi}}, \quad T(\bar{\Psi}) \equiv \bar{F}\frac{d\bar{F}}{d\bar{\Psi}}, \quad \bar{\mathbf{j}}_\phi = rP + \frac{1}{r}T \tag{6}$$

can be derived by minimization of the following energy functional with respect to $\bar{\Psi}$

$$\bar{W} = \frac{1}{2}\int_\Omega \left[\frac{|\nabla\bar{\Psi}|^2}{r^2} - 2\bar{p}(\bar{\Psi}) - \frac{\bar{F}^2(\bar{\Psi})}{r^2}\right]rdS + \oint_\Gamma \bar{\Psi}B_p^e dl. \tag{7}$$

Here, $dS = drdz$ is the surface element of the toroidal plasma cross-section, dl is the length element of its boundary contour Γ, and B_p^e is a given poloidal magnetic field at the plasma boundary. The bar in $\bar{W}$ notation is used because Eq. (7) is different from the real energy functional W by a factor 5/2.

The contour integral term in the functional (7), usually omitted in the theory papers, is ignorable only for the Dirichlet boundary conditions for the GSh equation, i.e., if $\bar{\Psi}$ at the plasma boundary is prescribed, and its variation $\delta\bar{\Psi}_\Gamma = 0$. But in practical cases, the magnetic field at the plasma boundary should be matched with the vacuum magnetic field of external coils and currents, and the second term in Eq. (7) is necessary. It makes the result of minimizing the energy functional equivalent to the solution of the GSh equation with Neumann boundary condition on the boundary Γ.

A. Toroidal coordinates and Hermite representation

Our goal is to find a solution to Eq. (5) in curvilinear toroidal coordinates which then will be aligned with the magnetic field by a special iterative procedure. Having this in mind, we will assume that the functions $\bar{p}, \bar{F}$ and correspondingly P, T are given as functions of a

$$P = P(a), \quad T = T(a). \tag{8}$$

Then, in toroidal coordinates $\bar{W}$ has the form

$$\bar{W} = \frac{1}{2}\int_\Omega \left(K\bar{\Psi}'^2_a - 2N\bar{\Psi}'_a\bar{\Psi}'_\theta + M\bar{\Psi}'^2_\theta\right.$$
$$\left. -2rDP\bar{\Psi} - \frac{2D}{r}T\bar{\Psi}\right)dad\theta + \int_\Gamma \bar{\Psi}B^e_\theta d\theta. \tag{9}$$

Here, B^e_θ is a prescribed covariant component (should be distinguished from the physical one, B^e_p) of the magnetic filed, serving as the Neumann boundary condition

$$B_\theta = B^e_\theta, \quad B_\theta \equiv (\mathbf{B} \cdot \mathbf{r}'_\theta) = B_r r'_\theta + B_z z'_\theta = -K\bar{\Psi}'_a + N\bar{\Psi}'_\theta. \tag{10}$$

The functional $\bar{W}$ allows to obtain the equations for any representation of $\bar{\Psi}(a, \theta)$. The global elements in the θ direction, such as Fourier harmonics (convenient due to their natural periodicity), are an optimal choice for the plasma core when several harmonics give a good, infinitely differentiable representation. Unfortunately, they cannot be extended in a straightforward way to the edge of the separatrix limited plasmas. Modifications of Fourier representation are possible, but can have only a limited applicability. Because this is a typical case for tokamak, a local element representation is more natural for the plasma edge.

For MHD problems, including the tokamak plasma equilibrium, the Hermite finite elements are the most appropriate. Each element contains the values of the function and its first derivatives at the grid points. The 1-D Hermite representation of a function $f(a,\theta)$ at given θ inside each element $a_0 \leq a \leq a_1$, $\theta_0 \leq \theta \leq \theta_1$ is

$$f(x, y) = A_0(x)f_0(y) + A_1(x)f'_{x,0}(y) + A_2(x)f_1(y)$$
$$+ A_3(x)f'_{x,1}(y), \tag{11}$$

where x, y are the local normalized coordinates

$$x \equiv \frac{a - a_0}{h_a}, \quad h_a \equiv a_1 - a_0, \quad y \equiv \frac{\theta - \theta_0}{h_\theta}, \quad h_\theta \equiv \theta_1 - \theta_0, \tag{12}$$

and functions $A_i(x)$, organized in a vector $\vec{A}(x)$, are given by

$$\vec{A}(x) = \{A_0, A_1, A_2, A_3\},$$
$$= \{1 - 3x^2 + 2x^3, x - 2x^2 + x^3, 3x^2 - 2x^3, -x^2 + x^3\}. \tag{13}$$

In the 2-D Hermite representation, the functions $f_0(y)$, $f'_{x,0}$, $f_1(y)$, $f'_{x,1}(y)$ are represented in a similar form

$$f_0(y) = \Theta_0 f_{00} + \Theta_1 f'_{y,00} + \Theta_2 f_{01} + \Theta_3 f'_{y,01}, \tag{14}$$

$$f'_{x0}(y) = \Theta_0 f'_{x,00} + \Theta_1 f''_{xy,00} + \Theta_2 f'_{x01} + \Theta_3 f''_{xy,01}, \tag{15}$$

$$f_1(x) = \Theta_0 f_{10} + \Theta_1 f'_{y,10} + \Theta_2 f_{11} + \Theta_3 f'_{y,11}, \tag{16}$$

$$f'_{x1}(x) = \Theta_0 f'_{x,10} + \Theta_1 f''_{xy,10} + \Theta_2 f'_{x,11} + \Theta_3 f''_{xy,11} \tag{17}$$

with

$$\vec{\Theta}(y) = \{1 - 3y^2 + 2y^3, y - 2y^2 + y^3, 3y^2 - 2y^3, -y^2 + y^3\}. \quad (18)$$

The advantage of Hermite elements with respect to the linear ones (or equivalently, the finite difference numerical schemes) is that the necessary functions and their first derivatives are explicitly contained in the solutions. The simpler linear elements require the ill-determined differentiation of the solution in order to obtain the first derivatives (in our case, the magnetic fields). At the same time, with the exception of mixed derivatives, Hermite elements do not contain excessive information, which is not related to the physics of the problem, such as second derivatives. Note that the continuity of mixed derivatives at the nodes guarantees that magnetic field at these points is divergence free.

In addition to consistency with the MHD model, the attractive computational feature of Hermite elements is that in the functional they interact only with the neighboring elements, similar to the finite difference schemes. This simplifies the structure of the matrix equations determining the minimization of $\bar{W}$.

B. Vector of unknowns and matrix equations

In a compact vector form, the function $\bar{\Psi}$ inside a finite element can be represented as

$$\bar{\Psi} = \vec{Y}^T \vec{V} = \sum_{j=0}^{j<16} Y_j V^j, \quad (19)$$

with four unknowns in each of four nodes, organized in a vector of sixteen unknowns $\vec{Y}$

$$\vec{Y}^T = \{\bar{\Psi}_{00}, \bar{\Psi}'_{x,00}, \bar{\Psi}'_{y,00}, \bar{\Psi}''_{xy,00}, \quad \bar{\Psi}_{10}, \bar{\Psi}'_{x,10}, \bar{\Psi}'_{y,10}, \bar{\Psi}''_{xy,10},$$
$$\bar{\Psi}_{11}, \bar{\Psi}'_{x,11}, \bar{\Psi}'_{y,11}, \bar{\Psi}''_{xy,11}, \quad \bar{\Psi}_{01}, \bar{\Psi}'_{x,01}, \bar{\Psi}'_{y,01}, \bar{\Psi}''_{xy,01}\}. \quad (20)$$

Here, the functions $V^j(x, y)$ are given by

$$\begin{pmatrix} V^0 & V^1 & V^4 & V^5 \\ V^2 & V^3 & V^6 & V^7 \\ V^{12} & V^{13} & V^8 & V^9 \\ V^{14} & V^{15} & V^{10} & V^{11} \end{pmatrix}$$
$$= \begin{pmatrix} \Theta_0 A_0 & \Theta_0 A_1 & \Theta_0 A_2 & \Theta_0 A_3 \\ \Theta_1 A_0 & \Theta_1 A_1 & \Theta_1 A_2 & \Theta_1 A_3 \\ \Theta_2 A_0 & \Theta_2 A_1 & \Theta_2 A_2 & \Theta_2 A_3 \\ \Theta_3 A_0 & \Theta_3 A_1 & \Theta_3 A_2 & \Theta_3 A_3 \end{pmatrix}. \quad (21)$$

The choice of the geometry $r(a, \theta)$, $z(a, \theta)$ of the finite elements deserves a special consideration. The simplest one would be a quadrilateral form, with straight line sides in the r–z space. Unfortunately, this choice is inconsistent with the accuracy level of cubic polynomial representation of the unknown function $\bar{\Psi}$.

In EEC, functions $\vec{r} = \{r, z\}$ are represented using the same Hermite representation

$$\vec{r}(a, \theta) = \sum_{j=0}^{j<16} V^j(a, \theta)\vec{R}_j, \quad \vec{r}'_a = \sum_{j=0}^{j<16} V'^j_a \vec{R}_j,$$
$$\vec{r}'_\theta = \sum_{j=0}^{j<16} V'^j_\theta \vec{R}_j, \quad (22)$$

where

$$\vec{R}^T = \{\vec{r}_{00}, \vec{r}'_{a00}, \vec{r}'_{\theta00}, \vec{r}''_{a\theta00}, \quad \vec{r}_{10}, \vec{r}'_{a10}, \vec{r}'_{\theta10}, \vec{r}''_{a\theta10},$$
$$\vec{r}_{11}, \vec{r}'_{a11}, \vec{r}'_{\theta11}, \vec{r}''_{a\theta11}, \quad \vec{r}_{01}, \vec{r}'_{a01}, \vec{r}'_{\theta01}, \vec{r}''_{a\theta01}\}. \quad (23)$$

The discrete form of the energy functional can be obtained by substituting the representations of $\bar{\Psi}$, r, and z into Eq. (7)

$$\bar{W} = \sum_i^{i<N_E} \int_0^1 \int_0^1 \left[\frac{1}{2} \left(\bar{K}\bar{\Psi}'^2_y - 2\bar{N}\bar{\Psi}'_x \bar{\Psi}'_y + \bar{M}\bar{\Psi}'^2_x \right) \right.$$
$$\left. - r\bar{D}P\bar{\Psi} - \frac{\bar{D}}{r} T\bar{\Psi} \right]_i dxdy$$
$$+ \sum_{j=0}^{j<J} \int_0^1 (\bar{\Psi}B_y)_j dy = \frac{1}{2} \sum_{i=0}^{i<N_E} \vec{Y}_i^T (\mathsf{E}_i \vec{Y}_i - 2\vec{R}_i). \quad (24)$$

Here, $N_E = I \cdot J$ is the total number of finite elements, and I, J are the number of a- and θ-intervals. The summation over poloidal intervals is taken at the boundary where Neumann conditions are applied. The modified metric coefficients $\bar{K}, \bar{N}, \bar{M}, \bar{D}$ are calculated using derivatives with respect of local coordinates x, y

$$\bar{K} \equiv \frac{g_{yy}}{r\bar{D}}, \quad \bar{N} \equiv \frac{g_{xy}}{r\bar{D}}, \quad \bar{M} \equiv \frac{g_{xx}}{r\bar{D}}, \quad \bar{D} \equiv r'_y z'_x - z'_y r'_x,$$
$$g_{xx} = r'_x r'_x + z'_z z'_x, \quad g_{yy} = r'_y r'_y + z'_y z'_y, \quad g_{xy} = r'_x r'_y + z'_z z'_y. \quad (25)$$

The local matrix E_i and the local right hand side $\vec{R}_i$ on each element are given by

$$\mathsf{E}_i^{16\times16} \equiv \int_{E_i} \left\{ \bar{K}\vec{V}'_y \vec{V}'^T_y - \bar{N}(\vec{V}'_x \vec{V}'^T_y + \vec{V}'_y \vec{V}'^T_x) + \bar{M}\vec{V}'_x \vec{V}'^T_x \right\} dxdy, \quad (26)$$

$$\vec{R}_i^{1\times16} \equiv \int_{E_i} \vec{V} \left(r\bar{D}P + \frac{\bar{D}}{r} T \right) dxdy - \int_{I_i} VB_y dy, \quad (27)$$

where the superscripts specify their dimensions.

The unknown function $\bar{\Psi}$ and its derivatives are defined on vertexes of the finite elements. Accordingly, we introduce a vector with $4 \cdot I_1 \cdot J$ components, where "4" stands for $\bar{\Psi}, \bar{\Psi}'_a, \bar{\Psi}'_\theta, \bar{\Psi}''_{a\theta}$ on each vertex, and $I_1 \equiv I + 1$ is the number of vertexes in a-direction. In the following for compactness, we will consider these four elements as a single component of the global vector of unknowns $\vec{\Psi}$

$$(\vec{\Psi})_{ij} \equiv \vec{\Psi}_{ij}, \quad \vec{\Psi}_{ij} \equiv \{\bar{\Psi}_{ij}, \bar{\Psi}'_{a,ij}, \bar{\Psi}'_{\theta,ij}, \bar{\Psi}''_{a\theta,ij}\}, \quad (28)$$

where i, j are the radial and poloidal counters.

The previous matrix expression of $\bar{W}$ in Eq. (24) in terms of finite elements should be converted into the vertex based matrix form

$$\bar{W} = \frac{1}{2} \vec{\Psi}^T \cdot (\mathsf{A} \cdot \vec{\Psi} - 2\vec{R})$$

$$= \frac{1}{2} \sum_{k'=0}^{k'<JI_1} \sum_{k=0}^{k<JI_1} \vec{\Psi}_{k'}((\mathsf{A})_k^{k'} \vec{\Psi}^k - 2R^{k'}), \qquad (29)$$

where k, k' are the vertex counters

$$k \equiv I_1 j + i, \qquad (30)$$

thus, determining the enumeration of vertexes.

By differentiation of $\bar{W}$ (29) with respect to $\bar{\Psi}_{k'}$, we obtain the linear equations, representing the matrix form of the GSh equation, for the vector $\vec{\Psi}$ of Hermite element parameters of $\bar{\Psi}(a, \theta)$

$$\mathsf{A}\vec{\Psi} = \vec{R}. \qquad (31)$$

C. Boundary conditions

We consider the boundary layer limited from inside by a virtual boundary, referenced by a superscript "v," and from outside by a plasma boundary, referenced by "p" (which should be distinguished from the subscript "p," standing for poloidal components) (Fig. 1).
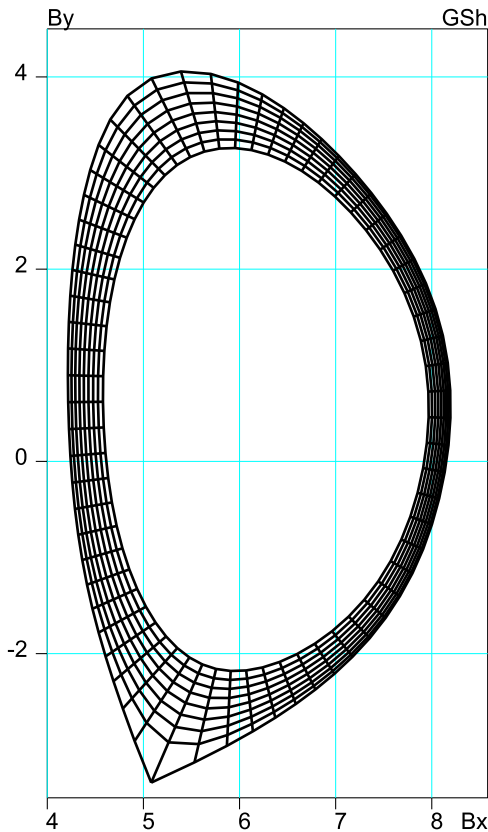

FIG. 1. Grid mesh in a boundary layer.

The case of the "fixed" plasma boundary with a specified shape corresponds to the Dirichlet boundary condition

$$\bar{\Psi}(a^p, \theta) = 0. \qquad (32)$$

In EEC, this condition is included explicitly into matrix equations (31) in a way preserving the symmetry of the matrix A.

The inner boundary of the layer is flexible. Its shape is determined by matching the core and the boundary layer solutions. Because of this, the boundary condition at the virtual boundary is of the Neumann type (10). It is automatically included into the matrix equations (31) after minimization of the quadratic form $\bar{W}$ (29).

In the case of calculation of equilibrium in an external magnetic field, the boundary condition can be derived using the virtual casing principle[27]

$$\oint_\Gamma \left[b_p(l, l') i(l') - b_p(l', l) i(l) \right] dl' = -B_p^{PFC}(l). \qquad (33)$$

Here, $l(\theta)$, is a length along the contour of the plasma cross-section, $b_p(l, l')$ is the poloidal component of magnetic field in the point (a^p, l) from a unit filament $(0 \leq \phi \leq 2\pi)$ current at the point (a^p, l'). The surface current $i(l)$ is to be determined by solving (33), and $B_p^{PFC}(l)$ is the poloidal magnetic field created by the Poloidal Field Coils (PFC). The unknown surface current $i(l)$ is related to $B_p^e(\theta)$ by the Ampere law

$$\mu_0 i = B_p^e. \qquad (34)$$

For the free-boundary equilibrium, the covariant component B_θ^e in the Neumann boundary condition (10) is then given by

$$B_\theta^e = \frac{\mu_0 i(l(\theta))}{\sqrt{r_\theta'^2 + z_\theta'^2}}, \qquad (35)$$

where $i(l)$ is the solution to Eq. (33).

D. The structure of the matrix equations

In the above specified enumeration (30), the matrix A has a structure of a block-tri-diagonal cycle matrix shown in Fig. 2.

The elementary tokens in matrix A in Fig. 2 are 4×4 matrices. These tokens compose tri-diagonal $I_1 \times I_1$ matrices. These radial matrices then compose a periodic tri-diagonal $J \times J$ matrix, corresponding to the poloidal coordinate θ.

Note that the application of Hermite elements for 3-D equilibria (e.g., in stellarator configurations) and minimization of the energy functional would result in a matrix retaining a similar block-diagonal structure. With $\bar{\Psi}, \bar{\Phi}$ functions unknown, the elementary token will be $(2 \times 8) \times (2 \times 8)$ matrices. Similarly to 2-D case, these tokens compose tri-diagonal $I_1 \times I_1$ radial matrices, which in turn compose a periodic tri-diagonal $J \times J$ matrices. As an additional layer in the structure, these poloidal matrices will compose a periodic tri-diagonal $L \times L$ matrix, corresponding to the toroidal angle ϕ.
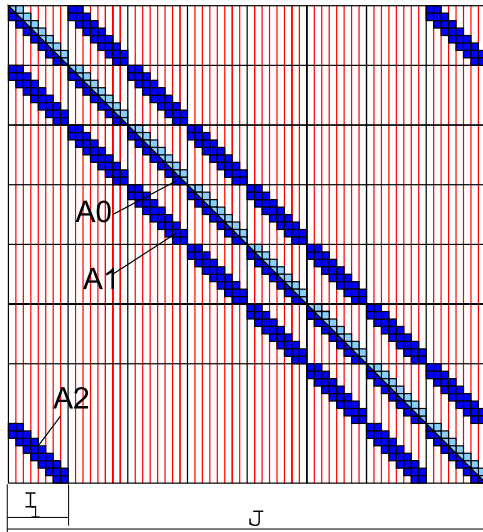
FIG. 2. Structure of the matrix A for 2-D case.

The block tri-diagonal algorithm for solving linear equations (31) was implemented into EEC as an initial approach. This version of EEC was used for debugging the code and for its verification against exact solutions to the GSh operator with prescribed right hand sides.

The important property of the energy functional is that it leads to a positive definite and symmetric matrix A. The second algorithm used in EEC takes advantage of the Cholesky decomposition of this kind of matrices

$$A = L^T \cdot D \cdot L, \tag{36}$$

where L is a lower triangular matrix with diagonal elements equal 1, and D is a diagonal matrix. Unlike the tri-diagonal algorithm, Cholesky decomposition does not involve matrix inversion. The diagonal matrix D and the unit lower triangular matrix L are calculated with the following algorithm:

$$D_i = A_{ii} - \sum_{k=0}^{k<i} L_{ik}^2 D_k,$$

$$L_{ji} = \frac{1}{D_i} \left(A_{ji} - \sum_{k=0}^{k<i} L_{jk} L_{ik} D_k \right), \quad \text{for } i < j < n. \tag{37}$$

In performing this Cholesky decomposition, EEC utilizes the block-diagonal cyclic structure of matrix A, which can be represented as a product of sparse unit-triangular matrices as it is shown in Fig. 3. After the decomposition, the resulting band matrices can be used for many right-hand sides (RHSs) of the GSh equation.

E. Modification of finite elements with the *X* point

Without loss of generality, EEC finite element grid is created from straight $\theta = $ const lines, and curved and smooth $a = $ const lines, while $r(a, \theta)$, $z(a, \theta)$ inside the elements have the Hermite representation (23). For such a grid structure, $\bar{\Psi}$ and its first and mixed derivatives are same in the common vertex of neighboring elements.

The exception is two finite elements near the X-point of the separatrix shown in Fig. 4. Because of the discontinuity of the derivatives of r, z with respect to θ at the X point, the components of the vector of unknowns in the X-point elements should be modified. In EEC, the local orthogonal coordinates u, v are introduced with the origin (r_X, z_X) in the X-point. The v coordinate is directed along the axis a

$$v = (r - r_X)c_\beta + (z - z_X)s_\beta, \quad u = (r - r_X)s_\beta - (z - z_X)c_\beta. \tag{38}$$

The vector (c_β, s_β) is the normal vector along a direction at X point

$$c_\beta \equiv \cos\beta = \frac{r'_x}{\sqrt{r'^2_x + z'^2_x}}\bigg|_X, \quad s_\beta \equiv \sin\beta = \frac{z'_x}{\sqrt{r'^2_x + z'^2_x}}\bigg|_X. \tag{39}$$

Since v is along the axis a and u, v are orthogonal, the derivative of u with respect to a and correspondingly the derivative of u with respect to x is zero

$$u'_{x,X} = 0. \tag{40}$$

The derivatives with respect to the local coordinates u, v can be derived by the chain rule and together with the condition $u'_{x,X} = 0$ it gives
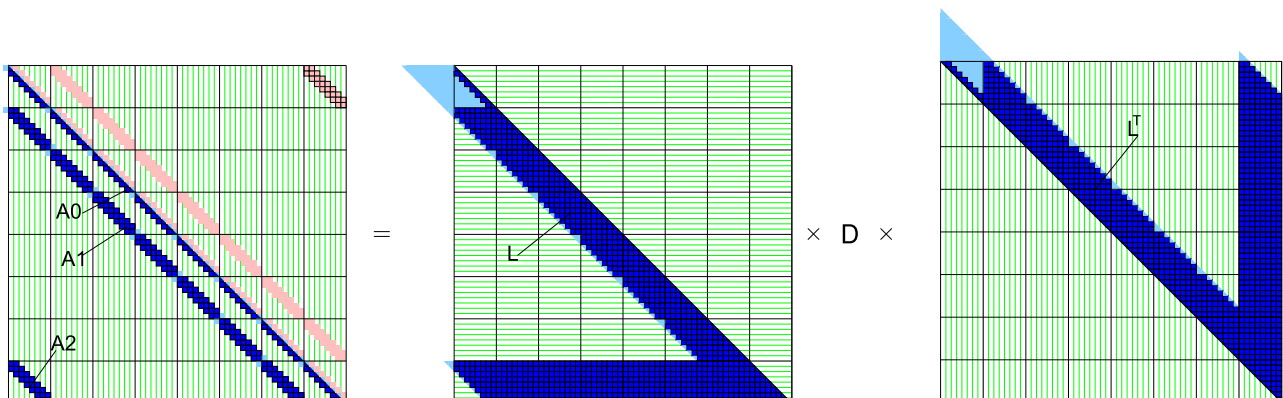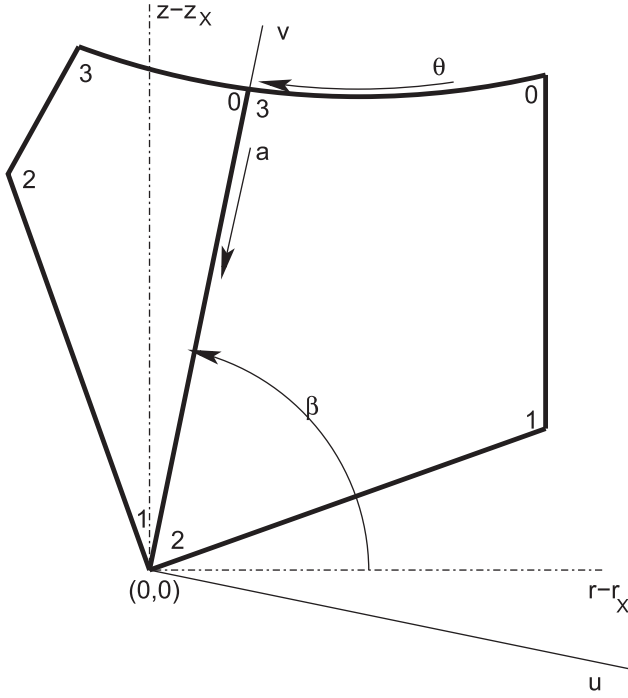

FIG. 3. The block-band structure of Cholesky matrices. Light blues and empty spaces represent zeros.

FIG. 4. Two finite elements and the local coordinates u, v near the X-point.

$$\bar{\Psi}'_x = v'_x \bar{\Psi}'_v, \quad \bar{\Psi}'_y = v'_y \bar{\Psi}'_v + u'_y \bar{\Psi}'_u$$

$$\bar{\Psi}''_{xy} = v''_{xy} \bar{\Psi}'_v + u''_{xy} \bar{\Psi}'_u + \frac{v'_y}{v'_x} \bar{\Psi}''_{xx} + u'_y v'_x \bar{\Psi}''_{vu}. \quad (41)$$

The second derivatives $\bar{\Psi}_{xx}$, necessary for representing $\bar{\Psi}''_{xy}$ in the X-point, can be expressed using the Hermite representation in terms of $\bar{\Psi}$ and $\bar{\Psi}_v$ in the vertexes

$$\begin{pmatrix} \bar{\Psi}''_{xx}{}^0 & \bar{\Psi}''_{xx}{}^1 \\ \bar{\Psi}''_{xx}{}^3 & \bar{\Psi}''_{xx}{}^2 \end{pmatrix} = \begin{pmatrix} \bar{\Psi}^0 & v'_x{}^0 \bar{\Psi}'_v{}^0 & \bar{\Psi}^1 & v'_x{}^1 \bar{\Psi}'_v{}^1 \\ \bar{\Psi}^3 & v'_x{}^3 \bar{\Psi}'_v{}^3 & \bar{\Psi}^2 & v'_x{}^2 \bar{\Psi}'_v{}^2 \end{pmatrix}$$
$$\times \begin{pmatrix} -6 & 6 \\ -4 & 2 \\ 6 & -6 \\ -2 & 4 \end{pmatrix}, \quad (42)$$

where enumeration in the superscripts is shown in Fig. 4. For the left element there, the X-point has the index "1," while for the right one it is indexed by "2."

The derivatives $\bar{\Psi}_v$, $\bar{\Psi}_u$, $\bar{\Psi}_{vu}$ in the X-point are continuous, and, thus, can be used as a set of unknowns in this vertex. The constraints $\bar{\Psi}_v = 0$, $\bar{\Psi}_u = 0$, valid for the X-point, can be added to the boundary conditions of the matrix form of the GSh equation.

III. ADAPTIVE GRID

The final goal of equilibrium calculations in both ESC and EEC is to make the calculation grid conformal to the magnetic surfaces. Here, we describe the specific details of this procedure in EEC.

A. Adjust grid to $\bar{\Psi} = const$ lines

At each iteration step, the solution $\bar{\Psi}$ obtained by EEC is dependent on θ along the $a = const$ grid. There are two

approaches used to adjust the grid to be aligned with magnetic surfaces. One is to search the $\bar{\Psi} = const$ points to update the grid. EEC uses another algorithm, the same as implemented in ESC. This algorithm provides a fast Newton scheme for adjusting the grid.

The poloidal flux can be expressed as the sum of the average part and an oscillatory perturbation, which is considered to be small

$$\bar{\Psi} = \bar{\Psi}_0(a) + \psi(a, \theta), \quad \bar{\Psi}_0(a) \equiv \frac{1}{2\pi} \oint \bar{\Psi}(a, \theta) d\theta. \quad (43)$$

The a-coordinate is massaged by

$$a \to a + \xi, \quad (44)$$

which leads to an updated grid

$$r \to r + r'_a \xi + r'_\theta \sigma, \quad z \to z + z'_a \xi + z'_\theta \sigma. \quad (45)$$

The terms $r'_\theta \sigma$, $z'_\theta \sigma$ are used for keeping the straight line geometry of $\theta = const$ lines, when their origins at the virtual boundary move during iterations (as it is explained a little bit later). The oscillatory part $\tilde{\xi}$ of displacement ξ is calculated by

$$\bar{\Psi}(a + \xi, \theta) = \bar{\Psi}_0(a + \xi) + \psi(a, \theta) = \text{const}, \quad (46)$$

$$\bar{\Psi}_0(a) + \bar{\Psi}'_0 \xi + \psi(a, \theta) = \text{const}, \quad \tilde{\xi} = -\frac{\psi}{\bar{\Psi}'_0}. \quad (47)$$

The choice of the averaged part ξ_0 depends on the choice of radial coordinate a, used in simulations. Some of them are explained below.

B. Toroidal cross-section

One useful choice of the radial coordinate is related to the area of toroidal cross-section

$$\bar{S} \equiv \frac{1}{2\pi} \int dS = \frac{1}{2\pi} \int dr dz = \frac{1}{2\pi} \oint r z'(\theta) d\theta, \quad \bar{S}'_a = \langle D \rangle \quad (48)$$

and $a \equiv \sqrt{\bar{S}/\bar{S}^p}$ can be considered as a label in radial direction. Here, $\langle D \rangle$ is the averaged part of D

$$\langle D \rangle \equiv \frac{1}{2\pi} \int D d\theta. \quad (49)$$

Then, ξ_0 can be obtained by the Newton scheme

$$s(a_i + \xi_0) = s(a_i) + s'(a_i) \xi_0 = a_i, \quad \xi_0 = \frac{a_i - s_i}{s'_i}, \quad (50)$$

where

$$s(a) \equiv \sqrt{\frac{\bar{S}(a)}{\bar{S}^p}} \quad (51)$$

is calculated at every iteration after adjustment of the coordinates with respect to the oscillatory $\tilde{\xi}$, and a_i is uniform in the boundary layer

$$a_i = a^v + i\Delta a, \quad \Delta a = \frac{1 - a^v}{I}. \tag{52}$$

C. Volume inside magnetic surfaces

The volume inside a magnetic surface is given by

$$\bar{V} = \frac{1}{4\pi^2}\int dV = \frac{1}{4\pi^2}\int \oint r\, dr\, dz\, d\phi = \frac{1}{4\pi}\oint r^2 z'(\theta)\, d\theta.$$
$$\bar{V}'_a = \langle rD \rangle. \tag{53}$$

Similarly to the previous subsection, we consider $a \equiv \sqrt{\bar{V}/\bar{V}^p}$. Then the formula for ξ_0 is

$$\xi_0 = \frac{a_i - v_i}{v'_i}, \tag{54}$$

where $v(a)$ is recalculated at every iteration as

$$v(a) \equiv \sqrt{\frac{\bar{V}(a)}{\bar{V}^p}}. \tag{55}$$

This choice of a is useful for equilibria used in transport simulations and is universal for all toroidal confinement configurations (tokamaks, stellarators, reverse filed pinches, spheromaks, field reversed configurations).

D. Toroidal induction

The toroidal induction is defined as

$$\bar{L} = \frac{1}{2\pi}\int \frac{dr\, dz}{r} = \frac{1}{2\pi}\oint \log\frac{1}{r}z'(\theta)\, d\theta, \quad \bar{L}'_a = \left\langle \frac{D}{r} \right\rangle. \tag{56}$$

Applying the same consideration as before for $a \equiv \sqrt{\bar{L}/\bar{L}^p}$, we have the formula for ξ_0

$$\xi_0 = \frac{a_i - l_i}{l'_i}, \quad l(a) \equiv \sqrt{\frac{\bar{L}(a)}{\bar{L}^p}}. \tag{57}$$

In the strong magnetic field approximation for tokamak (the so-called reduced MHD approximation), the vacuum toroidal induction as a radial coordinate approximates the toroidal flux and is useful as its simpler substitution for calculations of flux conserving equilibria and in vertical disruption simulations.

E. Toroidal flux

The toroidal flux of the magnetic field is given by

$$\bar{\Phi}(a) \equiv \frac{1}{2\pi}\int B_\phi dS = \int_0^a \bar{F}L'\, da = \bar{\Phi}(a^v) + \int_{a^v}^a \bar{F}L'\, da, \tag{58}$$

where

$$B_\phi = \frac{\bar{F}(a)}{r} \tag{59}$$

and $\bar{F}$ is calculated by

$$\bar{F}^2 = \bar{F}^{p,2} - 2\int_a^{a^p} T\bar{\Psi}'_0\, da. \tag{60}$$

The radial coordinates is determined as $a = \sqrt{\bar{\Phi}/\bar{\Phi}^p}$, and the formula for the displacement ξ_0 is given by

$$\xi_0 = \frac{a_i - \varphi_i}{\varphi'_i}, \quad \varphi_i \equiv \sqrt{\frac{\bar{\Phi}}{\bar{\Phi}^p}}. \tag{61}$$

This choice of a is natural for generating flux conserving equilibria and simulations of vertical disruptions in tokamaks.

The use of toroidal flux instead of the geometry-based radial coordinates introduces some complications in interfacing the core and the edge codes: in order to calculate the toroidal flux at the edge layer, EEC needs the information of Φ at the virtual boundary, which has to be provided by ESC. In its turn the core code ESC needs $\bar{F}$, which is provided by EEC.

Note that the frequently used poloidal flux $\sqrt{\Psi}$ as a radial coordinate represents the worse case for both convergence of equilibrium codes and their interfacing with other simulations.

F. Shift of the grid points to the new $\theta = const$ lines from the virtual boundary points

EEC communicates with ESC through a virtual boundary. Using the virtual boundary as the updated inner boundary, EEC needs to shift the grid points to the new $\theta = const$ lines

$$r = r^v + \Delta rt, \quad z = z^v + \Delta zt, \quad \Delta r \equiv r^p - r^v, \quad \Delta z \equiv z^p - z^v, \tag{62}$$

where $\mathbf{r}^v$ and $\mathbf{r}^p$ are pairs of points at the virtual and plasma boundaries, correspondingly.

The total displacements of r, z now consist of moving the original grid point along the radial direction and then along the θ direction

$$r = r(a,\theta) + r'_a\xi + r'_\theta\sigma, \quad z = z(a,\theta) + z'_a\xi + z'_\theta\sigma, \tag{63}$$

where $r(a, \theta)$, $z(a, \theta)$ are the current position of the grid point. Then, we can find the expressions for t, σ

$$t = \frac{(r - r^v + r'_a\xi)z'_\theta - (z - z^v + z'_a\xi)r'_\theta}{z'_\theta\Delta r - r'_\theta\Delta z},$$
$$\sigma = \frac{(r - r^v + r'_a\xi)\Delta z - (z - z^v + z'_a\xi)\Delta r}{z'_\theta\Delta r - r'_\theta\Delta z}. \tag{64}$$

Here, for compactness, $r = r(a, \theta)$, $z = z(a, \theta)$.

G. Adjustment of the derivatives of r,z

A better alignment of coordinates a, θ to the magnetic field can be achieved by aligning not only the grid points with $\bar{\Psi}(a,\theta) = const$ lines but also by an additional adjustment of the first derivatives of r, z. In their Hermite
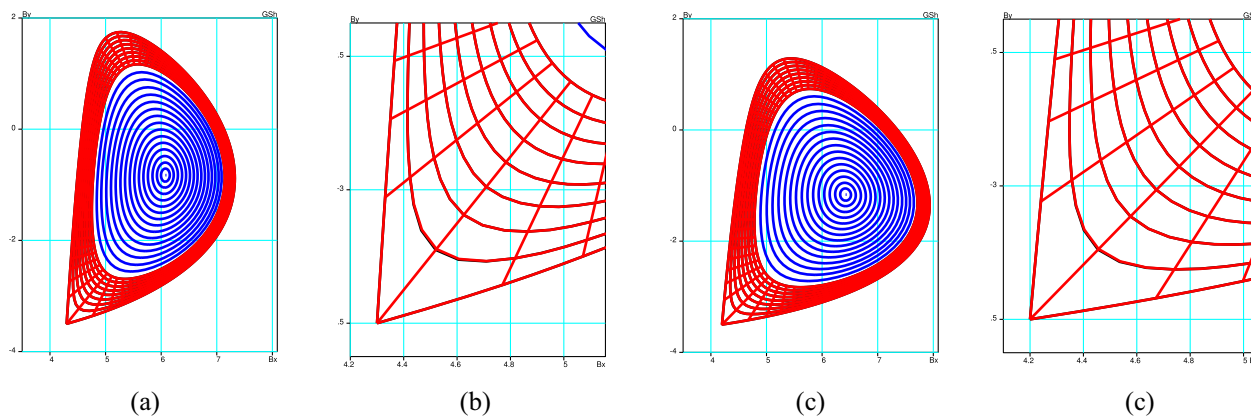
FIG. 5. Test Soloviev equilibria (a) $R_X = 4.5$, $Z_X = -3.5$, (b) expanded (a) near the X-point, (c) $R_X = 4.2$, $Z_X = -3.5$ (d) Expanded (c) near the X-point.

representation, the first and mixed derivatives are free parameters. At the same time, the desirable condition that $\bar{\Psi} = \bar{\Psi}(a)$ requires fulfillment of certain conditions

$$r'_a = \bar{r}'_{\bar{\Psi}} \bar{\Psi}'_a, \quad z'_a = \bar{z}'_{\bar{\Psi}} \bar{\Psi}'_a. \tag{65}$$

Now, assuming that at some $\theta = \theta_0$ line, the dependence $\bar{\Psi} = \bar{\Psi}_{\theta=\theta_0}(a)$ is chosen, the corrected Hermite first derivatives $\bar{r}'_a, \bar{z}'_a$ at other θ's can be calculated as

$$\bar{r}'_a \equiv r'_{\bar{\Psi}} \bar{\Psi}'_{\theta=\theta_0}(a) = \bar{a}'_a r'_a(a, \theta), \quad \bar{z}'_a = \bar{a}'_a z'_a(a, \theta),$$
$$\bar{a}'_a \equiv \frac{\bar{\Psi}'_{\theta=\theta_0}(a)}{\bar{\Psi}'_a(a, \theta)}, \tag{66}$$

where $r'_a(a, \theta), z'_a(a, \theta)$ are derivatives from the current iteration.
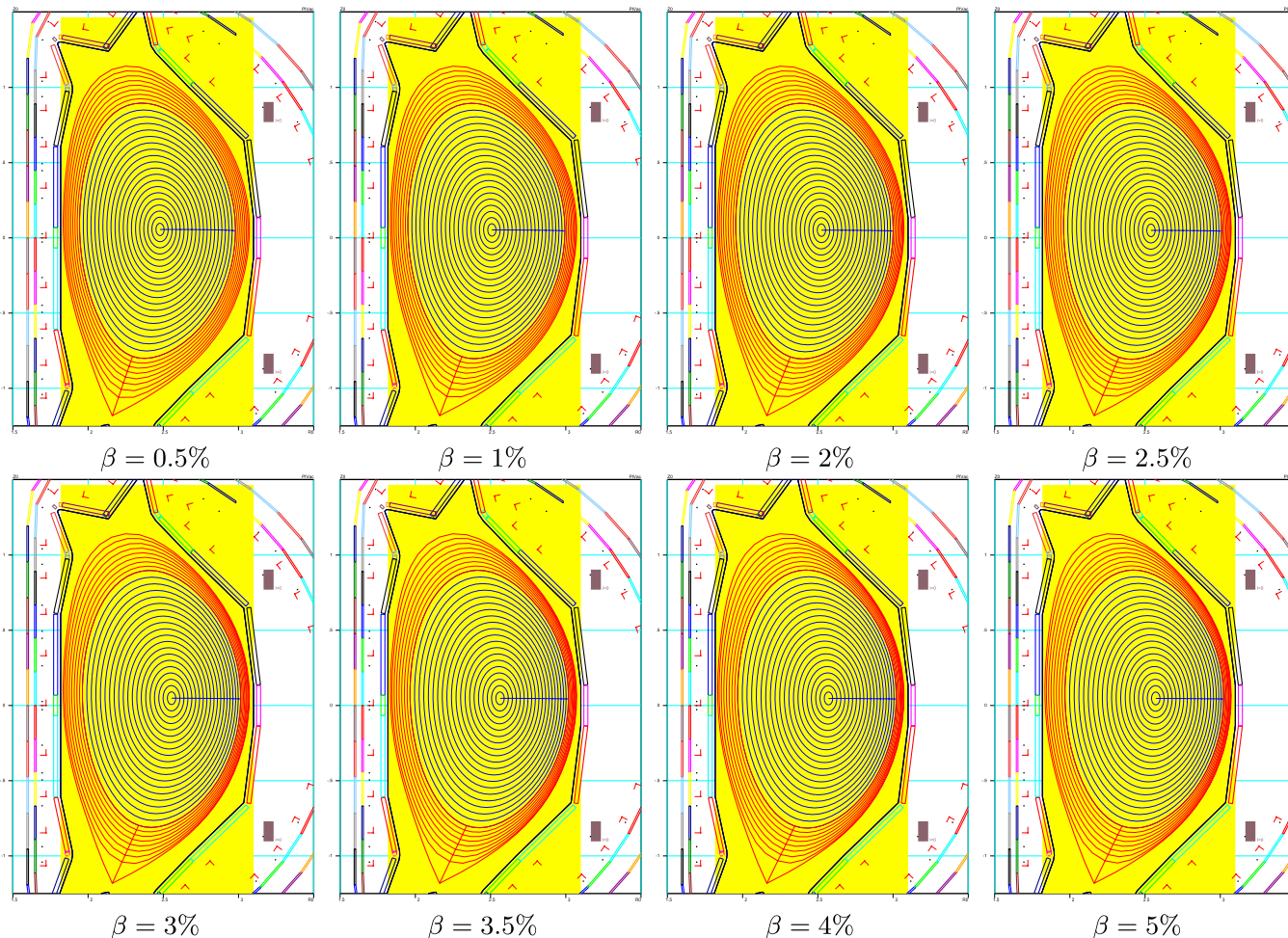
The poloidal derivatives are updated by

FIG. 6. EAST configuration.

$$\bar{r}'_\theta(a,\theta) = r'_\theta - r'_a(a,\theta)\bar{a}'_\theta, \quad \bar{z}'_\theta(a,\theta) = z'_\theta - z'_a(a,\theta)\bar{a}'_\theta,$$

$$\bar{a}'_\theta = \frac{\bar{\Psi}'_\theta}{\bar{\Psi}'_{\theta=\theta_0}(a)}, \tag{67}$$

and the mixed derivatives by

$$\bar{r}''_{a\theta}(a,\theta) = \frac{r''_{a\theta} - \bar{r}'_a(a,\theta)\bar{a}''_{a\theta}}{\bar{a}'_a} - \bar{r}''_{aa}(a,\theta)\bar{a}'_\theta, \tag{68}$$

$$\bar{a}''_{a\theta} = \frac{\bar{\Psi}''_{a\theta} - \bar{\Psi}''_{aa,\theta=\theta_0}(a)\bar{a}'_\theta\bar{a}'_a}{\bar{\Psi}'_{\theta=\theta_0}(a)}. \tag{69}$$

The above written adjustment of derivatives makes the grid geometry representation consistent with the accuracy of Hermite representation of the solution.

In the presence of an X-point at the plasma boundary, the derivative $\bar{a}'_a$ would be infinite at the X-point, if the coordinate a along $\theta = \text{const}$ represents a regular length, without special modifications. Accordingly, inside two finite elements adjacent to the X-point, $\bar{\Psi}$ remains function of both coordinates $\bar{\Psi} = \bar{\Psi}(a,\theta)$. This exception from flux coordinates is accepted in EEC at the moment, while a modification of the Hermite representation, which will remove this exception, is under development.

H. Matching core and boundary solutions

The solutions from the core equilibrium code ESC and the boundary code EEC are matched across a virtual boundary $a = a^v$, whose geometry is adjusted in order to provide

$$\bar{\Psi}^{ESC}(a^v,\theta) = \bar{\Psi}^{EEC}(a^v,\theta), \quad B_\theta^{ESC}(a^v,\theta) = B_\theta^{EEC}(a^v,\theta), \tag{70}$$

where B_θ is the covariant component of magnetic field. In both codes, the θ coordinate is the same at the virtual boundary and both conditions are not sensitive to the different choice of radial coordinate a in the core and the boundary regions.

At every iteration, the core equilibrium code ESC provides a fundamental set of internal solutions to the GSh equation at $a = a^v$. Then, the EEC code calculates their continuation with their own $B_\theta(\theta)$ to the boundary layer. The boundary condition at the plasma-vacuum interface specifies a unique combination $\bar{\Psi}^{EEC}(a,\theta)$ of fundamental solutions. The same combination determines the core solution in ESC. In particular, the shift ξ^v of the virtual boundary $a^v \to a^v + \xi^v$ is given by

$$\xi^v(\theta) = -\frac{\bar{\Psi}(a^v,\theta) - \bar{\Psi}_0(a^v)}{\bar{\Psi}'_0(a^v)}. \tag{71}$$
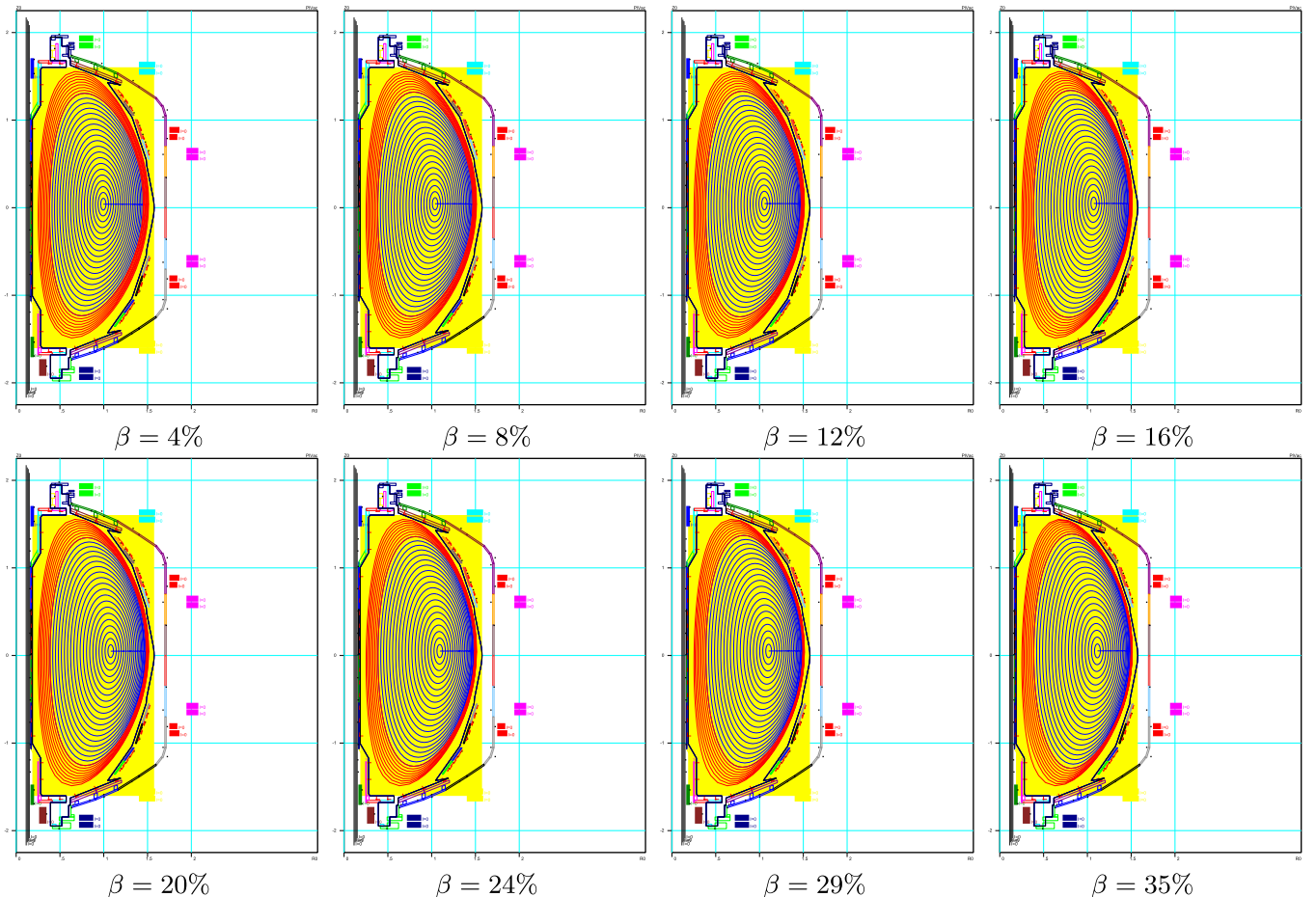

FIG. 7. NSTX configuration: smooth boundary.

IV. NUMERICAL EXAMPLES

Here, we present examples of simulations by ESC-EEC of exact analytical, so called, Soloviev equilibria and the plasma equilibria for different tokamak configurations. In all these cases, the plasma cross-section was subdivided on core and boundary layer regions. For Soloviev equilibria, the radial coordinate in both codes is the square root from the normalized area $a = \sqrt{S/S_{total}}$ of toroidal cross-section $\phi = $ const of magnetic surfaces. In the following examples of plasma equilibria in tokamaks, the square root from toroidal flux (typically used in transport simulations) is used as the radial coordinate $a = \sqrt{\Phi/\bar{\Phi}_{total}}$. The core was defined by the condition $a = 0.8$. In the boundary layer, the radial grid of EEC is twice denser than in the core.

The core plasma equilibrium (the blue regions in the figures below) is calculated by ESC with total 13 $\cos m\theta$, $\sin m\theta$ Fourier harmonics. The edge equilibrium (the red region in the figures below) is calculated by EEC using Hermite elements. The Neumann boundary condition is satisfied at the virtual boundary.

The linearized form of GSh equation,[12] used in ESC, is not yet implemented in EEC. The code takes about 5–6 iterations (0.2 s each on a single core Intel Xeon CPU 3.20 GHz processor) to find an equilibrium from scratch with the accuracy 10^{-6} in surface displacement ξ and 2–3 iterations in the sequence of evolving equilibria.

A. Soloviev test equilibria

The Soloviev equilibria[28] in Fig. (5) can be described (in Shafranov's form[29]) by a simple formula

$$\bar{\Psi} = \left(1 - \frac{r^2}{R_2^2} - \frac{z^2}{Z_2^2}\right)(r^2 - R_1^2) + A(3r^2 - 4z^2)r^2 z + Br^2 \tag{72}$$

corresponding to constant functions P, T in the current density distribution (6)

$$P = \frac{2}{Z_2^2} + \frac{8}{R_2^2}, \quad T = -\frac{2R_1^2}{Z_2^2}. \tag{73}$$

The first term in the right hand side describes a configuration with closed magnetic surfaces between R_1 and R_2 and between $-Z_2$ and Z_2. Its shape is given by intersection of a cylinder ($R_1 = 4.5$ in Fig. (5)) and an ellipsoid of rotation ($R_2 = 8$, $Z_2 = 4.43$). We added to it two simple additional terms in order to generate single null configurations with a specified position of the X-point. Note, that the separatrix in the Soloviev type of equilibria does not corresponds to the free boundary tokamak equilibria. In particular, the angle between the separatrix branches in the X-point is different from the necessary 90°.

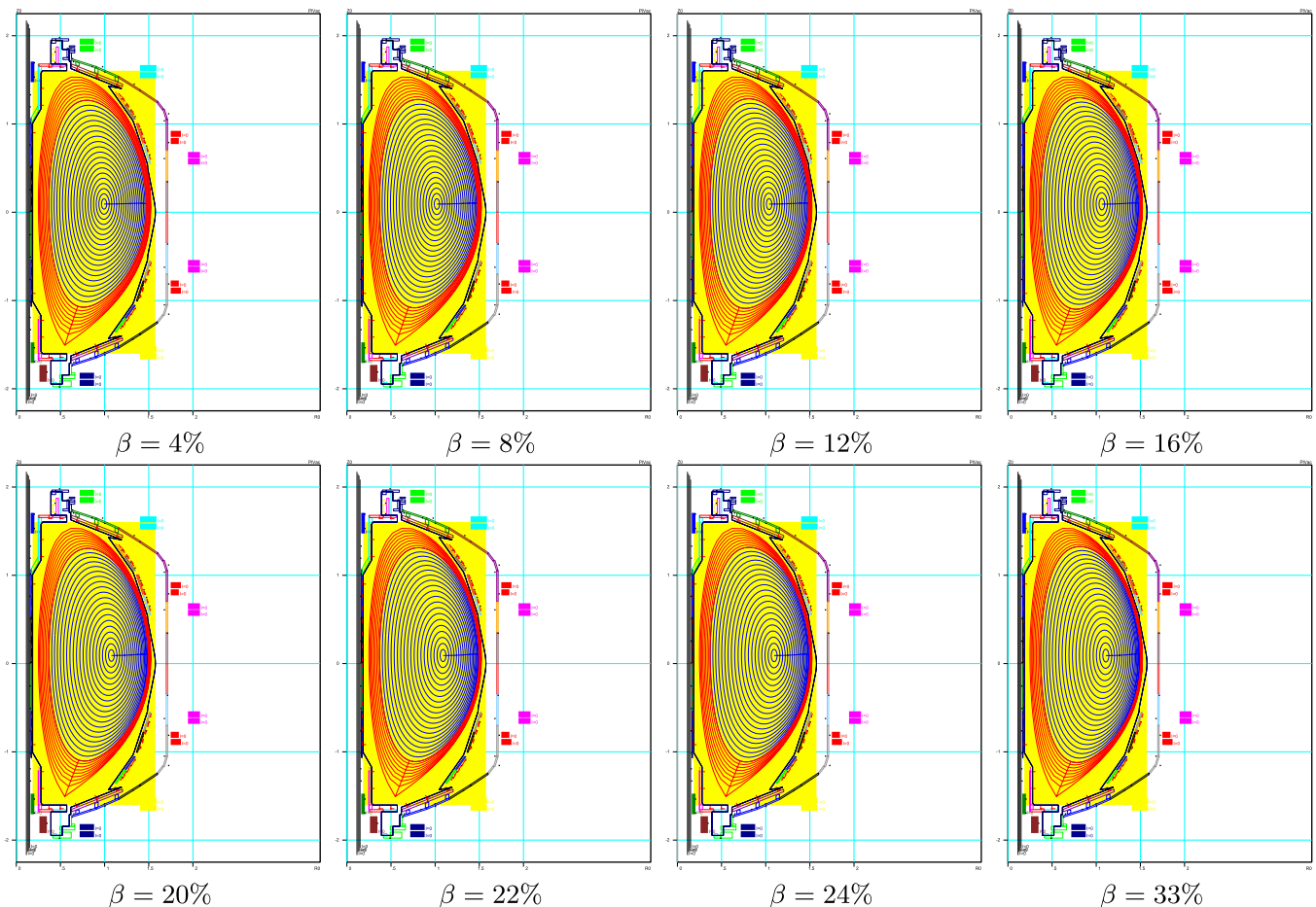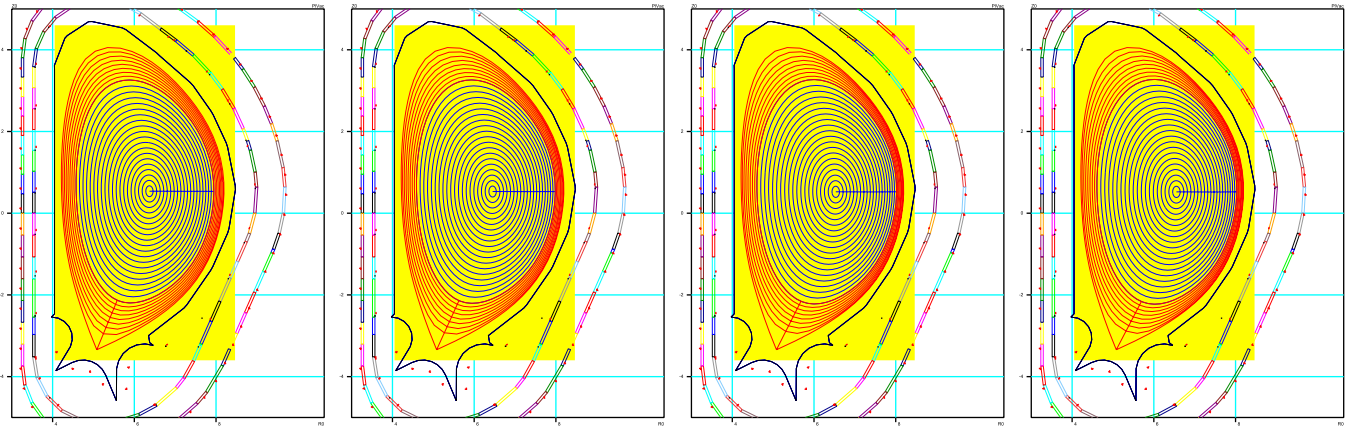|	
$\beta = 4\%$	$\beta = 8\%$
$\beta = 12\%$	$\beta = 16\%$
$\beta = 20\%$	$\beta = 22\%$
$\beta = 24\%$	$\beta = 33\%$

FIG. 8. NSTX configuration: with separatrix.

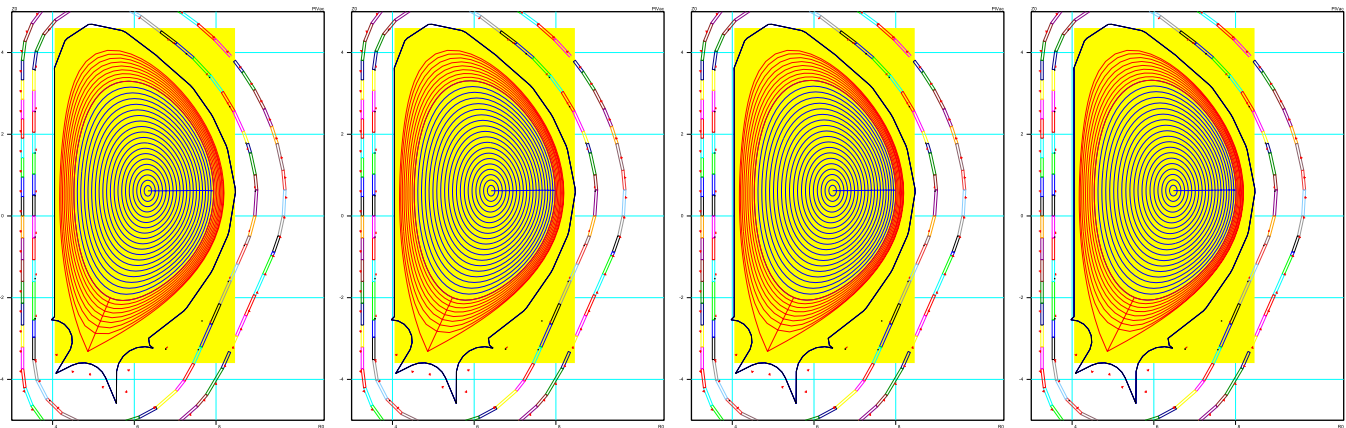FIG. 9. ITER configuration: Low triangularity case.

FIG. 10. ITER configuration: High triangularity case.

Magnetic surfaces from both numerical (from ESC-EEC, blue and red) and analytical (72, black) solutions are plotted in Fig. 5. The black color is essentially invisible on these plots. Its presence can be traced in the expanded views Figs. 5(b) and 5(c), where analytic surfaces apparently deviate from numerical (still most probably because of the plotting routine, rather than accuracy of simulations which is 10^{-6} in $\bar{\Psi}$ and 10^{-8} in ξ convergence).

B. Examples of EAST equilibria

Fig. 6 shows EAST[30,31] configurations with the major radius $R = 2$ m, and the minor radius $a = 0.5$ m with different β (which is the ratio of the volume average plasma pressure, to the magnetic pressure of the vacuum toroidal field at the major radius). As it is shown in the figures, the magnetic surfaces shift to the low field side when the plasma pressure increases.

C. Examples of NSTX equilibria

Fig. 7 shows the low aspect ratio NSTX[32] configurations with the major radius $R = 0.88$ m and the minor radius $a = 0.65$ m. The high-beta equilibria are shown with a plasma boundary with no X-point. The codes work smoothly even with a very large deformation of finite elements and cease to converge only at $\beta = 35\%$.

The examples in Fig. 8 show the equilibria of NSTX configuration with a separatrix. Similarly to the previous case, the codes can reproduce a very high-beta equilibria.

D. Examples of ITER equilibria

Fig. 9 shows the ITER like low triangularity configurations with different β, and Fig. 10 shows similar high triangularity cases.

V. SUMMARY

The creation of the edge equilibrium code, based on adaptive grids with sufficient accuracy of representing the solution using Hermite finite elements, fills the remaining gap in existing numerical codes for equilibrium tokamak problems. The code system *ESC–EEC* has acquired unmatched abilities in addressing the equilibrium problems.

The more mature part of the system, ESC,[12] covers almost all prominent formulations of equilibrium problems. Its Fourier harmonics based solver provides fast solution with the possibility of using 6 types of radial coordinates (with vertical semi-axis b and $\sqrt{\bar{\Psi}}$ in addition to those mentioned earlier for EEC) and, at least, a dozen of choices of input profiles for the GSh equation. The code is interfaced (using shared memory) with the ASTRA[33] transport simulation code, and fast speed of equilibrium generation allows to

call it at every time step in transport simulations. The perturbative nature of solving the GSh equation provides the unique ability of variance (sensitivity) analysis of equilibrium reconstruction and assessment of vertical stability. The separate r–z equilibrium solver routine of ESC for the free boundary problems, in addition to conventional capacities, includes equilibrium reconstruction of both conventional tokamak plasma and coaxial helicity injection configurations[34] with open field lines. The problem of accounting the eddy currents in reconstruction is resolved in ESC using the response function technique and a special calibration of the devices.[35] This technique has allowed to extract the energy confinement time from equilibrium reconstruction in the CDX-U tokamak[36] where the discharge time is comparable with the decay time of eddy currents. The free-boundary r–z solver of ESC is linked with the flux coordinate solver and generates the plasma boundary for it. But, this solver could not handle properly the cases with the plasma limited by a separatrix.

With EEC developed, the flux coordinate equilibria of ESC are extended up to the plasma separatrix. It is straightforward to extend the unique capacities of ESC to EEC, which, as already described, can use 4 kinds of radial coordinates. Although at the moment not all combinations of the input profiles are implemented in EEC, in addition to the classical GSh P, T profiles in the right hand side, EEC can use plasma pressure and $\bar{\Psi}(a)$ as the input for equilibrium. This makes the code ready for interfacing with transport simulation codes and, e.g., for reproducing flux conserving equilibria.

EEC should be able to work in flux coordinates as a free boundary code. The boundary conditions Eqs. (33)–(35) at the plasma-vacuum interface are not different from the matching conditions at the virtual boundary, but this mode of operation has not been tested yet. It will be fully developed in conjunction with simulations of the vertical disruptions on EAST tokamak, where the Hiro currents[37] were measured for the first time.

The extension of other capacities of ESC to EEC and utilization of GPU (Graphic Processing Units) for calculations of finite elements and Cholesky decomposition can enhance the speed of calculations to the to tens of milliseconds per iteration and will make the code system suitable for the Real Time Forecast of tokamak discharges. This emerging objective requires fast equilibrium reconstruction together with fast simultaneous transport simulations and variance analysis.[6] Another important topic is the simulation of vertical disruptions and wall touching vertical mode[37] where the use of adaptive grids is necessary for reproducing the Hiro currents.

ACKNOWLEDGMENTS

[1]L. L. Lao, H. St. John, R. D. Stambaugh, A. G. Kellman, and W. Pfeiffer, Nucl. Fusion **25**, 1611 (1985).
[2]L. L. Lao, J. R. Ferron, R. J. Groebner, W. Howl, H. St. John, E. J. Strait, and T. S. Taylor, Nucl. Fusion **30**, 1035 (1990).
[3]R. Albanese and F. Villone, Nucl. Fusion **38**, 723 (1998).
[4]R. Albanese, M. Mattei, and F. Villone, Nucl. Fusion **44**, 999 (2004).
[5]L. E. Zakharov, E. L. Foley, F. M. Levinton, and H. Y. Yuh, Plasma Phys. Rep. **34**, 173 (2008).
[6]L. E. Zakharov, J. Lewandowski, E. L. Foley, F. M. Levinton, H. Y. Yuh, V. V. Drozdov, and D. C. McDonald, Phys. Plasmas **15**, 092503 (2008).
[7]E. L. Foley, F. M. Levinton, H. Y. Yuh, and L. E. Zakharov, Nucl. Fusion **48**, 085004 (2008).
[8]V. Parail, R. Albanese, R. Ambrosino, J.-F. Artaud, K. Besseghir, M. Cavinato, G. Corrigan, J. Garcia, L. Garzotti, Y. Gribov, F. Imbeaux, F. Koechl, C. V. Labate, J. Lister, X. Litaudon, A. Loarte, P. Maget, M. Mattei, D. McDonald, E. Nardon, G. Saibene, R. Sartori, and J. Urban, Nucl. Fusion **53**, 113002 (2013).
[9]R. R. Khairutdinov and V. E. Lukash, J. Comput. Phys. **109**, 193 (1993).
[10]S. H. Kim, J. F. Artaud, V. Basiuk, V. Dokuka, R. R. Khayrutdinov, J. B. Lister, and V. E. Lukash, Plasma Phys. Controlled Fusion **51**, 105007 (2009).
[11]T. Takeda and S. Tokuda, J. Comput. Phys. **93**, 1 (1991).
[12]L. E. Zakharov and A. Pletzer, Phys. Plasmas **6**, 4693 (1999).
[13]A. Pataki, A. J. Cerfon, J. P. Freidberg, L. Greengard, and M. O'Neil, J. Comput. Phys. **243**, 28 (2013).
[14]P. Vabishevich, L. M. Degtyarev, and A. P. Favorski, Sov. J. Plasma Phys. **4**, 554 (1978).
[15]L. M. Degtyarev and V. V. Drozdov, Comput. Phys. Rep. **2**, 341 (1985).
[16]L. M. Degtyarev and V. V. Drozdov, Int. J. Mod. Phys. C **2**, 30 (1991).
[17]S. P. Hirshman and S. C. Jardin, Phys. Fluids **22**, 731 (1979).
[18]S. P. Hirshman, D. K. Lee, F. M. Levinton, S. H. Batha, M. Okabayashi, and R. M. Wieland, Phys. Plasmas **1**, 2277 (1994).
[19]S. P. Hirshman and J. Breslau, Phys. Plasmas **5**, 2664 (1998).
[20]L. L. Lao, S. P. Hirshman, and R. M. Wieland, Phys. Fluids **24**, 1431 (1981).
[21]L. L. Lao, Comput. Phys. Commun. **31**, 201 (1984).
[22]P. G. Ciarlet, *The Finite Element Method for Elliptic Problems* (North-Holland, Amsterdam, 1978).
[23]O. C. Zienkiewicz and R. L. Taylor, *The Finite Element Method*, 4th ed. (McGraw-Hill, London, 1989), Vol. 2.
[24]H. Lütjens, A. Bondesons, and A. Roy, Comput. Phys. Commun. **69**, 287 (1992).
[25]G. T. A. Huysmans, J. P. Goedbloed, and W. Kerner, in *Proceedings of CP90 Conference on Computational Physics Proceedings* (World Scientific Publ. Co., Singapore, 1991), p. 371.
[26]V. D. Shafranov, *Reviews of Plasma Physics* (Consultant Bureau, New York, 1966), Vol. 2, p. 103.
[27]L. E. Zakharov and V. D. Shafranov, Nucl. Fusion **12**, 599 (1972).
[28]L. S. Soloviev, Sov. Phys. JETP **26**, 400 (1966).
[29]L. E. Zakharov and V. D. Shafranov, *Reviews of Plasma Physics* (Consultants Bureau, New York, 1981), Vol. 11, p. 153.
[30]Y. X. Wan, J. Li, P. Weng, and EAST, GA, PPPL Team, in Proceedings of the 21st International Conference on Fusion Energy, Chengdu, China (IAEA, Vienna, 2006), see http://www-pub.iaea.org/MTCD/Meetings/FEC2006/ov1-1.pdf.
[31]B. N. Wan, EAST and HT-7 Teams, and International Collaborators, Nucl. Fusion **49**, 104011 (2009).
[32]J. E. Menard, S. C. Jardin, S. M. Kaye, C. E. Kessel, and J. Manickam, Nucl. Fusion **37**, 595 (1997).
[33]G. Pereverzev, P. Yushmanov, A. Dnestrovskii, A. Polevoi, K. Tarasjan, and L. Zakharov, *ASTRA: An Automatic System for Transport Analysis in a Tokamak* (Max-Planck-Institute fur Plasmaphysik, 1991), IPP 5/42.
[34]R. Raman, T. R. Jarboe, B. A. Nelson, W. T. Hamp, V. A. Izzo, R. G. O'Neill, A. J. Redd, P. E. Sieck, and R. J. Smith, Phys. Plasmas **11**, 2565 (2004).
[35]J. Spaleta, L. E. Zakharov, R. Majeski, and T. Gray, Rev. Sci. Instrum. **77**, 10E305 (2006).
[36]R. Majeski, R. Doerner, T. Gray, R. Kaita, R. Maingi, D. Mansfield, J. Spaleta, V. Soukhanovskii, J. Timberlake, and L. E. Zakharov, Phys. Rev. Lett. **97**, 075002 (2006).
[37]L. E. Zakharov, S. A. Galkin, S. N. Gerasimov, and JET-EFDA Contributors, Phys. Plasmas **19**, 055703 (2012).