

A Coupled FEM-BEM Approach for the Solution of the Free-Boundary Axi-Symmetric Plasma Equilibrium Problem

M. Bonotto^{1,2,*}, D. Abate¹, P. Bettini^{1,3} and F. Villone⁴

¹ *Consorzio RFX (CNR, ENEA, INFN, Università di Padova, Acciaierie Venete SpA), C.so Stati Uniti 4, 35127 Padova, Italy.*

² *INFN-LNL, Legnaro, Italy.*

³ *Dipartimento di Ingegneria Industriale (DII), Università di Padova, Italy.*

⁴ *Consorzio CREATE, DIETI, Università di Napoli Federico II, Italy.*

Received 29 March 2021; Accepted (in revised version) 27 September 2021

Abstract. In this paper we present a coupled Finite Element Method – Boundary Element Method (FEM-BEM) approach for the solution of the free-boundary axis-symmetric plasma equilibrium problem. The proposed method, obtained from an improvement of the Hagenow-Lackner coupling method, allows to efficiently model the equilibrium problem in unbounded domains by discretizing only the plasma region; the external conductors can be modelled either as 2D or 3D models, according to the problem of interest. The paper explores different iterative methods for the solution of the nonlinear Grad-Shafranov equation, such as Picard, Newton-Raphson and Newton-Krylov, in order to provide a robust and reliable tool, able to handle large-scale problems (e.g. high resolution equilibria). This method has been implemented in the FRIDA code (Free-boundary Integro-Differential Axisymmetric – <https://github.com/matteobonotto/FRIDA>), together with a suitable Adaptive Integration Technique (AIT) for the computation of the source term. FRIDA has been successfully tested and validated against experimental data from RFX-mod device, and numerical equilibria of an ITER-like device.

PACS: 52.65.Kj, 52.55.Fa, 52.55.Hc, 41.20.Gz

Key words: FEM-BEM, MHD, plasma equilibrium, Grad-Shafranov equation.

1 Introduction

In magnetic confinement fusion (MCF) research, the successful design of Tokamak devices, the set up of plasma operations, the prediction of performance scenarios and the

*Corresponding author. *Email addresses:* matteo.bonotto@igi.cnr.it (M. Bonotto), domenico.abate@igi.cnr.it (D. Abate), paolo.bettini@unipd.it (P. Bettini), fabio.villone@unina.it (F. Villone)

design of feedback control systems are all activities which rely, to an extent, on equilibrium codes [1, 2]. High accuracy MHD stability analysis also relies on given equilibrium reconstructions, which are usually performed in fixed-boundary conditions. One of the basic problems of equilibrium reconstructions is the more general quasi-static *free-boundary* plasma equilibrium problem, consisting in the numerical solution of the Grad-Shafranov equation [3, 4] with a plasma separatrix, the *plasma-vacuum interface*, which is not known a priori.

The solution of the free-boundary axis-symmetric equilibrium problem requires to deal with a two-dimensional, elliptic partial differential equation (PDE), defining the axis-symmetric plasma equilibrium, given the prescribed set of external currents, the plasma current density profile and the total plasma current value. The computational challenges in solving such magneto-quasistatic problem are related to the fact that (i) the computational domain is unbounded, (ii) the plasma current density profiles is a non-linear function of the poloidal magnetic flux and (iii) the plasma separatrix is unknown.

Despite the well-assessed theory concerning the axisymmetric quasi-static plasma equilibrium problem [5, 6], its numerical solution is still a hot topic. Many equilibrium codes are based on Finite-Difference (FD) or Finite-Element-Method (FEM) for the spatial discretization of the PDE, a fixed-point (Picard) iteration scheme to solve the nonlinearities, and a coupling method, based on the analytical Green's functions [7], to reduce the unbounded domain to a bounded region. Among these codes, some examples are FBT [8], MAXFEA [9], and TES [10]. Other computational tools are based on a Newton-Raphson like iterative scheme, and a more sophisticated analytic uncoupling on a semi-circular domain, which was introduced by Albanese, Blum and de Barbieri [11]. This approach is implemented, for examples, in the codes PROTEUS [12], CREATE-L [13], CREATE-NL+ [14], CEDRES++ [15] and FEEQS.M [5]. All these codes can be divided in two groups, depending on how the plasma equilibrium problem is treated. On one hand, we find all the tools based on robust and reliable Newton-Raphson like schemes used to solve the non-linear equilibrium problem in a rigorous way (i.e. PROTEUS, CREATE-L, CREATE-NL+, CEDRES++, and FEEQS.M). On the other hand, a group of codes (i.e. FBT, MAXFEA, and TES) rely on the idea that the free-boundary equilibrium problem is characterized by an *intrinsic axisymmetric instability which is encountered in all equilibrium calculations with a free-boundary condition*. This misconception is explicitly expressed in [10], and more indirectly in [16], and it is pretty common in the MCF community, with the consequence of believing that it is impossible to solve the free-boundary plasma equilibrium problem without handling this *intrinsic axisymmetric instability* by means of naive numerical approaches (e.g. with the insertion of a feedback loop). However, such misconceptions come from a misinterpretation of the convergence limitations of the Picard scheme, much less robust and reliable than Newton-Raphson-like scheme, as will be clarified in sections 5 and 6.

In the sake of a rigorous mathematical analysis of the problem [5, 6], in this paper we present a coupled Finite Element Method – Boundary Element Method (FEM-BEM) approach obtained improving the well-known Hagenow-Lackner (HL) coupling method

[7]. This method has been implemented in the FRIDA code (Free-boundary Integro-Differential Axisymmetric – <https://github.com/matteobonotto/FRIDA>), in which the boundary convolution integral over the plasma current is kept explicit instead of introducing an auxiliary variable to reduce the problem to a one dimensional convolution integral. This strategy allows to overcome the convergence limitations of the HL scheme, providing a FEM-BEM coupling scheme which has optimal convergence. Moreover, the computation of the Green matrix, which depends only on the geometry of the domain triangulation, is performed beforehand during a pre-processing phase and it does not need to be repeated if another equilibrium with the same mesh is analyzed. Finally, an Adaptive Integration Technique (AIT) for the computation of the plasma source term has been implemented, in order to treat the mismatch problem between plasma separatrix domain, which is not known a priori, and the mesh elements. The FRIDA code exploits Newton-like iterative schemes (Newton-Raphson, Newton-Krylov) in order to solve the free-boundary problem in a rigorous way. The Picard scheme is also implemented to compare its performances with Newton-like methods.

It is worth noting that the presented FEM-BEM coupling method has been derived also with the purpose to give a modular tool, suitable to deal with different representations of the external field sources: by solving the equilibrium problem only in the plasma region, the boundary conditions contain the contribution of all external field sources, both 2D axi-symmetric (i.e. poloidal field coils) and pure 3D (e.g. conductors with arbitrary geometry). Moreover, it is suitable to be generalized for time-evolution problems, considering also the eddy current contribution as part of the equilibrium problem through the boundary conditions. This approach is not novel, but, in the authors' knowledge, the only existing implementation is that of CarMa0NL code [17].

The paper is organized as follows. Sections 2 to 4 describe the mathematical and numerical formulation of the free-boundary axi-symmetric equilibrium problem. Section 5 describes the implementation of the iterative schemes used to deal with the non-linear problem. Section 6 presents the validation of the FRIDA code against experimental data from RFX-mod device and numerical equilibria of an ITER-like device. Finally, Section 7 draws the conclusions.

2 Axi-symmetric equilibrium equation

The magneto-static problem in the three-dimensional space can be cast as:

$$\nabla \times \left(\frac{1}{\mu_0} \nabla \times \vec{A} \right) = \vec{j}, \quad (2.1)$$

where \vec{A} is the magnetic vector potential, \vec{j} is the current density and μ_0 is the magnetic permeability of vacuum. We consider now a cylindrical coordinate system (r, ϕ, z) : under the assumption of axial symmetry, Eq. (2.1) becomes a scalar equation in the poloidal

plane (r, z) :

$$L(\psi) = -\mu_0 j_\phi(\vec{r}), \quad (2.2)$$

$$\psi(\vec{r})|_{r=0} = 0, \quad (2.3)$$

$$\lim_{\|\vec{r}\| \rightarrow \infty} \psi(\vec{r}) = 0, \quad (2.4)$$

where $L(\psi) = \nabla \cdot (\frac{1}{r} \nabla \psi)$, $\vec{r} = (r, z)$ and $\psi(\vec{r})$ is the poloidal flux *per radian*, defined as:

$$\psi(\vec{r}) = \frac{1}{2\pi} \int_{S_\psi(\vec{r})} (\nabla \times \vec{A}) \cdot \mathbf{n} \, dS, \quad (2.5)$$

where $S_\psi(\vec{r})$ is a circular surface centered on the z axis and obtained by rotating a certain point \vec{r} around the z axis.

The computational domain consists of the following regions:

- vacuum internal region Ω_v , the vacuum region *available* to the plasma (i.e. the space inside the vacuum chamber);
- vacuum external region Ω_v^* , a narrow region which encircles Ω_v and *not available* to the plasma (i.e. introduced for computational reasons);
- coils region $\Omega_c = \bigcup_{i=1}^{N_c} \Omega_{ci}$, the region of the N_c active coils.

The boundary of the computational domain is labelled as $\partial\Omega$. In addition to this, we label as Ω_p the plasma cross-section, i.e. the region inside the plasma separatrix Γ_p (i.e. the interface between the plasma and the vacuum region). It is worth noting that, for a given a MCF device, the regions Ω_v , Ω_v^* and Ω_c depend only on the machine geometry, and thus can be defined just once beforehand and kept fixed. Conversely, Ω_p is an unknown of the problem because it depends on the plasma equilibrium (i.e. *free-boundary problem*). Fig. 1 shows, for the RFX-mod device [18], a typical example of domain subdivision.

Therefore the source term j_ϕ assumes different definitions depending on the region:

$$j_\phi(\vec{r}) = \begin{cases} j_p(\vec{r}, \bar{\psi}) = \lambda \left[r \frac{dp(\bar{\psi})}{d\bar{\psi}} + \frac{f(\bar{\psi})}{\mu_0 r} \frac{df(\bar{\psi})}{d\bar{\psi}} \right], & \text{in } \Omega_p, & (2.6a) \\ j_{ci}, & \text{in } \Omega_{ci}, i = 1, \dots, N_c, & (2.6b) \\ 0, & \text{elsewhere,} & (2.6c) \end{cases}$$

where p is the kinetic pressure and $f(\psi)$ is the poloidal current function, defined as:

$$f(\psi) = \frac{I_{pol}(\psi)}{2\pi}. \quad (2.7)$$

The term $I_{pol}(\psi)$ is the total *poloidal* current, i.e. related to the component of the current density which flows in the poloidal direction, and it is defined accordingly with (6.12) of [2].

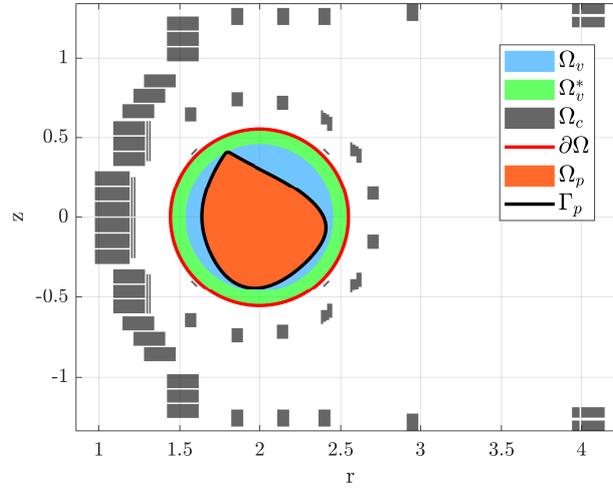


Figure 1: Definition of regions Ω_v , Ω_v^* , Ω_c , and Ω_p .

The profiles of $p(\bar{\psi})$ and $f(\bar{\psi})$ are usually given in terms of the normalized poloidal flux $\bar{\psi} = (\psi - \psi_a) / (\psi_b - \psi_a)$, where ψ_a and ψ_b are respectively the value of the flux at the magnetic axis and at the plasma separatrix Γ_p . These functions cannot be derived from the simple toroidal force balance, and must be determined either from experimental data, or from transport calculation, or using some suitable parametrization functions [6, 19].

In addition to this, the scaling parameter λ is needed to impose the total plasma current as:

$$\int_{\Omega_p} j_p(\vec{r}, \bar{\psi}) d\Omega = I_p. \quad (2.8)$$

Eq. (2.2) inside the plasma region, where the current density is defined in (2.6a), is called *Grad-Shafranov equation* (GSE), which is the basic equation describing the plasma equilibrium for axisymmetric toroidal configurations [2].

From a rigorous point of view, the computational domain is $[r, z] \in [0 \times \infty, -\infty \times \infty]$, but a limited-bounded domain is required to solve the problem numerically. For this reason a limited region of vacuum, which surrounds the plasma region without intersecting or overlapping the active conductors, is considered. An example of this configuration is shown in Fig. 1, where the limited computational domain $\Omega_d = \Omega_v \cup \Omega_v^*$ is highlighted. From a mathematical point of view, this means that the regularity conditions at infinity expressed by (2.4) can be replaced with:

$$\psi(\vec{r})|_{\partial\Omega} = \hat{\psi}, \quad (2.9)$$

where $\hat{\psi}$ is the set of Dirichlet boundary conditions (BCs) on the boundary $\partial\Omega$.

The unknowns are the value of ψ in Ω_d and $\hat{\psi}$ on $\partial\Omega$, together with additional degrees of freedom (DoFs) required in order to solve the free-boundary equilibrium problem,

such as: the current scaling parameter λ required to impose the total plasma current I_p , the profiles of the pressure $p(\bar{\psi})$ and the poloidal current function $f(\bar{\psi})$. The geometry and the current of the external coils (j_{Ci} , $i = 1, \dots, N_c$) are also required. Other quantities, such as the position of the magnetic axis, are unknowns of the equilibrium problem.

Eq. (2.2) can be written in a weak form in the limited-bounded domain Ω_d as:

$$\int_{\Omega_d} \frac{1}{r} \nabla w(\vec{r}) \cdot \nabla \psi(\vec{r}) d\Omega - \int_{\partial\Omega} \frac{1}{r} w(\vec{r}) \frac{\partial \psi(\vec{r})}{\partial \mathbf{n}} dS = \int_{\Omega_d} \mu_0 w(\vec{r}) j_\phi(\vec{r}) d\Omega, \quad (2.10)$$

where $w(\vec{r})$ is a scalar test function. This problem is treated numerically following the Finite Element Method (FEM) approach.

3 FEM-BEM coupling method

The numerical solution of the problem given by (2.2), (2.3) and (2.9) in unbounded domain requires, together with (2.10) inside Ω_d , a suitable coupling equation for the BCs on $\partial\Omega$. The total flux at the boundary can be split into:

$$\hat{\psi}(\vec{r}, \psi) = \hat{\psi}_p(\vec{r}, \psi) + \hat{\psi}_c(\vec{r}), \quad (3.1)$$

where $\hat{\psi}_p$, $\hat{\psi}_c$ are the contributions of the plasma and the external coils, respectively. The total flux at the boundary depends both on the known total value I_p and on the current density distribution, which is a non-linear function of ψ : for this reason $\hat{\psi}(\vec{r}, \psi)$ is also non-linear and depends on ψ . Conversely, the contribution of the external coils is fixed and depends only on the geometry and the current of the external coils.

It is worth noting that $\hat{\psi}_c$ contains the contribution of all external sources. This approach is the same described in [17] for the CarMa0NL code, which solves the axisymmetric non-linear plasma equilibrium problem, self-consistently coupled to eddy currents equations describing the three-dimensional (3D) structures surrounding the plasma. Therefore, our approach is suitable to be generalized for time-evolution problems, considering also the eddy current contribution as part of the equilibrium problem through the BCs $\hat{\psi}_c$.

The typical way to reduce the computation on the unbounded domain to a limited-bounded domain relies on the analytical Green's function. The most famous coupling method in MCF community is the von Hagenow-Lackner method (HL) [7], and it is based on the Green's function. Another method, which is less known and used, but noteworthy, is the aforementioned analytic uncoupling on a semi-circular domain by Albanese, Blum and de Barbieri (ABB) [11]. Additional coupling methods, well known in applied mathematics but not in the MCF community, are Johnson-Nédélec coupling (JN) [20] and Bielak-MacCamy coupling (BMC) [21]. A comprehensive review of these coupling methods can be found in [5].

Eq. (3.1) is treated recalling the fundamental solution of the operator $L(\psi)$ defined in Eq. (2.2), also known as Green function of $L(\psi)$, giving the integral formulation of the

Boundary Element problem:

$$\hat{\psi}_p(\vec{r}, \psi) = \int_{\Omega_p} G(\vec{r}_b, \vec{r}) j_p(\vec{r}, \psi) d\Omega, \quad (3.2)$$

$$\hat{\psi}_c(\vec{r}) = \sum_{i=1}^{N_c} \int_{\Omega_{ci}} G(\vec{r}_b, \vec{r}) j_{ci} d\Omega, \quad (3.3)$$

where $\vec{r}_b \in \partial\Omega$, Ω_{ci} is the i -th active coil, N_c is the total number of active coils and $G(\vec{r}, \vec{r}')$ is the Green's function defined as [7]:

$$G(\vec{r}, \vec{r}') = \frac{\mu_0 \sqrt{rr'}}{2\pi k(\vec{r}, \vec{r}')} \left[(2 - k^2(\vec{r}, \vec{r}')) K(k(\vec{r}, \vec{r}')) - 2E(k(\vec{r}, \vec{r}')) \right], \quad (3.4)$$

where

$$k^2(\vec{r}, \vec{r}') = \frac{4rr'}{(r+r')^2 + (z-z')^2}, \quad (3.5)$$

and $K(k)$, $E(k)$ are the complete elliptic integrals of the first and second kind, respectively.

Recalling the domain subdivisions presented in chapter 2, the additional region Ω_v^* , forbidden to the plasma, is intended to avoid singularities on integral (3.2). By doing this, the mesh nodes inside Ω_p , which are the sources of (3.2), and the boundary nodes never coincide.

The FEM-BEM coupling strategy is also the starting point of the Hagenow-Lackner coupling (HL). However, the HL coupling, in order to avoid the direct computation of the integral in (3.2) over the possible large domain Ω_p , introduces a new auxiliary unknown, which satisfies the homogeneous Dirichlet boundary value problem on $\partial\Omega$, thus allowing to replace the integral over plasma domain and coils by an integral over the boundary $\partial\Omega$ using the Neumann data of the auxiliary function. The difference between HL method and our method, and pros and cons, will be discussed in paragraph 6.1.

4 Galerkin formulation

Following the standard Galerkin approach, we use a Quadratic Lagrangian Element (QLE) based Finite Element Method. The computational domain is discretized by a suitable triangularization algorithm [22], and the following basis function is used [23]:

$$w(\vec{r}) = a_1 r^2 + a_2 z^2 + a_3 r z + a_4 r + a_5 z + a_6, \quad (4.1)$$

which is a complete second order polynomial of (r, z) of coefficients $\mathbf{a} = [a_1 \dots a_6]^T$. Thus, since triangular elements and nodal basis functions are used, the boundary elements are defined as the nodes at the boundary $\partial\Omega$, whose support is given by the triangles with one or more edges $\partial\Omega$.

The terms ψ and j_p can be expressed in terms of (4.1) as:

$$\psi(\vec{r}) = \sum_{i=1}^{N_n} \left(\sum_{j=1}^{N_e^i} w_j^i(\vec{r}) \right) \psi_i, \quad (4.2)$$

$$j_p(\vec{r}, \bar{\psi}) = \sum_{i=1}^{N_n} \left(\sum_{j=1}^{N_e^i} w_j^i(\vec{r}) \right) j_p(\bar{\psi}_i), \quad (4.3)$$

where N_n is the number of mesh nodes. For a given i -th node there are $j = 1, \dots, N_e^i$ triangles sharing such node and forming its local support Ω_e^i ; $w_j^i(\vec{r})$ is the basis function on the j -th triangle of the local support and related to the i -th node.

The current density can be obtained from the profiles of the pressure and the poloidal current function, or through a suitable parametrization [6, 19]. In the following, we will pose:

$$j_\phi = \lambda g(\psi, \psi_a, \psi_b), \quad (4.4)$$

where the current density function $g(\psi, \psi_a, \psi_b)$ can be defined as:

$$g(\psi, \psi_a, \psi_b) = r \frac{dp(\bar{\psi})}{d\bar{\psi}} + \frac{f(\bar{\psi})}{\mu_0 r} \frac{df(\bar{\psi})}{d\bar{\psi}}, \quad (4.5)$$

or

$$g(\psi, \psi_a, \psi_b) = \left[\frac{r\beta_0}{R_0} + \frac{(1-\beta_0)R_0}{r} \right] (1 - \bar{\psi}^{\alpha_M})^{\alpha_N}, \quad (4.6)$$

depending on which information is available.

In (4.6) R_0 is the machine major radius and the parameters $[\alpha_M, \alpha_N, \beta_0]$ have to be chosen to match the values of plasma internal inductance l_i and poloidal β_p . The best values of $[\alpha_M, \alpha_N, \beta_0]$ to characterize the equilibrium are determined by applying a least-squares minimization technique to the difference between the measured and calculated values of the poloidal field [19]. This means that, in (4.6), both $df(\bar{\psi})/d\bar{\psi}$ and $dp(\bar{\psi})/d\bar{\psi}$ have the same dependence on $\bar{\psi}$, because the use of only magnetic measures makes practically unfeasible the reconstruction of $df(\bar{\psi})/d\bar{\psi}$ and $dp(\bar{\psi})/d\bar{\psi}$ profiles separately. On the other hand, if also non-magnetic diagnostics are available (i.e. Thomson Scattering, Polarimetry etc), these profiles can be separated; this is usually necessary for particular operational conditions (e.g. profiles with a strong pedestal). A further generalization of this approach will involve also such a contribution.

The final system of equations is obtained: (i) by using the Galerkin formulation on (2.10) for the FEM problem (first integral of (2.10) written both for ψ and $\hat{\psi}$), and on (3.1)-(3.2)-(3.3) for the BEM problem, and writing the source as (4.4); (ii) by considering that the fluxes ψ_a, ψ_b are also unknown, because they depend on the solution $\psi(\vec{r})$; and (iii) by

imposing the total plasma current. It follows that:

$$\begin{aligned} \int_{\Omega_d} \frac{1}{r} \nabla w(\vec{r}) \cdot \nabla \psi(\vec{r}) d\Omega + \int_{\Omega_d} \frac{1}{r} \nabla w(\vec{r}) \cdot \nabla \hat{\psi}(\vec{r}) d\Omega \\ = \lambda \int_{\Omega_d} \mu_0 w(\vec{r}) g(\bar{\psi}, \psi_a, \psi_b) d\Omega, \end{aligned} \quad (4.7)$$

$$\hat{\psi}_c(\vec{r}) + \lambda \int_{\Omega_p} G(\vec{r}, \vec{r}_p) w(\vec{r}) g(\psi, \psi_a, \psi_b) d\Omega = \hat{\psi}(\vec{r}), \quad (4.8)$$

$$\max_{\Omega_p} \{\psi\} = \psi_a, \quad (4.9)$$

$$\max_{\partial\Omega_v^- \cup \Omega_{v0}} \{\psi\} = \psi_b, \quad (4.10)$$

$$\lambda \int_{\Omega_p} w(\vec{r}) g(\bar{\psi}, \psi_a, \psi_b) d\Omega = I_p, \quad (4.11)$$

where $\partial\Omega_v^-$ is the portion $\partial\Omega_v$ where $\vec{n} \cdot \nabla \psi < 0$, with \vec{n} unit vector outgoing from $\partial\Omega_v$, and Ω_{v0} is the region inside Ω_v where the poloidal magnetic field vanishes. Eqs. (4.9)-(4.10) are suitable for most of the typical limiter/diverted configurations: additional constrains might be necessary for more particular configurations (e.g. plasma during disruption events).

It is worth noting that the second term of (2.10) is identically zero for the interior elements (i.e. elements without nodes on the computational boundary), and it contributes only if BCs involving normal derivative of the unknown are considered (i.e. Neumann/Robin BCs). Since, in this context, only Dirichlet BCs are considered, and specifically through (4.8), the second term of (2.10) gives no contribution in (4.7).

Eqs. (4.7)-(4.11) give a system of $N_d + N_b + 3$ equations in $N_d + N_b + 3$ unknowns, where N_d is the number of the mesh nodes inside the domain Ω_d and N_b is the number of nodes on the boundary $\partial\Omega$, respectively.

The discrete version of the previous system can be written in matrix form, by defining the vector of unknowns $\mathbf{x} = [\psi, \hat{\psi}, \psi_a, \psi_b, \lambda]^T$:

$$\begin{bmatrix} \mathbf{K} & \mathbf{K}_{bc} & 0 & 0 & -\mu_0 \mathbf{i}(\mathbf{x}) \\ 0 & \mathbf{E} & 0 & 0 & -\mathbf{G}_{bp} \mathbf{i}(\mathbf{x}) \\ \chi_a(\mathbf{x}) & 0 & -1 & 0 & 0 \\ \chi_b(\mathbf{x}) & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & I_t(\mathbf{x}) \end{bmatrix} \begin{bmatrix} \psi \\ \hat{\psi} \\ \psi_a \\ \psi_b \\ \lambda \end{bmatrix} = \begin{bmatrix} 0 \\ \hat{\psi}_c \\ 0 \\ 0 \\ I_p \end{bmatrix}, \quad (4.12)$$

where $\psi, \hat{\psi}$ are the discrete counterparts of $\psi, \hat{\psi}$ and \mathbf{E} is the identity matrix. The matrix blocks $\mathbf{K}(i, j)$ and $\mathbf{K}_{bc}(i, j)$ are defined as:

$$\mathbf{K}(i, j) = \int_{\Omega_{ij}} \frac{1}{r} \nabla w_i \cdot \nabla w_j d\Omega, \quad i, j \in N_d, \quad (4.13)$$

$$\mathbf{K}_{bc}(i, j) = \int_{\Omega_{ij}} \frac{1}{r} \nabla w_i \cdot \nabla w_j d\Omega, \quad i \in N_d, j \in N_b, \quad (4.14)$$

where $\Omega_{ij} = \Omega_e^i \cap \Omega_e^j$. All integrals (4.13) and (4.14) and in following relations are evaluated numerically using high degree efficient symmetrical Gaussian quadrature rules for triangles [24].

The term $\mathbf{i}(\mathbf{x})$ is the vector of discrete currents, whose elements $I_i(\mathbf{x})$, $i = 1, \dots, N_n$ are obtained by integrating the source function $g(\bar{\psi}_i, \psi_a, \psi_b)$ on the mesh elements:

$$I_i(\mathbf{x}) = I_i(\boldsymbol{\psi}, \psi_a, \psi_b) = \int_{\Omega_e^i} \left(\sum_{j=1}^{N_e^i} w_j^i(\bar{\mathbf{r}}) \right) g(\bar{\psi}_i, \psi_a, \psi_b) d\Omega, \quad (4.15)$$

while I_t is the total current:

$$I_t(\mathbf{x}) = \sum_{i=1}^{N_n} I_i(\mathbf{x}). \quad (4.16)$$

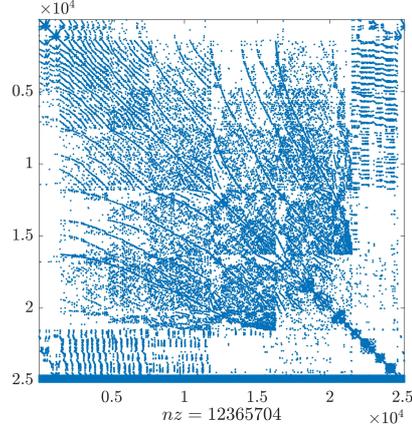
The discrete counterpart of (4.8) has been written assuming that, for the i -th node, the current is not distributed on the support Ω_e^i of each node but each node carries all the current. Thus the convolution integral in (4.8) becomes $\mathbf{G}_{bp} \mathbf{i}(\mathbf{x})$, where $\mathbf{G}_{bp} = \mathbf{G}(\bar{\mathbf{r}}_b, \bar{\mathbf{r}}_p)$ defined according to (3.4).

Eqs. (4.9)-(4.10) can be written, in a discrete manner, by introducing vectors $\boldsymbol{\chi}_a(\mathbf{x})$ and $\boldsymbol{\chi}_b(\mathbf{x})$, which are vectors of weights to identify the positions of the magnetic axis and the XP/limiter points respectively in terms of $\boldsymbol{\psi}$. To give a clarification, we consider the magnetic axis. Let Ω_t^{ax} be the 6-nodes triangle in which the magnetic axis is located, and $w_j^{ax}(\bar{\mathbf{r}})$, $j = 1, \dots, 6$ the set of basis functions of such a triangle. Since ψ_a is a combination of the values of $\boldsymbol{\psi}$ at the nodes of the triangle Ω_t^{ax} , $\boldsymbol{\chi}_a(\mathbf{x})$ will have all zero entries, except for the elements related to the nodes of triangle Ω_t^{ax} . Since $\boldsymbol{\psi}$ is parameterized in terms of quadratic basis functions, the weight vector $\boldsymbol{\chi}_a(\mathbf{x})$ is different from zero only on the triangle containing the magnetic axis, with weights equal to the values of each node's basis function on $\bar{\mathbf{r}}_a$. The same holds for ψ_b and $\boldsymbol{\chi}_a(\mathbf{x})$. This is written in a formal way as:

$$\boldsymbol{\chi}_a(\mathbf{x}) = \begin{cases} \sum_{j=1}^6 w_j^{ax}(\bar{\mathbf{r}}_a), & \text{for } i = \text{index of magnetic axis,} \\ 0, & \text{elsewhere,} \end{cases} \quad (4.17)$$

$$\boldsymbol{\chi}_b(\mathbf{x}) = \begin{cases} \sum_{j=1}^6 w_j^{XP}(\bar{\mathbf{r}}_{XP}), & \text{for } i = \text{index of XP/limiter,} \\ 0, & \text{elsewhere,} \end{cases} \quad (4.18)$$

where $\bar{\mathbf{r}}_a$ and $\bar{\mathbf{r}}_{XP}$ are the positions of the magnetic axis and the XP/limiter points respectively. In this work, the proposed method for the identification of $\bar{\mathbf{r}}_a$ and $\bar{\mathbf{r}}_{XP}$ is to search for the points where $\|\nabla \psi\| = 0$. This can be done element by element, as the components of the gradient, obtained by differentiating (4.1), are linear functions of (r, z) . However, since the Lagrangian-based FEM gives a solution which is not C^1 , the fact that the gradient is not, in principle, continuous has to be taken into account. The problem can be addressed by suitably refining the triangulation in the regions where such points are expected.

Figure 2: Sparsity pattern of the matrix $A(x)$ of (4.21).

By substituting the term $\lambda \mathbf{i}(x)$ from the FEM block

$$\lambda \mathbf{i}(x) = \frac{1}{\mu_0} (\mathbf{K}\psi + \mathbf{K}_{bc}\hat{\psi}), \quad (4.19)$$

into the BEM block, we get:

$$\underbrace{-\frac{1}{\mu_0} \mathbf{G}_{bp} \mathbf{K} \psi}_{\hat{\mathbf{K}}} + \underbrace{\left(\mathbf{E} - \frac{1}{\mu_0} \mathbf{G}_{bp} \mathbf{K}_{bc} \right)}_{\hat{\mathbf{K}}_{bc}} \hat{\psi} - \hat{\psi}_c = 0. \quad (4.20)$$

Thus, the system (4.12) becomes:

$$\underbrace{\begin{bmatrix} \mathbf{K} & \mathbf{K}_{bc} & 0 & 0 & -\mu_0 \mathbf{i}(x) \\ \hat{\mathbf{K}} & \hat{\mathbf{K}}_{bc} & 0 & 0 & 0 \\ \chi_a(x) & 0 & -1 & 0 & 0 \\ \chi_b(x) & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & I_t(x) \end{bmatrix}}_{A(x)} \underbrace{\begin{bmatrix} \psi \\ \hat{\psi} \\ \psi_a \\ \psi_b \\ \lambda \end{bmatrix}}_x = \underbrace{\begin{bmatrix} 0 \\ \hat{\psi}_c \\ 0 \\ 0 \\ I_p \end{bmatrix}}_b. \quad (4.21)$$

The reason of this manipulation is clarified in Section 5. The sparsity pattern of the matrix (4.21) is shown in Fig. 2, where the dense BEM block related to $\hat{\mathbf{K}}$ can be seen clearly in the lowest part of the matrix.

5 Solution of the non-linear equilibrium problem

The problem is non linear because the source term $\mathbf{i}(x)$ in Ω_p is a non-linear function of the solution x , and because the plasma domain, delimited by the plasma separatrix Γ_p , is

not known a priori. For this reason, (4.21) is usually written in the form:

$$F(\mathbf{x}) = A(\mathbf{x})\mathbf{x} - \mathbf{b} = 0, \quad (5.1)$$

and a suitable iterative scheme is used to find the zero of the function $F(\mathbf{x})$.

It is worth noting that a suitable initial guess \mathbf{x}_0 is needed: in this context, *suitable* means that $\mathbf{x}_0 = [\psi_0, \hat{\psi}_0, \psi_{a0}, \psi_{b0}, \lambda_0]$ must be *sufficiently close* to the solution \mathbf{x} . The initial guess can be obtained using several methods to model the plasma, such as a single filament placed in a suitable position (i.e. the current centroid if available, but the machine center can be used, as well) and carrying the total plasma current, or an equivalent filamentary model with a prescribed number of filaments matching the lower order moments of the plasma current density [25].

However, the solution of this problem is demanding for several reason. Firstly, it is an open boundary problem, in which the BCs $\hat{\psi}$ on the computational boundary $\partial\Omega$ are part of the solution. This leads to a large cost from the computational point of view, due to the dense BEM block in (4.21). Moreover, a robust and reliable iterative scheme is needed: as was already mentioned in the introduction, the choice of a robust iterative scheme such as Newton-like scheme is fundamental to avoid convergence issues, which are conversely common drawbacks of Picard-type schemes. To investigate this aspects, three numerical schemes will be analyzed in details in the next sections: Picard, Newton-Raphson, Newton-Krylov.

5.1 Picard iteration

The Picard iteration has been used in several implementations because it gives an easy and straightforward way to compute the solution \mathbf{x}_{k+1} by searching for the fixed-point of equation $A(\mathbf{x})\mathbf{x} = \mathbf{b}$:

$$\mathbf{x}_{k+1} = A^{-1}(\mathbf{x}_k)\mathbf{b}. \quad (5.2)$$

Unfortunately, it is well known that the Picard iterative scheme converges if and only if $\rho[J_F(\mathbf{x}_k)] < 1$ [26, 27], where $\rho[J_F(\mathbf{x}_k)]$ is the spectral radius of the Jacobian of $F(\mathbf{x}_k)$ (i.e. the largest eigenvalue in absolute value). This means that the function $F(\mathbf{x})$ is contracting over the iterations.

As it will be pointed out in Section 6.2, the assumption of $\rho[J_F(\mathbf{x}_k)] < 1$ is usually not satisfied: the solution can exhibit a progressive drift during the iterations, leading the plasma to move away from the expected equilibrium position and to collapse on boundary of the plasma region. It is worth noting that this behaviour does not depend on the good quality of the initial guess.

One way to avoid numerical instability during the Picard iterations is to fix some geometrical quantities, resulting in artificial numerical ways to help convergence in such cases where this numerical scheme is not supposed to converge. The most known artificial ways is to fix the magnetic axis through an additional set of coils' current, added to the equilibrium currents, in order to adjust the coils' current over the iterations to keep

the magnetic axis at the desired position [10] (i.e. resulting in a *numerical fictitious* feedback controller which has the only aim to help the convergence of the nonlinear solver).

5.2 Newton-Raphson method

One of the aims of this work is to show that the free-boundary equilibrium problem can be rigorously solved without relying on any artificial numerical way. For this purpose, we exploit the Newton-Raphson (NR) scheme, which is more stable and reliable if a suitable initial guess is given, and has, under certain assumptions, a quadratic convergence. The strengths of the NR scheme are reviewed in details in [5], and many examples of implementations of this scheme can be found in literature [6, 13, 14, 17].

For this reason, (5.1) is solved with the *relaxed* (or *damped*) Newton-Raphson scheme [28]:

$$J_F(\mathbf{x}_k)\mathbf{h}_{k+1} = -F(\mathbf{x}_k), \tag{5.3}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \alpha_k \mathbf{h}_{k+1}, \tag{5.4}$$

where J_F is the jacobian matrix and the relaxation parameter α_k is suitably chosen at every iteration to avoid an overshoot of the solution, especially during the very first iterations when the solution between two following steps typically changes dramatically.

For this specific problem, $J_F(\mathbf{x}_k)$, defined as:

$$J_F(\mathbf{x}_k) = \frac{\partial F(\mathbf{x}_k)}{\partial \mathbf{x}_k}, \tag{5.5}$$

can be written as the sum of two terms:

$$J_F(\mathbf{x}_k) = J'_F(\mathbf{x}_0) + J''_F(\mathbf{x}_k), \tag{5.6}$$

where

$$J'_F(\mathbf{x}_0) = \underbrace{\begin{bmatrix} \mathbf{K} & \mathbf{K}_{bc} & 0 & 0 & 0 \\ \hat{\mathbf{K}} & \hat{\mathbf{K}}_{bc} & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}}_{\text{fixed}}, \tag{5.7}$$

$$J''_F(\mathbf{x}_k) = \underbrace{\begin{bmatrix} -\mu_0 \frac{di}{d\psi} & 0 & -\mu_0 \frac{di}{d\psi_a} & -\mu_0 \frac{di}{d\psi_b} & -\mu_0 \frac{di}{d\lambda} \\ 0 & 0 & 0 & 0 & 0 \\ \chi_a(\mathbf{x}) & 0 & 0 & 0 & 0 \\ \chi_b(\mathbf{x}) & 0 & 0 & 0 & 0 \\ \frac{dI_t}{d\psi} & 0 & \frac{dI_t}{d\psi_a} & \frac{dI_t}{d\psi_b} & \frac{dI_t}{d\lambda} \end{bmatrix}}_{\text{to be updated at every iteration}}. \tag{5.8}$$

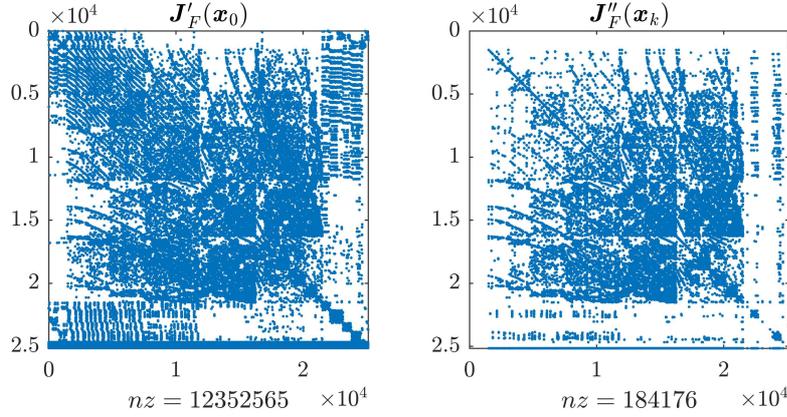


Figure 3: Sparsity pattern of the terms $J'_F(x_0)$ and $J''_F(x_k)$.

It is worth noting that $J'_F(x_0)$ depends only on the geometry of the problem, and it is always the same if the same mesh is considered. It follows that it can be computed only once during a pre-processing phase and then stored. On the other hand, $J''_F(x_k)$ has to be updated at every iteration, and it is computed using a hybrid semi-analytical approach which exploits the sparsity of the original matrix A . This approach is strongly parallelizable, ensuring a fast and efficient implementation, both in terms of CPU time and memory usage.

We underline that considering (4.21) rather than (4.12) is beneficial for several reasons. At first, the only non-trivial term in the derivatives of $F(x)$ corresponds to $di(\psi, \psi_a, \psi_b)/dx$, therefore the computation of the derivatives of the second block of equations, related to $\hat{\psi}$, is simplified because $i(\psi, \psi_a, \psi_b)$ is no longer present. Moreover, although both (4.12) and (4.21) allow the aforementioned partition of the Jacobian matrix into two terms $J'_F(x_0)$ and $J''_F(x_k)$, (4.21) leads to a more sparse $J''_F(x_k)$ with respect to (4.12), and this is because the equations related to the BEM block do not change during the iterations. For this reason the rows of $J''_F(x_k)$ related to the BEM block have all zero entries.

Fig. 3 shows the sparsity pattern of the terms $J'_F(x_0)$ and $J''_F(x_k)$. It can be seen that the BEM block \hat{K} appears only in $J'_F(x_0)$ (i.e. the lowest part).

5.3 Jacobian-free Newton-Krylov method

If, on one hand the computation of the Jacobian matrix $J_F(x_k)$ can be done in a very efficient way, on the other hand the solution of (5.3) can be computationally expensive, because of the dense BEM block of $J'_F(x_0)$, whose size depends on the number N_b of nodes on the boundary $\partial\Omega$. This is not a critical issue for small problems (i.e if the mesh is coarse), but it might be for large problems, because the computational cost of direct

methods scales as $O(n^3)$, with n the number of Degrees of Freedom (DoFs) of the system, making the problem unfeasible if a very fine mesh is considered.

To this purpose, the *Jacobian-Free* Newton-Krylov method (NK) have been implemented. The Newton-Krylov method is jacobian-free in the sense that (5.3) is solved without computing and inverting explicitly the Jacobian matrix. The computation of the jacobian matrix is sometimes "error prone and time consuming". If the jacobian is not sparse, also the solution of the system (5.3) can be time consuming. For this reason, this method has been exploited in plasma physics as well as in several other computational fields [29, 30]. For this specific problem the computation of the Jacobian matrix is not a particular issue, and has been efficiently implemented: it remains the issue related to the numerical solution of (5.3).

The NK method is a projection method which solves the problem $F(\mathbf{x}) = A(\mathbf{x})\mathbf{x} - \mathbf{b} = 0$ looking for the solution in the Krylov subspace[†]:

$$\mathcal{K}_j(J, \mathbf{r}_0) = \text{span}\{\mathbf{r}_0, J\mathbf{r}_0, J^2\mathbf{r}_0, \dots, J^{j-1}\mathbf{r}_0\}, \quad (5.9)$$

where here $J = J_F(\mathbf{x})$, $\mathbf{r}_0 = -F(\mathbf{x}) - J_F(\mathbf{x})\mathbf{h}_0$.

After j Krylov iterations, \mathbf{h}_j is:

$$\mathbf{h}_j = \mathbf{h}_0 + \sum_{i=0}^{j-1} \beta_i J^i \mathbf{r}_0, \quad (5.10)$$

where the scalars β_i minimizes the residual, which means that is determined as a linear combination of the orthonormal Arnoldi vectors produced by the Generalized Minimum Residual Method (GMRES) [31]. Eq. (5.10) shows that GMRES does not requires the jacobian matrix explicitly, but only in term of matrix vector product, which can be approximated as:

$$J_F(\mathbf{x})\mathbf{r} \approx \frac{[F(\mathbf{x} + \epsilon\mathbf{r}) - F(\mathbf{x})]}{\epsilon}, \quad (5.11)$$

where ϵ is a small perturbation. There are several approaches available in the literature to choose the perturbation ϵ , see for example [32].

The convergence of NK can be improved using a suitable preconditioner, in order to cluster near the unity the eigenvalues of the iteration matrix [26] and consequently reduce the number of GMRES (Krylov) iterations. A badly chosen preconditioner can even compromise at all the convergence. The method implemented in this work relies on the *right* preconditioning, which means that (5.3) becomes:

$$(J_F(\mathbf{x}_k)P^{-1})\mathbf{y} = -F(\mathbf{x}_k), \quad (5.12)$$

$$\mathbf{h}_{k+1} = P^{-1}\mathbf{y}, \quad (5.13)$$

[†]Since the NK solver is performed at *every* non-linear free-boundary iteration k , in the following description such index k is dropped, and only the index j of the NK iterations is considered.

where the matrix P is the preconditioner. Regarding the choice of the preconditioner, there are several strategies available in literature: one can choose either *Algebraic Preconditioners* (AP) [26], exploiting problem-independent matrix factorization techniques, or *Physics-Based Preconditioners* (PBP) [33], based on problem-dependent considerations. In this work we implemented both kind of preconditioners: specifically, for AP we extrapolate the optimal value of drop-tolerance of the DT-ILU factorization following an empirical rule, and, for PBP, we developed a far-field sparsification technique to use as preconditioner a sparsified version of the system matrix $A(x)$ in Eq. (4.21). These strategies are described in details in Appendix B, where an extensive comparison of the performances of preconditioned NK, against NR, is reported as well.

6 Validation of the FRIDA code

The validation of the FRIDA code involves several aspects: (i) the convergence of the FEM/BEM coupling, the (ii) investigation of the convergence capabilities of the numerical schemes adopted (Picard, Newton-Rahpson, Newton-Krylov), and (iii) the application of the code to realistic plasma equilibrium problems.

For these reasons, at first a simple magneto-static problem has been considered to study the convergence behaviour of the FEM-BEM coupling method, and to compare the outcome with the other coupling methods.

Secondly, two different devices are considered as test cases for the plasma equilibrium problem: RFX-mod [18], operating as low-current tokamak, and an ITER-like device. Fig. 4 shows the geometry of these devices and the regions Ω_p , Ω_c , and Ω_v .

All following simulations and numerical experiments have been performed on a Desktop machine with Intel Core i7 6 cores processor and 16GB 2666MHz. The FRIDA code has been developed in the MATLAB environment, with parallel C++ subroutines for the most demanding computations, such as the stiffness matrix in (4.13)-(4.14), the Green matrix in (4.8), and the terms of the Jacobian matrix to be updated ($J_F''(x_k)$ in 5.6, which is computed at every iteration following the procedure described in paragraph 5.2).

6.1 Convergence of FEM-BEM coupling method

In order to analyze the convergence of the FEM-BEM coupling method described in Section 3, we consider the simple axi-symmetric magneto-static problem given by a source Ω_S carrying a current density j_ϕ :

$$L(\psi) = \begin{cases} j_\phi(\vec{r}), & \text{in } \Omega_S, \\ 0, & \text{elsewhere.} \end{cases} \quad (6.1a)$$

$$(6.1b)$$

Two cases will be analyzed:

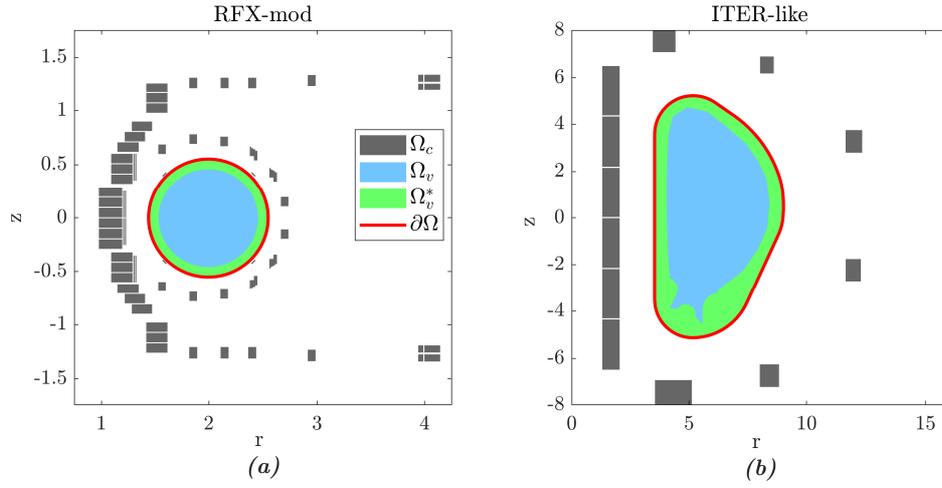


Figure 4: Definition of Ω_v , Ω_v^* , Ω_c and $\partial\Omega$ for RFX-mod (a) and ITER-like (b) geometries.

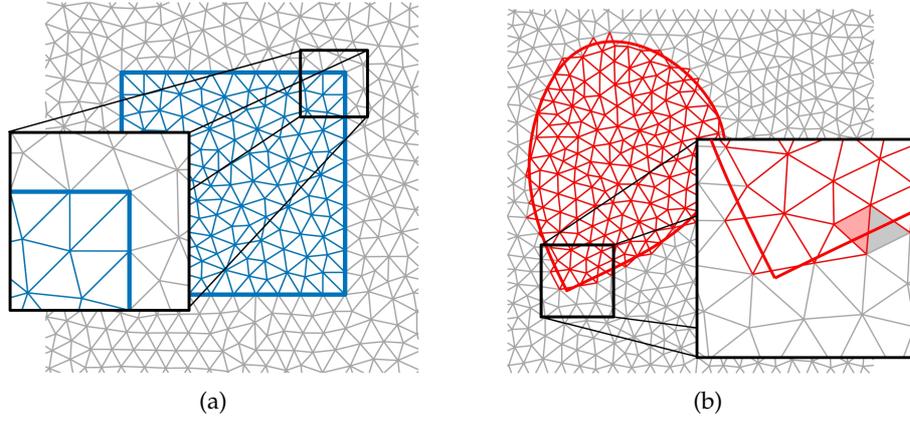


Figure 5: Two examples of triangulations: (a) matching and (b) non-matching the boundary of the source term. Filled triangles in (b): red = center of mass inside Γ_p , grey = outside.

1. $j_\phi(\vec{r}) = J_C$: a constant current density as in (2.6b), square Ω_S , describing a single coil as source (which will be called *vacuum case*);
2. $j_\phi(\vec{r}) = j_p(\vec{r}, \bar{\psi})$: a realistic plasma current density as in (2.6a), $\Omega_S = \Omega_p$ is a realistic separatrix of a diverted plasma configuration (which will be called *plasma case*).

These two cases have different purposes. The *vacuum case* is performed in order to prove the maximum achievable rate of convergence of the FEM-BEM coupling method, which occurs where the triangulation perfectly matches the edges of the source domain, as it can be seen in Fig. 5(a). This is the same test case already used in [5] to obtain the rate of convergence of the methods HL, ABB, JN and BMC. Specifically, by using this test it is shown that, under the aforementioned assumptions, the proposed coupling

method ensures a optimal rate of convergence, differently from the original HL method that exhibit a lower rate of convergence.

The *plasma case* resembles the typical free-boundary problem, where the domain of the plasma source term is not defined a priori. Since the plasma domain Γ_p is computed at every iteration of the non-linear problem, a triangulation which matches perfectly the boundary of the source domain would require to mesh the computational domain at every iteration, being not feasible from a computational point of view. An *Adaptive Integration Technique* (AIT) has been developed (see Appendix A) in order to compute the source term by properly integrating the triangles on the boundary Γ_p .

To evaluate the error we introduce a second region Ω_ϵ , as can be seen in Figs. 6-7-8 (green square). The error ϵ is measured in the L^2 -norm as:

$$\epsilon = \sqrt{\int_{\Omega_\epsilon} (\psi(\vec{r}) - \psi_R(\vec{r}))^2 d\Omega}, \quad (6.2)$$

where $\psi_R(\vec{r})$ is the reference value, obtained in different ways depending on the case and specified for each case in the related paragraph. In order to increase the accuracy, the integration of the quantities over triangles (e.g. for the computation of the stiffness matrix through (4.13) and (4.14)) has been done using high degree efficient symmetrical Gaussian quadrature rules [24].

6.1.1 Vacuum case

The source domain $\Omega_S = \Omega_C$ is a square region (blue in Fig. 6a) and the computational domain Ω_d (grey in Fig. 6a) is a rectangular region including Ω_ϵ (green in Fig. 6a) and Ω_C . The reference value $\psi_R(\vec{r})$ in (6.2) is obtained using semi-analytic high precision quadrature formulas for axi-symmetric sources with rectangular cross-section [34].

Fig. 6b, which reports the convergence obtained with basis functions of the first (cyan) and second (red) order, as a function of the mesh size h , shows that the proposed coupling method ensures a quadratic and cubic convergence respectively for the linear and quadratic basis functions. Comparing these results with those presented in [5], we can see an optimal rate of convergence like the one of the JN and ABB methods. Moreover, unlike BMC, the proposed method does not suffer from loss of convergence rate due to the presence of sharp corners on the computational boundary.

It is important to underline that the HL method has non-optimal rate of convergence [5]. This happens because the HL introduces an auxiliary variable u [5, (7)] to decrease the computational complexity. The 2D convolution integral in (4.8) is then replaced by a 1D convolution integral with the Neumann data of an auxiliary variable u , causing a loss of optimality in the convergence rate. Our approach does not rely on this approximation, keeping explicit the integral in (4.8), thus the optimality in the convergence rate is maintained.

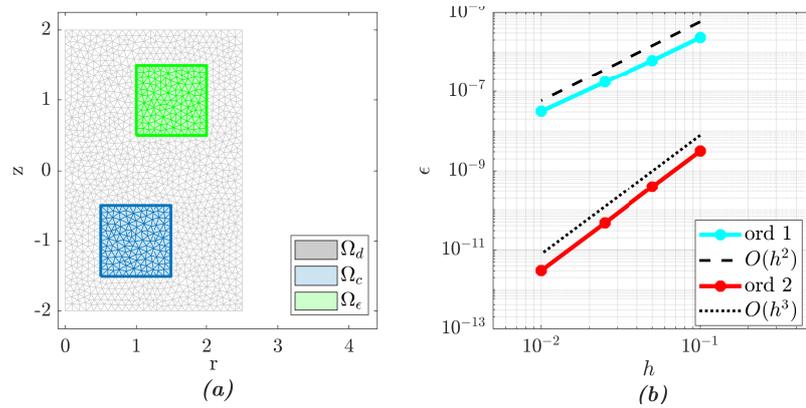


Figure 6: Vacuum case. (a): Regions Ω_d (grey), Ω_C (blue) and Ω_ϵ (green); (b) rates of convergence for linear and quadratic basis functions.

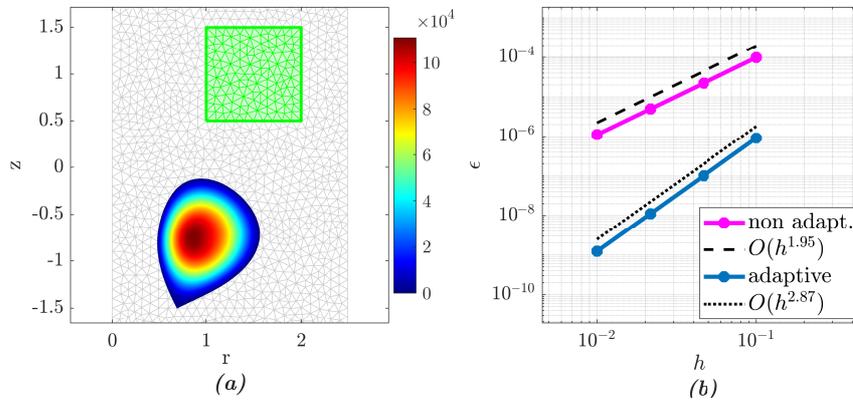


Figure 7: Plasma case. (a): Regions Ω_d (grey), Ω_C (monotonic current density profile) and Ω_ϵ (green); (b) rates of convergence (magenta=non-adaptive integration, blue=adaptive integration).

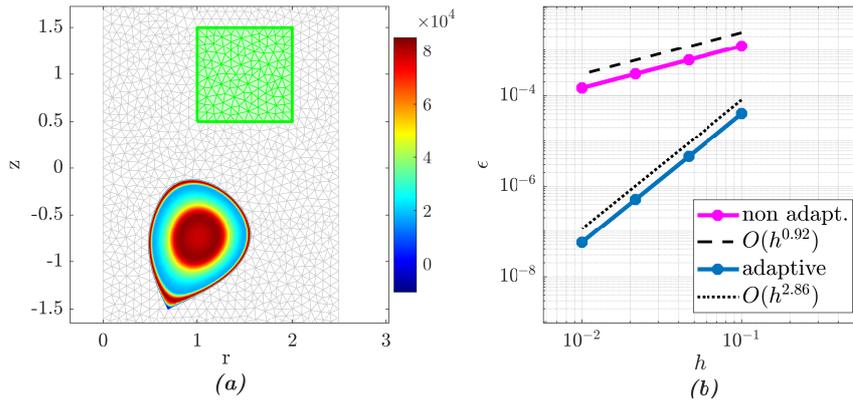


Figure 8: Plasma case. (a): Regions Ω_d (grey), Ω_C (current density profile with pedestal) and Ω_ϵ (green); (b) rates of convergence (magenta=non-adaptive integration, blue=adaptive integration).

6.1.2 Plasma case

The same convergence test of the previous paragraph is repeated considering as source $\Omega_S = \Omega_p$ a realistic plasma equilibrium with diverted configuration. Two different current density profiles are considered: the first is monotonic, while the latter exhibits a strong pedestal fraction in the proximity of the plasma separatrix (see respectively Fig. 7a and Fig. 8a). The reference solution is obtained by solving the same problem on a fine triangular mesh that perfectly matches Γ_p .

While in Section 6.1 the presented FEM-BEM coupling method has been proven to have optimal convergence rate (i.e. cubic rate), it is clear that this result is no longer valid if the triangulation does not match exactly the source boundary. However, we can see clearly that the proposed AIT is able to maintain a quasi-optimal rate of convergence (≈ 2.87), which is also independent from the distribution of the source current (blue lines in Figs. 7b and 8b).

Here it is reported also the comparison with a less sophisticated integration technique, obtained considering in the computation of the source term only the triangles with center off mass inside the boundary of Ω_p (see Fig. 5(b): red triangle is inside, grey triangle is outside). We observe a rate of convergence of ≈ 1.95 for the monotonic current density, and ≈ 0.95 for the current density with pedestal (magenta lines Figs. 7b and 8b), respectively. This means that, for the optimistic case of a plasma current which vanishes smoothly to the boundary, the rate of convergence is almost the same of a FEM model with linear basis functions. The performance worsen if a consistent fraction of the plasma current is close to the boundary.

6.2 Convergence of the non-linear problem

In this section, the results of a thorough numerical study on the convergence performance of Picard iteration and of more robust and reliable methods, such as Newton-Raphson/Newton-Krylov, are presented. As a test problem, a RFX-mod Upper Single Null plasma equilibrium is considered (i.e. shot no. 39122 at 0.85s, see Section 6.3 for further details). For this test case, the condition number the matrix $A(\mathbf{x})$ of (4.21) is $\kappa \approx 1.45 \times 10^6$, and the spectral radius of the jacobian of (5.1) is $\rho_J = 6.16$.

Fig. 9 reports the behaviour of Picard (9(a)) and Newton-Raphson. The same initial guess (blue curves) is used for both schemes. We can clearly see how much the solution provided by Picard is drifting in the outward direction (red curves), after a certain number of iterations, while Newton-Raphson does not exhibits such behaviour, but it monotonically converges. Fig. 9(c) gives a quantitative idea of the residue, defined as $\|F(\mathbf{x}_k)\|/\|\mathbf{x}_k\|$, versus the number of iterations: Newton-Raphson converges to almost machine working precision after 15 iterations, while Picard shows an unstable behaviour.

On the other hand, if the initial guess is not well chosen, the Newton-Raphson/Newton-Krylov schemes can suffer of an overshoot in the computation of $\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{h}_{k+1}$. This is most likely to happen during the very first iterations, in which the term \mathbf{h}_{k+1} is higher. These overshoots can be avoided using a *relaxed* (or *damped*) ver-

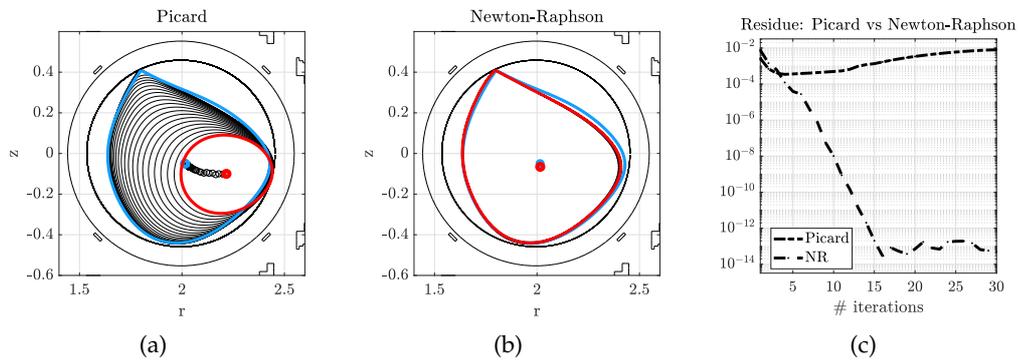


Figure 9: Picard (a) vs Newton-Raphson (b): plasma separatrix and centroid are shown during the iterations (blue = initial guess, red = after 30 iterations). Residues (c).

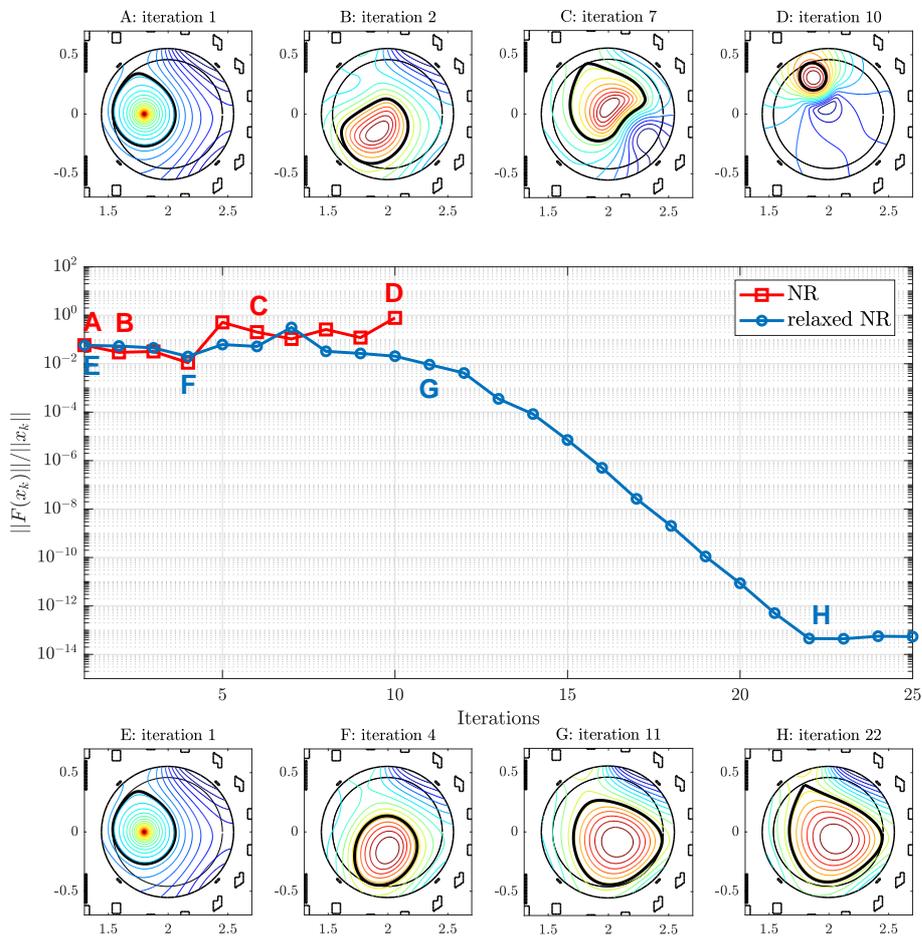


Figure 10: Impact of the relaxation parameter on the convergence. Solution at different iterations of standard NR (top) and relaxed NR (bottom) and residues (middle). Standard NR scheme stopped after 10 iterations.

sion of NR/NK, where the relaxation parameter α_k gives a weighted step $\alpha_k \mathbf{h}_{k+1}$. The value of α_k is chosen at every NR/NK iteration, by performing a preliminary check on the updated solution. Through this strategy it is possible to understand if an overshoot is occurring as well as to estimate its magnitude, in order to choose a suitable value of α_k to avoid it.

Fig. 10 investigates how the relaxation parameter affects the convergence of the relaxed NR with respect to the standard (not relaxed) NR, starting from the same initial guess, deliberately chosen non suitable. The solutions \mathbf{x}_{k+1} of for these two cases, at different iterations, are represented in terms of 2D contour map in Fig. 10. The standard NR (Fig. 10 top) is very sensitive to the initial guess, leading the iterative scheme to explore different branches of solutions, eventually stopping, after 10 iterations, without finding the correct solution (an internal check on the consistency of the solution is used as stopping criterion). On the other hand, convergence to the correct solution is achieved by tuning α_k for a weighted step $\alpha_k \mathbf{h}_{k+1}$, in order to have much more stable iterations (Fig. 10 bottom). Fig. 10 shows also the trend of the residues for both schemes.

The same considerations about relaxing the iteration step are valid also for Newton-Krylov, thus (5.13) becomes $\mathbf{h}_{k+1} = \alpha_k \mathbf{P}^{-1} \mathbf{y}$, by choosing α_k following the same strategy used for NR.

6.3 Numerical validation

Three experimental Single Null plasma equilibria achieved by RFX-mod, operating as low current tokamak, are analyzed using the FRIDA code. The main plasma equilibria parameters $[\beta_0, \alpha_M, \alpha_N]$, related to the parametrization of the current density in (4.6), are summarized in Table 1. These parameters are obtained with an optimization procedure to achieve the best match on the experimental magnetic field measurements [19, 35]. The free-boundary equilibrium solutions are shown in Figs. 11-13 in terms of poloidal magnetic flux surfaces, plasma current density and residual.

Table 1: Main plasma parameters of RFX-mod Single Null equilibria.

Shot #	I_p [A]	β_0	α_M	α_N
36922	58096	0.31	1.54	1.10
37829	50969	0.10	1.32	1.20
39122	63335	0.79	0.97	1.05

The trend of the residue over the iterations shows that the FRIDA code converges to a minimum of the function $F(\mathbf{x})$ within 10 to 15 iterations[‡], without drifting from this

[‡]In all the test cases, the tolerance has been set to machine working precision (e.g. 2.220410^{-16}) to evaluate how the obtained residual is close to the theoretical one (i.e. residual = 0). We underline that, despite the proposed Newton-like solvers can reach values of tolerance of the order of $10^{-(13\sim 15)}$, the solutions at two following steps of the nonlinear solver are practically indistinguishable already at residuals of the order of 10^{-8} .

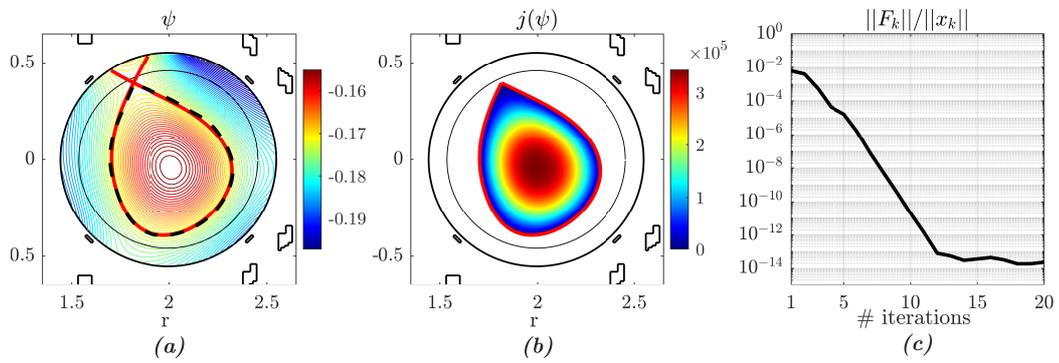


Figure 11: RFX-mod shot #36922: black dashed = reference boundary, red solid = FRIDA boundary. Flux (a), plasma current density (b) and residue (c).

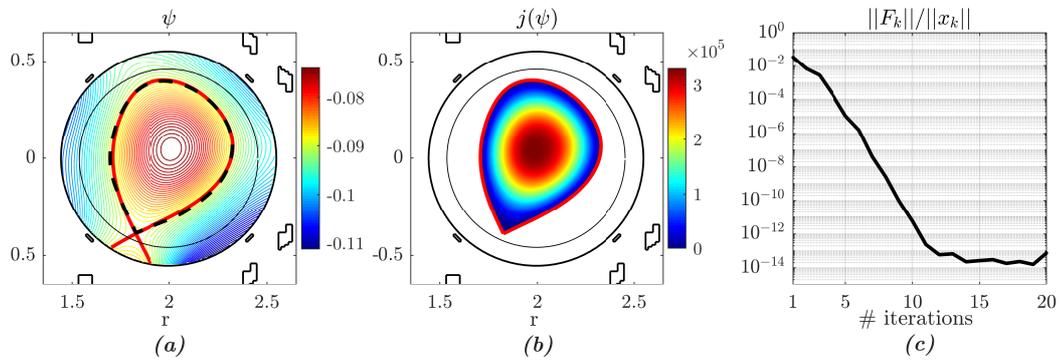


Figure 12: RFX-mod shot #37829: black dashed = reference boundary, red solid = FRIDA boundary. Flux (a), plasma current density (b) and residue (c).

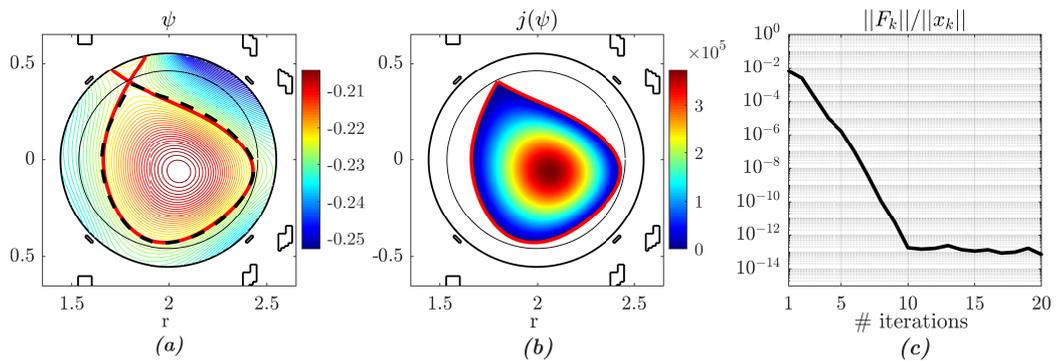


Figure 13: RFX-mod shot #39122: black dashed = reference boundary, red solid = FRIDA boundary. Flux (a), plasma current density (b) and residue (c).

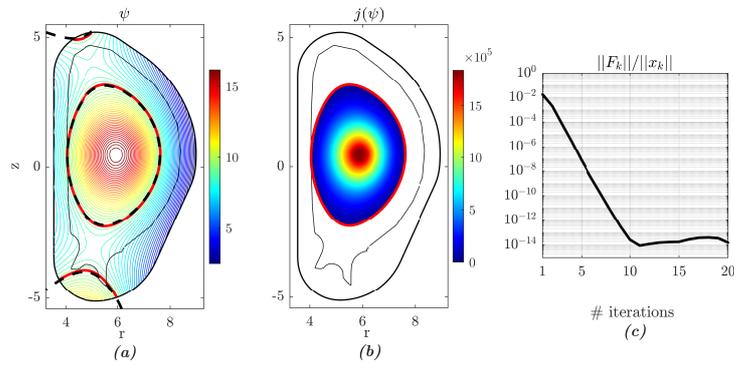


Figure 14: ITER-like case #1: black dashed = reference boundary, red solid = FRIDA boundary. Flux (a), plasma current density (b) and residue (c).

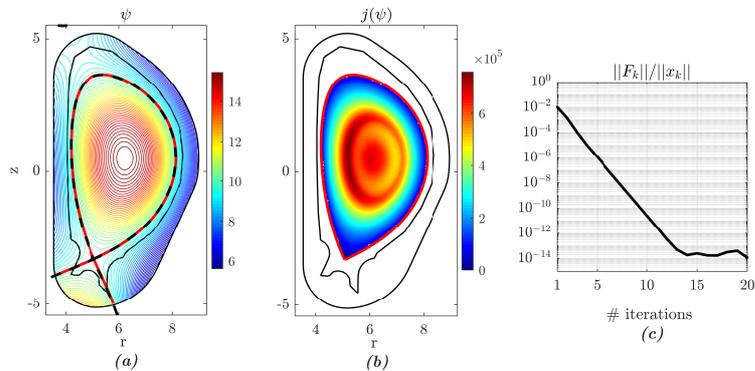


Figure 15: ITER-like case #2: black dashed = reference boundary, red solid = FRIDA boundary. Flux (a), plasma current density (b) and residue (c).

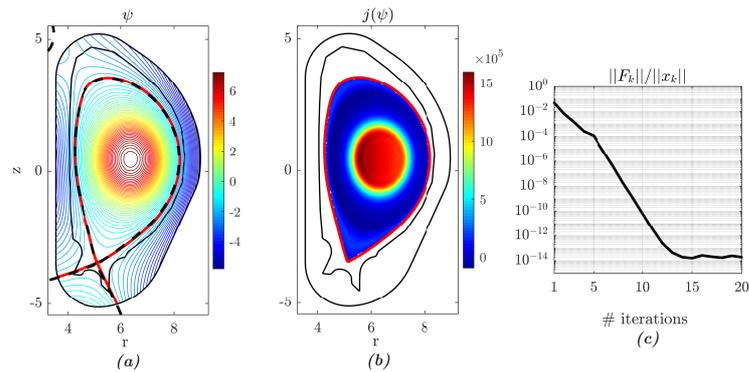


Figure 16: ITER-like case #3: black dashed = reference boundary, red solid = FRIDA boundary. Flux (a), plasma current density (b) and residue (c).

equilibrium point. This means that the code is robust and reliable, and does not suffer of the aforementioned *intrinsic axisymmetric instability*. It is clear that the plasma boundaries given by the FRIDA code (red) are in fairly good agreement with the experimental references (black dashed) reconstructed by extrapolating the poloidal flux in the vacuum region between the plasma and the magnetic measurements [36]

The same numerical tests has been performed on a ITER-like device. The analyzed equilibria involves one limiter and two diverted configurations, in order to assess the flexibility of the FRIDA code. As for RFX-mod, the code converges after 10÷15 iterations, and the boundaries are indistinguishable from the references, obtained with the codes CREATE-L [13] and CREATE-NL [14].

7 Conclusions

In this paper a coupled FEM-BEM approach for the solution of the free-boundary axisymmetric plasma equilibrium problem has been presented. The proposed method, obtained from an improvement of the Hagenow-Lackner (HL) coupling method, allows to handle the unbounded domain by reducing the computational domain only to the region inside the vacuum chamber. With this approach the external environment, including coils and conducting structures, can be described by means of Dirichlet boundary conditions on the surface of the limited computational domain. The external environment can be represented with both different level of complexity (i.e. 2D, 3D). These considerations would be very useful in simulating time-domain scenarios in which the 3D representation of the conductors is relevant (e.g. eddy currents problem, disruption events).

The paper explores different iterative methods for the solution of the non-linear Grad-Shafranov equation, such as Picard, Newton-Raphson and Newton-Krylov, in order to provide a robust and reliable tool, able to handle large-scale problems (e.g. high resolution equilibria). The convergence properties of each scheme are investigated leading to the result that the misconception of an "*intrinsic axisymmetric instability which is encountered in all equilibrium calculations with a free-boundary condition*" [10] arises from the use of the fixed-point/Picard scheme, which is well known to suffer for limitations on convergence [5]. On the other hand, Newton-type schemes (i.e. Newton-Raphson, Newton-Krylov) are the most robust and reliable, allowing to achieve a fast convergence to machine working precision. In particular, the relaxed (or damped) versions of these Newton-type schemes overshoots and/or instabilities in following the correct branch of solution, even when the initial guess is not well chosen. The robustness of Relaxed/Damped Newton's based scheme could be of fundamental use in modeling scenarios in which the non-linear plasma current density evolves in time following profiles given by multi-physics calculations (i.e. transport codes) which is a condition characterized by a much more pronounced non-linearity. Large-scale problems (i.e. high resolution equilibria) are efficiently handled by using the jacobian-free Newton-Krylov scheme, that revealed to be much faster with respect to the Newton-Raphson and also use a less

amount of memory.

A dedicated convergence analysis has shown that, unlike HL, this method exhibits optimal rate of convergence of the L^2 -norm of the error as the mesh size decreases. Moreover, an Adaptive Integration Technique (AIT) has been developed to compute the source term related to the plasma. This is mandatory if a suitable integration is required on a triangulation which does not match the source domain, as happens with this free-boundary equilibrium problem. The combination of this FEM-BEM coupling method together with the proposed adaptive integration technique gives a quasi-optimal convergence rate ($O(h^{2.87})$), even for free-boundary equilibrium problem characterized by a current density profile with a strong pedestal.

The proposed FEM-BEM approach has been implemented in the FRIDA code (Free-boundary Integro-Differential Axisymmetric – <https://github.com/matteobonotto/FRIDA>), together with the aforementioned iterative methods. The FRIDA code has been validated against both experimental (i.e. RFX-mod plasma discharges) and numerical data (i.e. ITER-like device).

Acknowledgments

The authors acknowledge at first Eng. Giuseppe Marchiori, for his important and deep review of the manuscript, and also Prof. Piergiorgio Alotto and Eng. Dimitri Voltolina for the interesting and helpful discussions about the treatment of some numerical aspects useful for the development of this work. The authors would also like to thank Prof. Massimiliano Mattei (CREATE) for providing the data needed for the comparison in the ITER-like case. This work was supported in part by Italian MIUR under PRIN grant 20177BZMAH.

Appendices

A Adaptive integration procedure for the computation of plasma source term

For a typical free-boundary problem, where the plasma domain Γ_p is computed at every iteration of the non-linear problem, it is not possible to have a triangulation which matches perfectly the boundary of the plasma domain. This would require, in principle, to mesh the computational domain at every iteration. This way of addressing the problem has undeniable drawbacks from the computational point of view, as it would require to compute at every iteration all the quantities defined in Sections 3-4. From a practical point of view this means that a certain number of triangles are crossed by the separatrix Γ_p , and thus, for each triangle, one can identify a region inside Γ_p (we will refer as Ω_p^∇ to the portion of each triangle inside Γ_p).

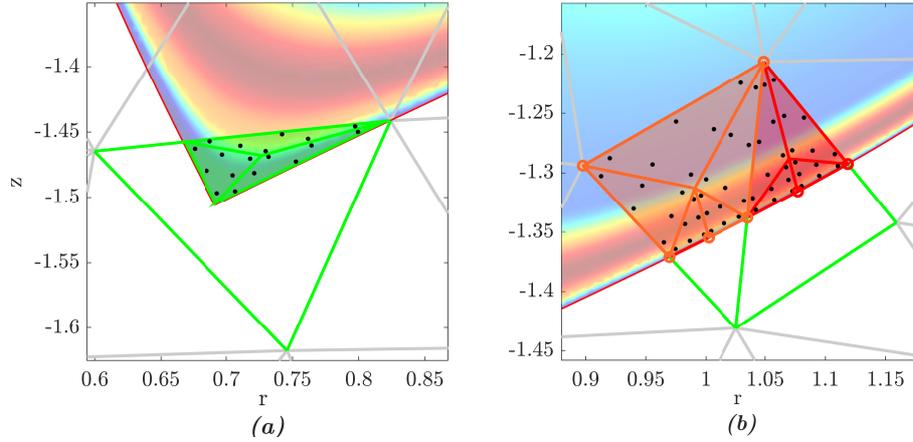


Figure 17: Decomposition of Ω_p^X into triangles: (a) triangle containing the X-Point, (b) different cases depending on how many vertices are inside Γ_p . Related Gauss points (black) are shown.

In order to compute the source term related to the plasma, an *adaptive* integration technique can be used for the triangles on the boundary. For each triangle on the boundary, the region Ω_p^X is evaluated, considering not only the intersection between the triangle and Γ_p , but also the midpoint of the curved arc given by the portion of Γ_p inside the triangle, in order to allow a better geometrical representation of the curved boundary. The resulting polygon describing Ω_p^X is then decomposed into triangles, in order to perform the integration with $N_{Gauss} = 6$ Gauss points for each triangle.

An example is reported in Figs. 17a-17b for the test problem used in paragraph 6.1.2 (current density with pedestal), where the region Ω_p^X can be:

1. quadrilateral with a curved edge if two vertices are inside Γ_p (red in Fig. 17b);
2. triangular with a curved edge if one vertex is inside Γ_p (orange in Fig. 17b);
3. triangular/quadrilateral if, in case of a diverted configuration, the triangle contains the X-Point (green in 17a).

Triangles close to that containing the X-Point should be treated carefully; in such case as also polygons like pentagon may arise.

This method was already proposed by Heumann [15] using 3-nodes triangles and 4-nodes quadrilaterals to approximate the region Ω_p^X , and resulting in first order local basis function. The integration of the source term inside Ω_p^X is performed using one Gauss node in both cases.

B Preconditioning techniques for the jacobian-free Newton-Krylov method

The use of a Preconditioned Newton-Krylov (NK) is mandatory, as the convergence of NK strongly depends on the preconditioner P_k used [31]. The purpose is to efficiently cluster eigenvalues of the iteration matrix, which in turn will reduce the required number of GMRES iterations [33]. Unlikely the common use of GMRES method to solve a linear system of equation, in which the system matrix is known a priori, in the NK method the system matrix is the jacobian, whose computation would be avoided. For this reason, NK preconditioners are usually obtained starting from A_k , i.e. the system matrix of Eq. (4.21). Traditionally, problem-independent standard iterative methods, such as Incomplete LU factorization (ILU), are applied to A_k when constructing a preconditioner [31], i.e. *Algebraic* Preconditioning (AP) [26]. On the other hand, a possible choice can be using a matrix which is an approximation of A_k , i.e. *Physics-Based* Preconditioning (PBP). Both AP and PBP have strengths and drawbacks; however, we will describe how these drawbacks have been overcome in this work in order to obtain high-performance preconditioners.

Regarding the AP, an effective technique is the *Drop-Tolerance* ILU (DT-ILU), already used in the solution of BEM problems [37]. DT-ILU establishes a continuum between the direct method and diagonal preconditioning, allowing the existence of an *optimal drop-tolerance* ϵ_{ilu}^{opt} which minimizes the total CPU time of GMRES [38]. Although a priori determination of the optimal drop tolerance remains an open problem [38], the approach followed in this work is to obtain *problem-specific* empirical trend to extrapolate the optimal drop-tolerance as a function of the number of Degrees of Freedom (DoFs).

Considering the PBP, the main limitation is imposed by the fact that the approximation $P_k \approx A_k$ should be sparse, while A_k is not completely sparse because of the BEM block (see Figs. 2 and 3). To overcome this issue we developed a *far-field sparsification* technique, in order to obtain A_k^* , a sparse version of A_k , satisfying the constrain $A_k^* \approx A_k$. The sparsified version of A_k is then used as preconditioner $P_k = A_k^*$. This technique is applied only to the block \hat{K} , i.e. the BEM block, and is based on neglecting interaction between *far* points, exploiting the fact that a node closer to a point of the computational boundary has a stronger effect on it if compared to nodes which are far away. This means to neglect the entries of each row of \hat{K} smaller than a factor α ($\alpha = 0$ means *no sparsification*, $\alpha = 1$ means *full sparsification*, i.e. only the entry with maximum absolute value per each row is maintained).

Since, as already said, for this specific problem the jacobian can be computed, the effectiveness of the preconditioners can be fully tested by considering also the cases of $P_k = J$ (true jacobian) and $P_k = J_k^*$ (far-field sparsification applied to J_k).

A preliminary comparison is shown in Fig. 18a, where a convergence analysis of GMRES with and without preconditioners is reported. It is clear that the preconditioner is essential to reduce the number of iteration of 2 orders of magnitude. On the other hand, as expected [39], GMRES preconditioned using the exact jacobian leads to convergence

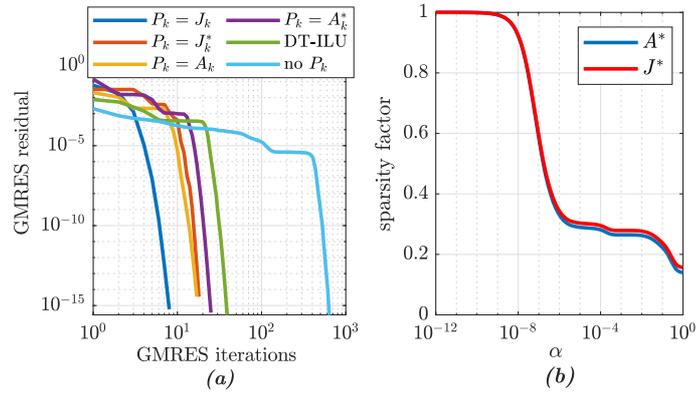


Figure 18: (a) convergence analysis of GMRES; (b) sparsity factor of A^* and J^* .

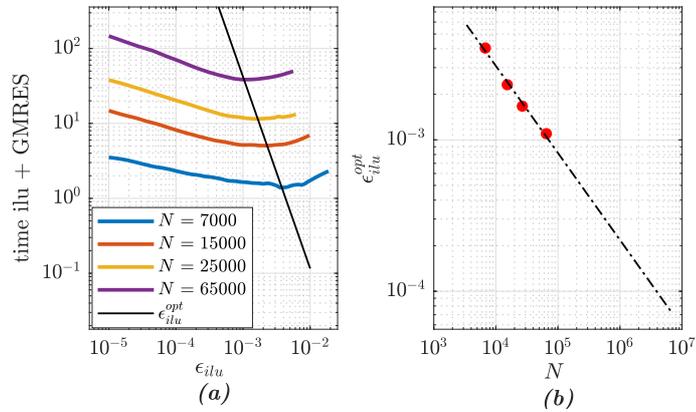


Figure 19: (a): total time (DT-ILU + GMRES) as a function of the drop-tolerance for 4 meshes with incremental refinement; (b) extrapolated optimal drop-tolerance as a function of N .

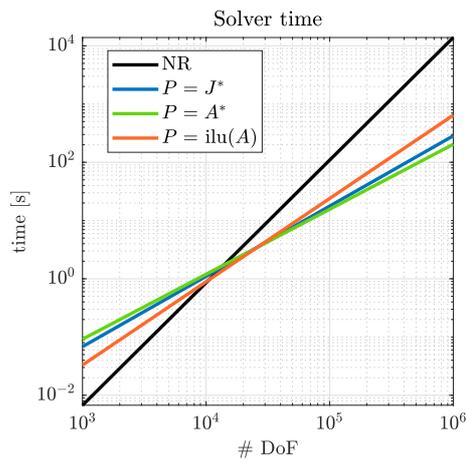


Figure 20: NR versus NK, time per iteration vs DoF (three preconditioners are considered).

Table 2: Timing of preconditioned GMRES.

Preconditioner	total GMRES time	time per GMRES iteration
$P_k = J_k$	7.25s	0.9063s
$P_k = J_k^*$	1.64s	0.0911s
$P_k = A_k$	11.31s	0.6653s
$P_k = A_k^*$	2.00s	0.0800s
$P_k = \text{ilu}(A_k)$	1.17s	0.0285s

with the least number of iterations. However, a fast convergence in terms of number of iterations does not necessarily means fast convergence in terms of total CPU time, as can be see from Table 2, which reports the total GMRES time. We can see how, a non fully sparse preconditioner (i.e. J_k, A_k) needs more computational time, being practically not appealing.

Fig. 18b shows the effectiveness of the *far-field sparsification* technique applied to the BEM block \hat{K} , reporting the sparsity factor of A^* (resp. J^*), defined as $\text{nnz}(A^*)/\text{nnz}(A)$, i.e., the ratio between the non-zero entries of A^* (resp. J^*) over the non-zero entries of A (resp. J). Fig. 18b shows an evident sparsification of the considered matrices as $\alpha \rightarrow 1$, proving that the block \hat{K} has $\approx 90\%$ of all the entries of the matrices.

Regarding the AP, an attempt to obtain *problem-specific* empirical trend of the optimal drop-tolerance of the DT-ILU is reported in Fig. 19. Specifically, Fig. 19a shows the total time per step of the non-linear solver, i.e. factorization (DT-ILU) plus solution (GMRES), as a function of the drop-tolerance ϵ_{ilu} and for 4 meshes with incremental refinement (number of nodes from $N = 7000$ to $N = 65000$). The minimum value of each curve gives the *experimental* optimal drop-tolerance $\epsilon_{\text{ilu}}^{\text{opt}}$. Fig. 19b shows an extrapolation, using a power regression, of the $\epsilon_{\text{ilu}}^{\text{opt}}$ as a function of the number of DoFs. This strategy gives a good *a priori* guess of $\epsilon_{\text{ilu}}^{\text{opt}}$. We underline that this approach is problem-specific, and a general way to a priori determine $\epsilon_{\text{ilu}}^{\text{opt}}$ still remains an open problem [38].

Fig. 20 shows how the time per iteration t_{tot} of the non-linear solver, i.e. the time needed to (i) compute the preconditioner plus (ii) to solve Eq. (5.12), scales with N . While, for Newton-Raphson (NR), $t_{\text{tot}} \propto N^{2.1}$, for NK we obtain:

- AP: $t_{\text{tot}} \propto N^{\approx 1.4}$ for $P_k = \text{ilu}(A_k)$;
- PBP $t_{\text{tot}} \propto N^{\approx 1.1}$ for $P_k = A_k^*, P_k^* = J_k$.

thus NR is preferable for small N but less appealing as N increases (e.g. a finer triangulation). On the other hand, Physic-Based Preconditioned NK scales almost linearly with N , which is slightly better that Algebraic Preconditioned NK. This is a remarkable result, showing how the far-field sparsification technique can be more appealing, with respect to DT-ILU with optimal drop-tolerance, if a very high number of DoFs is required.

References

- [1] J. Wesson and D. Campbell, *Tokamaks*. International Series of Monogr, OUP Oxford, 2011.
- [2] J. P. Freidberg, *Ideal MHD*. Cambridge University Press, 2014.
- [3] H. Grad and H. Rubin, "Hydromagnetic equilibria and force-free fields," in *Proc. Second UN Intern. Conf. on Peaceful Uses of Atomic Energy* 31, p. 190–197, 1959.
- [4] V. D. Shafranov, "Plasma Equilibrium in a Magnetic Field," *Reviews of Plasma Physics*, vol. 2, p. 103, Jan. 1966.
- [5] B. Faugeras and H. Heumann, "FEM-BEM coupling methods for tokamak plasma axisymmetric free-boundary equilibrium computations in unbounded domains," *Journal of Computational Physics*, vol. 343, pp. 201–216, 2017.
- [6] J. Blum, *Numerical simulation and optimal control in plasma physics*. United States: John Wiley and Sons Inc, 1989.
- [7] K. Lackner, "Computation of ideal MHD equilibria," *Computer Physics Communications*, vol. 12, no. 1, pp. 33–44, 1976.
- [8] F. Hofmann, "FBT – a free-boundary tokamak equilibrium code for highly elongated and shaped plasmas," *Computer Physics Communications*, vol. 48, no. 2, pp. 207–221, 1988.
- [9] P. Barabaschi, "The maxfea code," *Proceedings Plasma Control Technical Meeting*, 1993.
- [10] Y. M. Jeon, "Development of a free-boundary tokamak equilibrium solver for advanced study of tokamak equilibria," *Journal of the Korean Physical Society*, vol. 67, p. 843–853, Sep 2015.
- [11] R. Albanese, J. Blum, and O. D. Barbieri, "On the solution of the magnetic flux equation in an infinite domain," *EPS. 8th Europhysics Conference on Computing in Plasma Physics*, 1986.
- [12] R. Albanese, J. Blum, and O. D. Barbieri, "Numerical studies of the next european torus via the PROTEUS code," *12th Conf. on Numerical Simulation of Plasmas*, 1987.
- [13] R. Albanese and F. Villone, "The linearized CREATE-I plasma response model for the control of current, position and shape in tokamaks," *Nuclear Fusion*, vol. 38, pp. 723–738, may 1998.
- [14] R. Albanese, R. Ambrosino, and M. Mattei, "CREATE-NL+: A robust control-oriented free boundary dynamic plasma equilibrium solver," *Fusion Engineering and Design*, vol. 96-97, pp. 664–667, 2015. Proceedings of the 28th Symposium On Fusion Technology (SOFT-28).
- [15] H. Heumann, J. Blum, C. Boulbe, B. Faugeras, G. Selig, J. Anè, S. Bremond, V. Grandgirard, P. Hertout, and E. Nardon, "Quasi-static free-boundary equilibrium of toroidal plasma with cedres++: Computational methods and applications," *Journal of Plasma Physics*, vol. FirstView, pp. 1–35, 06 2015.
- [16] J. L. Johnson, H. Dalhed, J. Greene, R. Grimm, Y. Hsieh, S. Jardin, J. Manickam, M. Okabayashi, R. Storer, A. Todd, *et al.*, "Numerical determination of axisymmetric toroidal magnetohydrodynamic equilibria," *Journal of Computational Physics*, vol. 32, no. 2, pp. 212–234, 1979.
- [17] F. Villone, L. Barbato, S. Mastrostefano, and S. Ventre, "Coupling of nonlinear axisymmetric plasma evolution with three-dimensional volumetric conductors," *Plasma Physics and Controlled Fusion*, vol. 55, p. 095008, Jul 2013.
- [18] P. Piovesan, D. Bonfiglio, F. Auriemma, F. Bonomo, L. Carraro, R. Cavazzana, G. De Masi, A. Fassina, P. Franz, M. Gobbin, L. Marrelli, P. Martin, E. Martines, B. Momo, L. Piron, M. Valisa, M. Veranda, N. Vianello, B. Zaniol, M. Agostini, M. Baruzzo, T. Bolzonella, A. Canton, S. Cappello, L. Chacón, G. Ciaccio, D. F. Escande, P. Innocente, R. Lorenzini, R. Paccagnella, M. E. Puiatti, P. Scarin, A. Soppelsa, G. Spizzo, M. Spolaore, D. Terranova, P. Zanca, L. Zanotto, and M. Zuin, "RFX-mod: A multi-configuration fusion facility for three-

- dimensional physics studies," *Physics of Plasmas*, vol. 20, no. 5, p. 056112, 2013.
- [19] J. Luxon and B. Brown, "Magnetic analysis of non-circular cross-section tokamaks," *Nuclear Fusion*, vol. 22, pp. 813–821, Jun 1982.
- [20] C. Johnson and J. C. Nedelec, "On the coupling of boundary integral and finite element methods," *Mathematics of Computation*, vol. 35, no. 152, pp. 1063–1079, 1980.
- [21] J. BIELAK and R. C. MacCAMY, "An exterior interface problem in two-dimensional elastodynamics," *Quarterly of Applied Mathematics*, vol. 41, no. 1, pp. 143–159, 1983.
- [22] C. Geuzaine and J.-F. Remacle, "Gmsh: A 3-D finite element mesh generator with built-in pre- and post-processing facilities," *International Journal for Numerical Methods in Engineering*, vol. 79, pp. 1309–1331, 09 2009.
- [23] D. Pepper and J. Heinrich, *The Finite Element Method: Basic Concepts and Applications with MATLAB, MAPLE, and COMSOL, Third Edition*. Computational and Physical Processes in Mechanics and Thermal Sciences, CRC Press, 2017.
- [24] D. A. Dunavant, "High degree efficient symmetrical Gaussian quadrature rules for the triangle," *International Journal for Numerical Methods in Engineering*, vol. 21, no. 6, pp. 1129–1148, 1985.
- [25] A. Cenedese, P. Bettini, and M. Bonotto, "Model-based approach for magnetic reconstruction in axisymmetric nuclear fusion machines," *IEEE Transactions on Plasma Science*, vol. 46, pp. 636–644, March 2018.
- [26] C. T. Kelley, *Iterative Methods for Linear and Nonlinear Equations*. No. 16, SIAM, 1995.
- [27] S. Banach, "Sur les opérations dans les ensembles abstraits et leur application aux équations intégrales," *Fundamenta Mathematicae*, vol. 3, no. 1, pp. 133–181, 1922.
- [28] M. Crisfield, "Accelerating and damping the modified newton-raphson method," *Computers & Structures*, vol. 18, no. 3, pp. 395–407, 1984.
- [29] W. Xu and T. F. Coleman, "Solving nonlinear equations with the Newton-Krylov method based on automatic differentiation," *Optimization Methods and Software*, vol. 29, no. 1, pp. 88–101, 2014.
- [30] S. Jardin, *Computational Methods in Plasma Physics*. USA: CRC Press, Inc., 1st ed., 2010.
- [31] Y. Saad, *Iterative Methods for Sparse Linear Systems*. USA: Society for Industrial and Applied Mathematics, 2nd ed., 2003.
- [32] D. Knoll and D. Keyes, "Jacobian-free Newton-Krylov methods: a survey of approaches and applications," *Journal of Computational Physics*, vol. 193, no. 2, pp. 357–397, 2004.
- [33] D. Knoll, V. Mousseau, L. Chacon, and J. Reisner, "Jacobian-free Newton-Krylov methods for the accurate time integration of stiff wave systems," *Journal of Scientific Computing*, vol. 25, pp. 213–230, 11 2005.
- [34] L. Urankar, "Vector potential and magnetic field of current-carrying finite arc segment in analytical form, part iii: Exact computation for rectangular cross section," *IEEE Transactions on Magnetics*, vol. 18, pp. 1860–1867, November 1982.
- [35] D. Abate, G. Marchiori, and F. Villone, "Modelling and experimental validation of RFX-mod tokamak shaped discharges," *Fusion Engineering and Design*, vol. 146, pp. 135–138, 2019. SI:SOFT-30.
- [36] O. Kudlacek, P. Zanca, C. Finotti, G. Marchiori, R. Cavazzana, and L. Marrelli, "Real time measurement of plasma macroscopic parameters on RFX-mod using a limited set of sensors," *Physics of Plasmas*, vol. 22, no. 10, p. 102503, 2015.
- [37] J. H. Kane, D. E. Keyes, and K. G. Prasad, "Iterative solution techniques in boundary element analysis," *International Journal for Numerical Methods in Engineering*, vol. 31, no. 8, pp. 1511–1536, 1991.

- [38] N. Jovanovic, D. Keyes, and G. Prasad, "Drop tolerance ilu preconditioners for iterative solution techniques in boundary element analysis," in *Boundary Elements XV* (C. Brebbia and J. Rencis, eds.), pp. 501–516, Publ by Computational Mechanics Publ, 1993. Proceedings of the International Conference on Boundary Element Methods (BEM XV); Conference date: 10-08-1993 Through 13-08-1993.
- [39] L. Chacón, "An optimal, parallel, fully implicit Newton-Krylov solver for three-dimensional viscoresistive magnetohydrodynamics," *Physics of Plasmas*, vol. 15, no. 5, p. 056103, 2008.