A 3D FINITE VOLUME SCHEME FOR THE SIMULATION OF EDGE PLASMA IN TOKAMAKS *

M. BILANCERI¹, L. COMBE¹, H. GUILLARD¹, B. NKONGA¹ ET A. SANGAM¹

Résumé. Une méthode de volumes finis en coordonnées cylindriques pour la simulation de la région du bord dans les tokamaks est proposé. Contrairement aux méthodes classiques, qui sont basés sur la projection des équations en coordonnées curvilignes, une nouvelle méthode est proposée ici. Cette technique est basée sur la discrétisation de la forme vectorielle des équations et ensuite sur la projection sur les vecteurs de base du système curviligne associé à chaque volume de contrôle. La Méthodologie proposée a été validée sur plusieurs cas-tests simples et appliqué sur une simulation 3D d'un modèle MHD réduit.

Abstract. A finite volume method in cylindrical coordinates for the simulation of the edge region in tokamaks is proposed. In contrast to standard methods, which require the projected form of the equations in curvilinear coordinates, a new method is proposed. This technique is based on the discretization on the vector form of the equations and then on the projection on the basis of the curvilinear system associated to each control volume. The proposed methodology has been validated on several simple test cases and applied on a 3D simulation of a reduced MHD model.

INTRODUCTION

Due to the rarefaction of fossil energies as well as their adverse effects in term of environmental issues, the research of alternative energy sources is a major issue in the coming years. Among others, magnetically confined, controlled thermonuclear fusion is a promising possibility that leads to the construction of experimental reactors called Tokamaks (like ITER in Cadarache) to explore the feasibility of this concept. Many technical and conceptual difficulties have however to be overcome before the actual use of fusion energy. The recourse of numerical simulation methods will play a decisive role in this context allowing a better understanding of the physical phenomena and improvement in the design and performance of the machines. The objective of this work is to develop numerical models for the simulations of edge plasma in tokamak reactors in the framework of fluid modeling. From a purely numerical standpoint, the simulations of existing (TORE-SUPRA, JET, ...) and future (WEST, ITER, DEMO, ...) tokamak devices must now be able to consider realistic complex geometries. Due to the requirement to obtain quantitative prediction that can be used in the actual design of the machines, the use of simple finite difference approximations associated to analytic definition of the magnetic field lines is no more appropriate and more sophisticated numerical tools have to be developed [4]. In this work, we have chosen to concentrate on a finite volume approach that has potentially the capability to incorporate high order schemes with desired regularity of shape functions. Finite volume methods are specialized techniques to approximate systems of conservation laws. However, since tokamaks are axisymmetric devices, the use of the

* thanks to ANR ESPOIR which funds the work of M. Bilanceri and L. Combe

¹ Inria Sophia-Antipolis 2004 Route des Lucioles, BP 93, 06902 Sophia-Antipolis Cedex and Univ. Nice Sophia Antipolis, LJAD, UMR 7351, 06100 Nice, France

Cartesian coordinate system is not the most appropriate to the geometry of the problem. Unfortunately, the application of FV methods to curvilinear systems of coordinate is problematic because the space variation of the metric coefficients introduces artificial source terms. However it can be shown that whatever the curvilinear system used, a strong conservation form of the equations exists at the level of vector variables. Note that this strong conservation form do not exist at the level of the scalar components of the vectors in the curvilinear system due to the aforementioned space dependence of the metric coefficients. Based on this result, we report in this work on the design of an original Finite Volume technique that uses an approximation of the vector form of the equation followed by a local projection on the curvilinear system. This paper is organized as follows : Section 1 gives the basis of our approach and explain how to deal with curvilinear system of coordinate in a general way. Section 2 details this approach for the specific case of the toroidal geometry of tokamaks and gives the geometric computations necessary to implement the method. Section 3 presents the final finite volume method. Finally, in the last section, we apply this technique to a reduced MHD model.

1. THE DISCRETIZATION-PROJECTION METHOD

Let us consider a general hyperbolic conservation laws equation written in a coordinate free manner as

$$\frac{\partial \mathbf{W}}{\partial t} + \nabla \cdot \mathbf{F}(\mathbf{W}) = 0,$$

where $\mathbf{W}$ is the state variable and $\mathbf{F}(\mathbf{W})$ is its flux.

Let us also consider a curvilinear transformation $\phi : \xi \mapsto \mathbf{x}$, whose determinant of Jacobian is J . It is possible to show [1], that in this coordinate system, the above equation becomes:

$$\frac{\partial \mathbf{W}}{\partial t} + \frac{1}{J} \frac{\partial}{\partial \xi^k} \left(J \mathbf{F}(\mathbf{W}) \cdot \mathbf{e}^k \right) = 0.$$

where $\mathbf{e}^k$ are the contravariant basis vector associated to the transformation $\phi : \xi \mapsto \mathbf{x}$. Let us consider the special case with $\mathbf{W} = \mathbf{V}$ is a vector and $\mathbf{F}(\mathbf{W}) = \mathbf{T}$ is a tensor. The hyperbolic equation turns into :

$$\frac{\partial \mathbf{V}}{\partial t} + \frac{1}{J} \frac{\partial}{\partial \xi^k} \left(J \mathbf{T} \cdot \mathbf{e}^k \right) = 0. \quad (1)$$

The momentum equation in fluids dynamics is such a kind of equations.

Note that, since $\mathbf{V}$ is a vector, it has to be stored component by component on a given basis. The traditional approach consists in taking the scalar product of equation (1) by the covariant basis vectors $\mathbf{e}_k$ and then to obtain scalar equations for the components of the vector field $\mathbf{V}$. Finally these scalar equations are discretized. In the sequel, we will designate this method as the projection-discretization method. This approach has one important shortcoming : because the basis vectors are spatially dependent, they do not commute with the differential operators and therefore source terms appear in the equations [1,5]. The expression of these source terms depends on the specific curvilinear system used and their approximation is not obvious if some properties as for instance angular momentum conservation are required.

We therefore advocate the use of the following procedure that we will be called the discretization-projection method. First we define an auxiliary reference system associated to the i^{th} -cell. This auxiliary system is coincident with the curvilinear reference system in center of mass of the cell. The second step is to consider the integral formulation of (1), that is :

$$\frac{\partial}{\partial t} \int_{\mathcal{V}} \mathbf{V} d\mathcal{V} + \oint_{\mathcal{S}} \mathbf{T} \cdot \mathbf{n} d\mathcal{S} = 0 \quad (2)$$

where $\mathbf{n}$ denotes the unit external normal to the boundary $\mathcal{S}$ of the control volume $\mathcal{V}$. The third step is the discretization of (2) by the introduction of suitable numerical flux functions and quadrature rules, as shown in

Sec. 2. Finally the discretized version of (2) is projected on the auxiliary reference system by simply taking the dot product between the discretized equation and the auxiliary basis vectors.

Note that in the standard projection-discretization approach first the continuous equation is projected and the resulting scalar equations are discretized while in our discretization-projection method first we discretized a vector equation and then we project the results. This procedure is quite simple and it allows for a general (and implicit) discretization of the curvature terms.

2. TOROIDAL GEOMETRY AND MESH GENERATION

Since, the geometry of a tokamak is axisymmetric, it can be considered as the revolution of a plane, the poloidal plane, about an axis. The description of this kind of geometries, as well as the development of numerical methods, is much easier if, instead of the standard Cartesian system of coordinates, we use a cylindrical coordinate system.

2.1. System of coordinates

Let us consider a standard Cartesian system of coordinates (x, y, z) where the z -axis is coincident with the axis of revolution of the poloidal plane. The coordinates R, φ, Z are related to the standard Cartesian coordinates as

$$\begin{cases} x = R \cos \varphi \\ y = R \sin \varphi \\ z = Z \end{cases} \quad \text{or} \quad \begin{cases} R = \sqrt{x^2 + y^2} \\ \varphi = \begin{cases} 0 & \text{if } R = 0 \\ \arccos(x/R) & \text{if } y \geq 0 \\ 2\pi - \arccos(x/R) & \text{if } y < 0 \end{cases} \\ z = Z \end{cases} \quad (3)$$

Given the position vector $\mathbf{x} = (x, y, z)$, we will note $\mathbf{e}_R, \mathbf{e}_\varphi, \mathbf{e}_Z$ the covariant basis associated to the transformation (3), that is

$$\begin{cases} \mathbf{e}_R = \frac{\partial \mathbf{x}}{\partial r} = \frac{\partial x}{\partial r} \mathbf{i} + \frac{\partial y}{\partial r} \mathbf{j} + \frac{\partial z}{\partial r} \mathbf{k} \\ \mathbf{e}_\varphi = \frac{\partial \mathbf{x}}{\partial \varphi} = \frac{\partial x}{\partial \varphi} \mathbf{i} + \frac{\partial y}{\partial \varphi} \mathbf{j} + \frac{\partial z}{\partial \varphi} \mathbf{k} \\ \mathbf{e}_Z = \frac{\partial \mathbf{x}}{\partial z} = \frac{\partial x}{\partial z} \mathbf{i} + \frac{\partial y}{\partial z} \mathbf{j} + \frac{\partial z}{\partial z} \mathbf{k} \end{cases} \quad (4)$$

The Jacobian of the transformation is defined by

$$J = \mathbf{e}_R \cdot (\mathbf{e}_\varphi \times \mathbf{e}_Z) = R \quad (5)$$

The basis vector in (4) are not normalized and, thus, we will use the notation

$$\tilde{\mathbf{e}}_R = \mathbf{e}_R / |\mathbf{e}_R|, \quad \tilde{\mathbf{e}}_\varphi = \mathbf{e}_\varphi / |\mathbf{e}_\varphi|, \quad \tilde{\mathbf{e}}_Z = \mathbf{e}_Z / |\mathbf{e}_Z| \quad (6)$$

for unit vectors of the basis.

2.2. Common operations using the cylindrical system of coordinates

In the following we will consider some common mathematical operation that are needed in the development of the numerical scheme. In particular the standard Cartesian formulation and its equivalent formulation in the cylindrical coordinate system are considered. The first operation is the integration of a scalar function f over a volume $\mathcal{V}$:

$$\int_{\mathcal{V}} f d\mathcal{V} = \int_{\mathcal{V}(x,y,z)} f(x,y,z) dx dy dz = \int_{\mathcal{V}(R,\varphi,Z)} f(R,\varphi,Z) R dR d\varphi dZ \quad (7)$$

Note that using this approach the measure of a control volume, which will be denoted by $\mathcal{V}^{3D}$ is defined by:

$$\mathcal{V}^{3D} \doteq \int_{\mathcal{V}} d\mathcal{V} = \int_{\mathcal{V}(x,y,z)} dx dy dz = \int_{\mathcal{V}(R,\varphi,Z)} R dR d\varphi dZ \quad (8)$$

Also, the average value of a function over a control volume, f_{ave} , is simply defined by the following relation:

$$f_{ave} = \frac{\int_{\mathcal{V}} f d\mathcal{V}}{\int_{\mathcal{V}} d\mathcal{V}} = \frac{\int_{\mathcal{V}} f d\mathcal{V}}{\mathcal{V}^{3D}} \quad (9)$$

Let us consider a surface $\mathcal{S}$ in the (R, Z) -plane; for this surface it possible to define two different measures, the 2D measure, $\mathcal{S}_{2D}^{(R,Z)}$, and the “toroidal” one, $\mathcal{S}^{(R,Z)}$, which are defined by:

$$\mathcal{S}_{2D}^{(R,Z)} \doteq \int_{\mathcal{S}(R,Z)} dR dZ \quad \mathcal{S}^{(R,Z)} \doteq \int_{\mathcal{S}(R,Z)} R dR dZ = \mathcal{S}_{2D}^{(R,Z)} R_{g\mathcal{S}} \quad (10)$$

where $R_{g\mathcal{S}}$ is the R -coordinate of the center of mass of the surface $\mathcal{S}$.

FIGURE 1. Definition of the center of gravity of a surface in the poloidal plane

A kind of control volume which is particularly important for our case is the one which is obtained by the rotation of a surface $\mathcal{S}$ in the (R, Z) -plane around the Z -axis. Let define φ_1 and φ_2 as the starting and ending angles of revolution; the measure of this control volume can be easily calculated as follows:

$$\mathcal{V}^{3D} \doteq \int_{\mathcal{V}} d\mathcal{V} = \int_{\varphi_1}^{\varphi_2} \int_{\mathcal{S}(R,Z)} R dR dZ d\varphi = \Delta\varphi \mathcal{S}^{(R,Z)} \quad (11)$$

where $\Delta\varphi = \varphi_2 - \varphi_1$. Even for a segment $\mathcal{I}$ in the (R, Z) -plane it is possible to define two different measures, a 1D measure, $\mathcal{I}_{1D}^{(R,Z)}$ and a toroidal one, $\mathcal{I}^{(R,Z)}$, defined as follows:

$$\mathcal{I}_{1D}^{(R,Z)} \doteq \int_l dl \quad \mathcal{I}^{(R,Z)} \doteq \int_l R(l) dl = \mathcal{I}_{1D}^{(R,Z)} R_{g\mathcal{I}} \quad (12)$$

where l is the coordinate associated to the segment $\mathcal{I}$ and $R_{g\mathcal{I}}$ is the R -coordinate of its center of mass. A special kind of surfaces, $\mathcal{S}^{pol}$, which we will call poloidal surfaces, can be obtained by the rotation between the

angles φ_1 and φ_2 of a segment $\mathcal{I}$ in the (R, Z) plane. The measure of this surface, $\mathcal{S}^{(l, \varphi)}$, can be computed as follow:

$$\mathcal{S}^{(l, \varphi)} \doteq \int_{\mathcal{S}} d\mathcal{S} = \int_{\varphi_1}^{\varphi_2} \int_l R(l) dl d\varphi = \Delta\varphi \mathcal{I}^{(R, Z)} \quad (13)$$

Finally, let us consider a vector $\mathbf{V}$ which is independent on φ and whose non zero components are always in the (R, Z) -plane, that is:

$$\mathbf{V} = V_R(R, Z)\tilde{\mathbf{e}}_R + V_Z(R, Z)\tilde{\mathbf{e}}_Z \quad (14)$$

In the development of the numerical scheme it will be required to integrate this kind of vectors over a poloidal surface $\mathcal{S}^{pol}$. To compute this integral it is useful to consider an auxiliary Cartesian reference system defined as follows:

$$\mathbf{i} = \tilde{\mathbf{e}}_R(\bar{\varphi}), \quad \mathbf{j} = \tilde{\mathbf{e}}_{\varphi}(\bar{\varphi}), \quad \mathbf{k} = \tilde{\mathbf{e}}_Z, \quad \bar{\varphi} = \frac{\varphi_2 + \varphi_1}{2} \quad (15)$$

Using this auxiliary reference system the expression of the vector $\mathbf{V}$ becomes:

$$\mathbf{V} = \cos \tilde{\varphi} V_R(R, Z)\mathbf{i} + \sin \tilde{\varphi} V_R(R, Z)\mathbf{j} + V_Z(R, Z)\mathbf{k}, \quad \tilde{\varphi} = \varphi - \bar{\varphi} \quad (16)$$

And, thus the expression of the integral of $\mathbf{V}$ over $\mathcal{S}^{pol}$ becomes:

$$\begin{aligned} \int_{\mathcal{S}^{pol}} \mathbf{V} d\mathcal{S} &= \int_{-\frac{\Delta\varphi}{2}}^{\frac{\Delta\varphi}{2}} \int_l (\cos \tilde{\varphi} V_R(R, Z)\mathbf{i} + \sin \tilde{\varphi} V_R(R, Z)\mathbf{j} + V_Z(R, Z)\mathbf{k}) R dl d\tilde{\varphi} = \\ &= \int_l V_R(R, Z)\mathbf{i} \left(\sin \left(\frac{\Delta\varphi}{2} \right) - \sin \left(-\frac{\Delta\varphi}{2} \right) \right) R dl \\ &\quad - \int_l V_R(R, Z)\mathbf{j} \left(\cos \left(\frac{\Delta\varphi}{2} \right) - \cos \left(-\frac{\Delta\varphi}{2} \right) \right) R dl \\ &\quad + \int_l \Delta\varphi V_Z(R, Z)\mathbf{k} R dl = \int_l \left(2 \sin \left(\frac{\Delta\varphi}{2} \right) V_R(R, Z)\mathbf{i} + \Delta\varphi V_Z(R, Z)\mathbf{k} \right) R dl \\ &= \int_l \left(2 \sin \left(\frac{\Delta\varphi}{2} \right) V_R(R, Z)\tilde{\mathbf{e}}_R(\bar{\varphi}) + \Delta\varphi V_Z(R, Z)\tilde{\mathbf{e}}_Z \right) R dl \quad (17) \end{aligned}$$

where the last equivalent follows from (15) and the fact that $\tilde{\mathbf{e}}_R$ is independent on R and Z . Finally in the special case where V_R and V_Z are constants the integration reduces to:

$$\int_{\mathcal{S}^{pol}} \mathbf{V} d\mathcal{S} = \left(\frac{2}{\Delta\varphi} \sin \left(\frac{\Delta\varphi}{2} \right) V_R \tilde{\mathbf{e}}_R(\bar{\varphi}) + V_Z \tilde{\mathbf{e}}_Z \right) \mathcal{S}^{(l, \varphi)} \quad (18)$$

One important application of (18) is the computation of the (average) normal vector to a surface $\mathcal{S}^{pol}$. Indeed, in this case we can replace $\mathbf{V}$ with $\mathbf{n}_{\mathcal{I}}$, the normal to the segment from which $\mathcal{S}^{pol}$ is generated. Since $\mathcal{I}$ is a segment, $\mathbf{n}_{\mathcal{I}}$ is a constant in the (R, Z) plane. Thus, by integrating $\mathbf{n}_{\mathcal{I}}$ over the surface $\mathcal{S}^{pol}$ it is possible to define the (average) normal $\mathbf{n}$ to $\mathcal{S}^{pol}$ which, from (18) is

$$\mathbf{n} \mathcal{S}^{(l, \varphi)} = \left(\frac{2}{\Delta\varphi} \sin \left(\frac{\Delta\varphi}{2} \right) n_{\mathcal{I}, R} \tilde{\mathbf{e}}_R(\bar{\varphi}) + n_{\mathcal{I}, Z} \tilde{\mathbf{e}}_Z \right) \mathcal{S}^{(l, \varphi)} \quad (19)$$

2.3. Integration of a vector polynomial function

In this subsection we will compute the integration of a vector polynomial function using the cylindrical system of coordinates. First let us recall the following identities:

$$\frac{\partial \tilde{\mathbf{e}}_R}{\partial \varphi} = \tilde{\mathbf{e}}_{\varphi}, \quad \frac{\partial \tilde{\mathbf{e}}_{\varphi}}{\partial \varphi} = -\tilde{\mathbf{e}}_R \quad (20)$$

$$\begin{cases} \tilde{\mathbf{e}}_R(\alpha) - \tilde{\mathbf{e}}_R(-\alpha) = 2 \sin(\alpha) \tilde{\mathbf{e}}_\varphi(0) \\ \tilde{\mathbf{e}}_\varphi(\alpha) - \tilde{\mathbf{e}}_\varphi(-\alpha) = -2 \sin(\alpha) \tilde{\mathbf{e}}_R(0) \end{cases} \quad \begin{cases} \tilde{\mathbf{e}}_R(\alpha) + \tilde{\mathbf{e}}_R(-\alpha) = 2 \cos(\alpha) \tilde{\mathbf{e}}_R(0) \\ \tilde{\mathbf{e}}_\varphi(\alpha) + \tilde{\mathbf{e}}_\varphi(-\alpha) = 2 \cos(\alpha) \tilde{\mathbf{e}}_\varphi(0) \end{cases} \quad (21)$$

The integral that we want to compute is of the following form:

$$\int_{-\frac{\Delta\varphi}{2}}^{\frac{\Delta\varphi}{2}} \mathbf{f}(\varphi) d\varphi \quad (22)$$

assuming that $\mathbf{f}(\varphi)$ is a polynomial function in φ (and thus $\mathbf{f}(\varphi) \in C^\infty$). Note that the form (22) is quite general. Indeed $\mathbf{f}$ can be the results of an integration in the (R, Z) -plane such as $\mathbf{f}(\varphi) = \int_{(R, Z)} \mathbf{g}(R, \varphi, Z) dR dZ$. First, let us focus on the following special case:

$$\int_{-\frac{\Delta\varphi}{2}}^{\frac{\Delta\varphi}{2}} \tilde{\mathbf{e}}_R f(\varphi) d\varphi \quad (23)$$

By using (20)-(21) and by repeatedly integrating by part it is possible to obtain

$$\begin{aligned} \int_{-\frac{\Delta\varphi}{2}}^{\frac{\Delta\varphi}{2}} \tilde{\mathbf{e}}_R f(\varphi) d\varphi &= - \int_{-\frac{\Delta\varphi}{2}}^{\frac{\Delta\varphi}{2}} \frac{\partial \tilde{\mathbf{e}}_\varphi}{\partial \varphi} f(\varphi) d\varphi = -\tilde{\mathbf{e}}_\varphi f \Big|_{-\frac{\Delta\varphi}{2}}^{\frac{\Delta\varphi}{2}} + \int_{-\frac{\Delta\varphi}{2}}^{\frac{\Delta\varphi}{2}} \tilde{\mathbf{e}}_\varphi f'(\varphi) d\varphi = \\ &= -\tilde{\mathbf{e}}_\varphi f \Big|_{-\frac{\Delta\varphi}{2}}^{\frac{\Delta\varphi}{2}} + \int_{-\frac{\Delta\varphi}{2}}^{\frac{\Delta\varphi}{2}} \frac{\partial \tilde{\mathbf{e}}_R}{\partial \varphi} f'(\varphi) d\varphi = -\tilde{\mathbf{e}}_\varphi f \Big|_{-\frac{\Delta\varphi}{2}}^{\frac{\Delta\varphi}{2}} + \tilde{\mathbf{e}}_R f' \Big|_{-\frac{\Delta\varphi}{2}}^{\frac{\Delta\varphi}{2}} - \int_{-\frac{\Delta\varphi}{2}}^{\frac{\Delta\varphi}{2}} \tilde{\mathbf{e}}_R f''(\varphi) d\varphi = \dots \\ &= -\tilde{\mathbf{e}}_\varphi (f - f'' + f'''' - \dots) \Big|_{-\frac{\Delta\varphi}{2}}^{\frac{\Delta\varphi}{2}} + \tilde{\mathbf{e}}_R (f' - f''' + f'''' - \dots) \Big|_{-\frac{\Delta\varphi}{2}}^{\frac{\Delta\varphi}{2}} \end{aligned} \quad (24)$$

where $g|_\alpha^\beta$ means

$$g|_\alpha^\beta \doteq g(\beta) - g(\alpha) \quad (25)$$

Let us break down any product $\tilde{\mathbf{e}}_\star g|_\alpha^\beta$ where $\star$ can be R or φ in the following way :

$$\tilde{\mathbf{e}}_\star g|_\alpha^\beta = \bar{g} \Delta \tilde{\mathbf{e}}_\star + \bar{\mathbf{e}}_\star \Delta g \quad (26)$$

where

$$\bar{k} = \frac{k(\beta) + k(\alpha)}{2}, \quad \Delta k = k(\beta) - k(\alpha) \quad (27)$$

Using (26) and (21) it is possible to recast (24) as follows:

$$\begin{aligned} \tilde{\mathbf{e}}_R(0) &\left[2 \sin\left(\frac{\Delta\varphi}{2}\right) \bar{f} + \left(\cos\left(\frac{\Delta\varphi}{2}\right) \Delta f' - 2 \sin\left(\frac{\Delta\varphi}{2}\right) \bar{f}'' \right) \right. \\ &\quad \left. - \left(\cos\left(\frac{\Delta\varphi}{2}\right) \Delta f''' - 2 \sin\left(\frac{\Delta\varphi}{2}\right) \bar{f}'''' \right) + \dots \right] + \\ &\tilde{\mathbf{e}}_\varphi(0) \left[\left(2 \sin\left(\frac{\Delta\varphi}{2}\right) \bar{f}' - \cos\left(\frac{\Delta\varphi}{2}\right) \Delta f \right) - \left(2 \sin\left(\frac{\Delta\varphi}{2}\right) \bar{f}''' - \cos\left(\frac{\Delta\varphi}{2}\right) \Delta f'' \right) + \dots \right] \end{aligned} \quad (28)$$

Note that the general term $\left(2 \sin\left(\frac{\Delta\varphi}{2}\right)\bar{f}^{k+1} - \cos\left(\frac{\Delta\varphi}{2}\right)\Delta f^k\right)$ is at least a second-order term as function of $\Delta\varphi$

$$\begin{aligned} \lim_{\Delta\varphi \rightarrow 0} \frac{1}{\Delta\varphi} \left(2 \sin\left(\frac{\Delta\varphi}{2}\right)\bar{f}^{k+1} - \cos\left(\frac{\Delta\varphi}{2}\right)\Delta f^k\right) = \\ \lim_{\Delta\varphi \rightarrow 0} \left(\frac{2}{\Delta\varphi} \sin\left(\frac{\Delta\varphi}{2}\right)\bar{f}^{k+1} - \cos\left(\frac{\Delta\varphi}{2}\right)\frac{\Delta f^k}{\Delta\varphi}\right) = \lim_{\Delta\varphi \rightarrow 0} \left(\bar{f}^{k+1} - \frac{\Delta f^k}{\Delta\varphi}\right) = 0 \end{aligned} \quad (29)$$

Indeed it is of the order $\Delta\varphi^2$. If a first-order or a second-order scheme is considered, this terms can be safely neglected in the integration process but for high-order methods (more than second-order) they should be taken into account. Using the same approach it should be straightforward to compute the integral of

$$\int_{-\frac{\Delta\varphi}{2}}^{\frac{\Delta\varphi}{2}} \tilde{\mathbf{e}}_\varphi f(\varphi) d\varphi \quad (30)$$

Finally we can compute the integral of a general vector polynomial function by simply using the identity

$$\int_{-\frac{\Delta\varphi}{2}}^{\frac{\Delta\varphi}{2}} \mathbf{f}(\varphi) d\varphi = \int_{-\frac{\Delta\varphi}{2}}^{\frac{\Delta\varphi}{2}} \tilde{\mathbf{e}}_R f_R(\varphi) d\varphi + \int_{-\frac{\Delta\varphi}{2}}^{\frac{\Delta\varphi}{2}} \tilde{\mathbf{e}}_\varphi f_\varphi(\varphi) d\varphi + \int_{-\frac{\Delta\varphi}{2}}^{\frac{\Delta\varphi}{2}} \tilde{\mathbf{e}}_Z f_Z(\varphi) d\varphi \quad (31)$$

2.4. Mesh generation

The generations of the mesh, which we will use in our code, exploit the features of tokamaks geometries. Since the geometry of the poloidal plane could be arbitrary complex, as a first step an unstructured 2D dual mesh is generated. First, the considered computational domain (the poloidal plane), is divided into N_t triangles. Let T_j denotes the j^{th} -triangle; the i^{th} finite-volume cell V_i , associated with the i^{th} -vertex is given by

$$V_i = \bigcup_j V_i^{(j)}$$

where the union is extended to all the triangles sharing the i^{th} -vertex and $V_i^{(j)}$ represents the subset of T_j which is defined by further dividing T_j into six sub-triangles by means of its medians and subsequently considering those two sub-triangles which share the i^{th} -vertex (see figure 2). As shown in Fig. 2, the interface between the i^{th} -cell and the j^{th} -one is composed by two segments, $\Gamma_{ij,1}$ and $\Gamma_{ij,2}$. Once a 2D mesh is generated the 3D mesh is generated by the revolution of the poloidal mesh about the axis of revolution. In practice the interval $[0, 2\pi]$ is divided into N_{plan} segments defined by the points $(\varphi_1, \varphi_2, \dots, \varphi_{N_{plan}})$. For each 2D control volume V_i we obtain N_{plan} 3D control volumes $\mathcal{V}_i^{3D}$ which are obtained by the rotation of V_i between the angles φ_k, φ_{k+1} (A similar argument holds true even for the triangles of the 2D mesh). Note that the boundary of the generic control volume $\mathcal{V}_i^{3D}$ can be divided into two kinds of surfaces, the poloidal surfaces $\mathcal{S}^{pol}$ and the toroidal ones, $\mathcal{S}^{tor}$. The poloidal surfaces are obtained by the revolution of the segments $\Gamma_{ij,1}$ and $\Gamma_{ij,2}$ of the poloidal mesh. The toroidal surfaces are always two and are the images of the revolved 2D control volume V_i .

3. NUMERICAL DISCRETIZATION

In this section we will consider the discretization of the isothermal Euler model:

$$\begin{cases} \frac{\partial n}{\partial t} + \nabla \cdot n\mathbf{u} = 0 \\ \frac{\partial n\mathbf{u}}{\partial t} + \nabla \cdot (n\mathbf{u} \otimes \mathbf{u}) + \nabla p = \mathbf{0} \end{cases} \quad (32)$$

FIGURE 2. Generation of the dual Mesh.

where c^2 is the square of the (constant) sound speed,

A Finite-Volume formulation is considered to discretize the convective part of system (32) and, thus, it is convenient to recast system (32) in the following integral form

$$\frac{\partial}{\partial t} \int_{\mathcal{V}} n d\mathcal{V} + \oint_{\mathcal{S}} n \mathbf{u} \cdot \mathbf{n} d\mathcal{S} = 0 \quad (33a)$$

$$\frac{\partial}{\partial t} \int_{\mathcal{V}} n \mathbf{u} d\mathcal{V} + \oint_{\mathcal{S}} (n \mathbf{u} \otimes \mathbf{u}) \cdot \mathbf{n} d\mathcal{S} + \oint_{\mathcal{S}} p \mathbf{n} d\mathcal{S} = \mathbf{0} \quad (33b)$$

which, by using a standard finite-volume discretization over the i^{th} control volume becomes

$$\mathcal{V}_i^{3D} \frac{\partial}{\partial t} n_i + \sum_j \int_{\mathcal{S}_{ij}} f_n(W_i, W_j, \mathbf{n}_{ij}) d\mathcal{S} = 0 \quad (34a)$$

$$\mathcal{V}_i^{3D} \frac{\partial}{\partial t} n_i \mathbf{u}_i + \sum_j \int_{\mathcal{S}_{ij}} f_{\mathbf{u}}(W_i, W_j, \mathbf{n}_{ij}) d\mathcal{S} = \mathbf{0} \quad (34b)$$

where the summation over j is extended to all the neighbors of the control volume and $W_i = (n_i, n \mathbf{u}_i)^T$. Note in particular that since our 3D control volume is obtained by the revolution of a 2D surface we can distinguish two different kinds of surfaces, the poloidal and the toroidal ones. Thus, in the following it will be shown that it is useful to split the summation in system (34) as

$$\mathcal{V}_i^{3D} \frac{\partial}{\partial t} n_i + \sum_{j \in \mathcal{S}^{pol}} \int_{\mathcal{S}_{ij}} f_n(W_i, W_j, \mathbf{n}_{ij}) d\mathcal{S} + \sum_{j \in \mathcal{S}^{tor}} \int_{\mathcal{S}_{ij}} f_n(W_i, W_j, \mathbf{n}_{ij}) d\mathcal{S} = 0 \quad (35a)$$

$$\mathcal{V}_i^{3D} \frac{\partial}{\partial t} n_i \mathbf{u}_i + \sum_{j \in \mathcal{S}^{pol}} \int_{\mathcal{S}_{ij}} f_{\mathbf{u}}(W_i, W_j, \mathbf{n}_{ij}) d\mathcal{S} + \sum_{j \in \mathcal{S}^{tor}} \int_{\mathcal{S}_{ij}} f_{\mathbf{u}}(W_i, W_j, \mathbf{n}_{ij}) d\mathcal{S} = \mathbf{0} \quad (35b)$$

As usual, in Finite-Volume methods, it is necessary to define the surface integrals as a function of the two states on the left and on right of the interfaces. In particular in this work we considered the standard HLL

scheme [2]. Once a numerical flux function is defined, system (35) can be recast as

$$\mathcal{V}_i^{3D} \frac{\partial}{\partial t} \begin{pmatrix} n_i \\ n\mathbf{u}_i \end{pmatrix} + \sum_{j \in \mathcal{S}^{pol}} \int_{\mathcal{S}_{ij}} F(W_i, W_j, \mathbf{n}_{ij}) d\mathcal{S} + \sum_{j \in \mathcal{S}^{tor}} \int_{\mathcal{S}_{ij}} F(W_i, W_j, \mathbf{n}_{ij}) d\mathcal{S} = \mathbf{0} \quad (36)$$

The last step to completely define the spatial discretization of the convective fluxes is to describe the strategy employed to compute the surfaces integrals in (36). Since we want to exploit the curvilinear coordinate system associated to the toroidal geometry, the flux function will be expressed in the cylindrical coordinate system. Considering the definitions and properties demonstrated in Sec. 2.3 the computation of the surface integrals should be straightforward. In particular, if the flux function F_{ij} is considered constant, the result of the integration over a poloidal surface and toroidal surface is given by

$$\int_{\mathcal{S}^{pol}} F(W_i, W_j, \mathbf{n}_{ij}) d\mathcal{S} = \mathcal{S}^{(l,\varphi)} \begin{pmatrix} F_n \\ \frac{2}{\Delta\varphi} \sin \frac{\Delta\varphi}{2} (F_{u_R} \mathbf{e}_R(i) + F_{u_\varphi} \mathbf{e}_\varphi(i)) + F_{u_Z} \mathbf{e}_Z(i) \end{pmatrix} \quad (37a)$$

$$\int_{\mathcal{S}^{tor}} F(W_i, W_j, \mathbf{n}_{ij}) d\mathcal{S} = \mathcal{S}^{(R,Z)} \begin{pmatrix} F_n \\ F_{u_R} \mathbf{e}_R(i) + F_{u_\varphi} \mathbf{e}_\varphi(i) + F_{u_Z} \mathbf{e}_Z(i) \end{pmatrix} \quad (37b)$$

where $\mathbf{e}_R(i)$, $\mathbf{e}_\varphi(i)$, $\mathbf{e}_Z(i)$ are the basis vectors associated to the i^{th} -control volume. Note however that the proposed methodology is not restricted to constant flux function. Indeed by using the expressions in Sec. 2.3 it is possible to exactly integrated any polynomial function and, thus, it is possible to increase the order of the method. By using (37b) into (36) the resulting semi-discrete formulation is

$$\mathcal{V}_i^{3D} \frac{\partial}{\partial t} \begin{pmatrix} n_i \\ n\mathbf{u}_i \end{pmatrix} + \sum_{j \in \mathcal{S}^{pol}} \mathcal{S}_{ij}^{(l,\varphi)} \begin{pmatrix} F_{ij,n} \\ F_{ij,u_R} \frac{2}{\Delta\varphi} \sin \frac{\Delta\varphi}{2} \\ F_{ij,u_\varphi} \frac{2}{\Delta\varphi} \sin \frac{\Delta\varphi}{2} \\ F_{ij,u_Z} \end{pmatrix} + \sum_{j \in \mathcal{S}^{tor}} \mathcal{S}_{ij}^{(R,Z)} \begin{pmatrix} F_{ij,n} \\ F_{ij,u_R} \\ F_{ij,u_\varphi} \\ F_{ij,u_Z} \end{pmatrix} = \mathbf{0} \quad (38)$$

3.1. Second-order extension

To achieve the second-order extension in space a MUSCL-like technique as been employed [3]. First, a gradient is associated to each control volume, then at each interfaces $\mathcal{S}$ a reconstructed state is computed as

$$W_{ij} = W_i + \nabla W_i \cdot (\mathbf{s}_{ij} - \mathbf{g}_i) \quad W_{ji} = W_j + \nabla W_j \cdot (\mathbf{s}_{ij} - \mathbf{g}_j) \quad (39)$$

where $\mathbf{g}_i$ and $\mathbf{g}_j$ are the centers of mass of the control volumes and $\mathbf{s}_{ij}$ is the midpoint of the segment joining $\mathbf{g}_i$ and $\mathbf{g}_j$ (which is on the surface by construction). Finally, the flux function is computed as a function of the reconstructed states, that is $F_{ij} = F(W_{ij}, W_{ij}, \mathbf{n}_{ij})$ instead of the first order method $F_{ij} = F(W_i, W_j, \mathbf{n}_{ij})$. To compute the gradient ∇W_i first the compute a non-limited gradient by exploiting the features of our mesh which is unstructured in the poloidal plane (R, z) and structured in the toroidal direction. As a consequence, the gradient in the toroidal direction is computed using a 1D centered approximation

$$(\nabla W_i)_\varphi = \frac{1}{r_i} \frac{W_{i_n} - W_{i_p}}{\varphi_{i_n} - \varphi_{i_p}} \quad (40)$$

where i_n and i_p are the indexes of the control volumes, images of the i^{th} one, which belong to the next and previous poloidal plane. Instead, the gradient in the poloidal plane is computed in finite-elements fashion, that

is

$$(\nabla W_i)^{(R,Z)} = \frac{\sum_j \mathcal{S}_{T_j}^{(R,Z)} \nabla W_i|_{T_j}}{\sum_j \mathcal{S}_{T_j}^{(R,Z)}} \quad (41)$$

where the summation is extended to all the triangular elements to which the i^{th} node belongs, $\mathcal{S}_{T_j}^{(R,Z)}$ is the surface in the (R, Z) of the triangle and $\nabla W_i|_{T_j}$ is the gradient restricted to the triangle T_j computed using a standard P1 finite-element approximation. Finally the gradient considered in (39) is obtained by applying a Minmod like limiter to the previous computed non-limited gradient.

3.2. Discretization of the diffusion term

In this section we consider a parabolic system of the following form

$$\begin{cases} \frac{\partial n}{\partial t} = \nabla \cdot (\mu_n D \nabla n) \\ \frac{\partial n u_{\parallel}}{\partial t} = \nabla \cdot (\mu_u D \nabla (n u_{\parallel} \mathbf{b})) \end{cases} \quad \text{with} \quad (42)$$

The diffusion terms in (42) can be recast as surface integrals as

$$\int_{\mathcal{V}} \nabla \cdot (\mu_n D \nabla n) d\mathcal{V} = \int_{\mathcal{S}} \mu_n (D \nabla n) \mathbf{n} d\mathcal{S} \quad (43)$$

with a similar expression for the momentum equation. Since in general in Finite-Volume methods the solution is only required to be piecewise continuous, it is necessary to define a gradient on the surfaces $\mathcal{S}$. In this work we chose to use the following definition of the gradient

$$\int_{\mathcal{S}_{ij} \cap T_k} \mu_n (D \nabla n) \mathbf{n}_{ij} d\mathcal{S} = \int_{\mathcal{S}_{ij} \cap T_k} \mu_n (D \nabla n|_{T_k}) \mathbf{n}_{ij} d\mathcal{S} \quad (44)$$

where $\mathcal{S}_{ij} \cap T_k$ is the intersection between the surface $\mathcal{S}_{ij}$ and the triangle T_k and $\nabla n|_{T_k}$ is the gradient computed using a standard P1 finite element approximation, as in the previous subsection.

3.3. Time advancing

For the time advancing we considered an explicit scheme. The first-order time advancing is obtained by simple using the Euler method, that is

$$\begin{pmatrix} n_i \\ n u_{\parallel, i} \end{pmatrix}^{n+1} = \begin{pmatrix} n_i \\ n u_{\parallel, i} \end{pmatrix}^n + \frac{\Delta t}{\mathcal{V}_i^{3D}} \text{RHS}^n \quad (45)$$

where RHS^n is the contribution of the convective and diffusive fluxes computed using the solution at time n . The second-order accuracy in time is obtained by using the a TVD RK2 [6] scheme, that is

$$\begin{pmatrix} n_i \\ n u_{\parallel, i} \end{pmatrix}^{n+1/2} = \begin{pmatrix} n_i \\ n u_{\parallel, i} \end{pmatrix}^n + \frac{\Delta t}{\mathcal{V}_i^{3D}} \text{RHS}^n \quad (46a)$$

$$\begin{pmatrix} n_i \\ n u_{\parallel, i} \end{pmatrix}^{n+1} = \begin{pmatrix} n_i \\ n u_{\parallel, i} \end{pmatrix}^n + \frac{\Delta t}{2\mathcal{V}_i^{3D}} (\text{RHS}^n + \text{RHS}^{n+1/2}) \quad (46b)$$

4. NUMERICAL EXPERIMENTS

The numerical method described in the previous sections has been validated using several different test cases. A first test case is introduced in order to validate the discretization of the diffusion terms. Then an axisymmetric simulation using the JET geometry has been considered..

4.1. Validation of the viscous terms

In order to validate the discretization of the viscous term two different test cases have been considered, a radial diffusion test case and a toroidal diffusion one. In both test-cases only isotropic diffusion is considered, that is we consider the following reduced system:

$$\begin{cases} \frac{\partial n}{\partial t} = \nabla \cdot (\mu_n \nabla n) \\ \frac{\partial n \mathbf{u}}{\partial t} = \nabla \cdot (\mu_u \nabla (n u_{\parallel} \mathbf{b})) \end{cases} \quad (47)$$

The computational domain restricted to the poloidal plane is a rectangle in the region $20 \leq R \leq 120, 0 \leq Z \leq 10$ and it is meshed using 4696 points. In the computational domain 20 equidistant poloidal planes are used, as show in Fig. 3.

(A) Poloidal Plane

(B) Toroidal Plane

FIGURE 3. Mesh used for the diffusive test-cases.

4.1.1. Radial diffusion problem

In this test case we want to solve system (47) using the following boundary conditions

$$n|_{R_{min}} = 1, \quad n|_{R_{max}} = 10, \quad \frac{\partial n}{\partial Z}\bigg|_{Z_{min}} = 0, \quad \frac{\partial n}{\partial Z}\bigg|_{Z_{max}} = 0 \quad (48)$$

and $\mathbf{u}$ is imposed equal to zero on all the boundaries. For this problem the analytic stationary solution is given by $\mathbf{u} = 0$ and

$$n(R) = C_1 \ln \left(\frac{R}{R_{min}} \right) + C_2 \quad \Rightarrow \quad n(R) = \frac{9}{\ln 6} \ln \left(\frac{R}{20} \right) + 1 \quad (49)$$

where the second expression is the one obtained using the boundary conditions in (48). We started our simulation with the analytic solution and we waited until the residual is below 10^{-7} . In Fig. 4 the solution computed by the numerical method and the error with the analytic one are shown. It is possible to see the good agreement between the reference and computed solution, thus validating the implementation of the diffusive terms in the poloidal plane.

FIGURE 4. Results for the radial diffusive test-case.

4.1.2. Toroidal diffusion problem

In this test case we consider a modified computational domain defined between the angles $\varphi_{min} \leq \varphi \leq \varphi_{max}$ with $\varphi_{min} = 0$ and $\varphi_{max} = \frac{19}{10}\pi$. The boundary conditions applied to (47) are the following

$$n|_{\varphi_{min}} = 1, \quad n|_{\varphi_{max}} = 10, \quad \frac{\partial n}{\partial Z}\bigg|_{R_{min}} = 0, \quad \frac{\partial n}{\partial Z}\bigg|_{R_{max}} = 0, \quad \frac{\partial n}{\partial Z}\bigg|_{Z_{min}} = 0, \quad \frac{\partial n}{\partial Z}\bigg|_{Z_{max}} = 0 \quad (50)$$

and $\mathbf{u}$ is imposed equal to zero on all the boundaries. For this problem the analytic stationary solution is given by $\mathbf{u} = 0$ and

$$n(\varphi) = C_1 + C_2(\varphi - \varphi_{min}) \quad \Rightarrow \quad n(\varphi) = 1 + 9 \frac{\varphi - \varphi_{min}}{\varphi_{max} - \varphi_{min}} \quad (51)$$

Note that (51) is a linear solution and, thus, our method should be able to compute the exact solution up to machine precision. Indeed we used the analytic solution as initial condition for our code and it preserve this solution, as shown in Fig. 5, thus validating also the discretization of the toroidal diffusive terms.

FIGURE 5. Results for the toroidal diffusive test-case.

4.2. A reduced MHD system

In this test-case we consider the simulation of the reduced MHD model:

$$\left\{ \begin{array}{l} \frac{\partial n}{\partial t} + \nabla \cdot n \mathbf{u} = \nabla \cdot (\mu_n D \nabla n) \\ \frac{\partial n u_{\parallel}}{\partial t} + \nabla \cdot (n u_{\parallel} \mathbf{u}) + \nabla_{\parallel} p = \nabla \cdot (\mu_u D \nabla (n u_{\parallel} \mathbf{b})) + n \mathbf{u} \cdot \frac{D \mathbf{b}}{Dt} \end{array} \right. \quad \text{with} \quad \left\{ \begin{array}{l} \mathbf{u} = u_{\parallel} \mathbf{b} + \frac{\mathbf{B} \times \nabla \Phi}{B^2} \\ \Phi = \ln \left(\frac{n}{n_0} \right) \\ p = c^2 n, \quad c = \sqrt{\frac{T_e}{m_i}} \end{array} \right. \quad (52)$$

where T_e is the (constant) temperature of the electrons, m_i is the ion mass, $\mathbf{B}$ is the magnetic field, $\mathbf{b}$ is the direction of the magnetic field. Φ is the electric potential evaluated thanks to the adiabatic assumption $\Phi = \ln \left(\frac{n}{n_0} \right)$ where n_0 is the average of n over a magnetic surface. The quantity $\frac{\mathbf{B} \times \nabla \Phi}{B^2}$ is the electric drift velocity. Finally μ_n and μ_u are two diffusion coefficients and D is a diffusion tensor defined as

$$D = I - \mathbf{b} \otimes \mathbf{b}^T \quad (53)$$

where I is the identity tensor. Note that the physical meaning of D is that the diffusion is highly anisotropic; in particular there is no diffusion in the direction parallel to the magnetic field. System (52) can be scaled using the following reference values

$$\tilde{\mathbf{x}} = \frac{\mathbf{x}}{L_{ref}}, \quad \tilde{n} = \frac{n}{n_{ref}}, \quad \tilde{u}_{\parallel} = \frac{u_{\parallel}}{c}, \quad \tilde{t} = t \frac{c}{L_{ref}}, \quad \tilde{\Phi}_e = \Phi \frac{e}{T_e}, \quad \tilde{\mathbf{B}} = \frac{\mathbf{B}}{B_0}$$

and, as consequence, we can derive the following adimensionalized quantities

$$\tilde{\mathbf{u}} = \tilde{u}_{\parallel} \mathbf{b} + \rho^* \frac{\tilde{\mathbf{B}} \times \nabla \tilde{\Phi}}{\tilde{B}^2}, \quad \rho^* = \frac{m_i c}{e B_0 L_{ref}}, \quad \tilde{\mu}_n = \frac{\mu_n}{L_{ref} c}, \quad \tilde{\mu}_u = \frac{\mu_u}{L_{ref} c},$$

Note in particular the expression of the adimensionalized velocity; the adimensionalized factor ρ^* , which is the adimensionalized Larmor radius, multiplies the term associated to the drift velocity. Indeed this shows that ρ^* is an indicator of the ratio between the parallel and drift components of the velocity. Given the results of section 2, the discretization of this model is straightforward. It is clear that the momentum equation for the parallel velocity $u_{\parallel}$ is nothing but the projection of the vector momentum equation on the direction of the magnetic field. Therefore it is sufficient to project the momentum equation on the direction parallel to the magnetic field. Indeed, by defining $\mathbf{b}_i$ the direction of the magnetic field in the control volume, it follows

$$\mathbf{u} \cdot \mathbf{b}_i = \left(u_{\parallel,i} \mathbf{b}_i + \frac{\mathbf{B}_i \times \nabla \Phi_i}{B_i^2} \right) \cdot \mathbf{b}_i = u_{\parallel,i} \quad (54)$$

and, thus using the results of section 3, we obtain from (38)

$$\mathcal{V}_i^{3D} \frac{\partial}{\partial t} n_i + \sum_{j \in \mathcal{S}^{pol}} \mathcal{S}_{ij}^{(l,\varphi)} F_{ij,n} + \sum_{j \in \mathcal{S}^{tor}} \mathcal{S}_{ij}^{(R,Z)} F_{ij,n} = 0 \quad (55a)$$

$$\mathcal{V}_i^{3D} \frac{\partial}{\partial t} n u_{\parallel,i} + \mathbf{b}_i \cdot \left(\sum_{j \in \mathcal{S}^{pol}} \mathcal{S}_{ij}^{(l,\varphi)} \begin{pmatrix} F_{ij,u_R} \frac{2}{\Delta\varphi} \sin \frac{\Delta\varphi}{2} \\ F_{ij,u_\varphi} \frac{2}{\Delta\varphi} \sin \frac{\Delta\varphi}{2} \\ F_{ij,u_Z} \end{pmatrix} + \sum_{j \in \mathcal{S}^{tor}} \mathcal{S}_{ij}^{(R,Z)} \begin{pmatrix} F_{ij,u_R} \\ F_{ij,u_\varphi} \\ F_{ij,u_Z} \end{pmatrix} \right) = 0 \quad (55b)$$

that provides a complete description of the numerical algorithm.

In the considered test case the magnetic geometry contains an X point and the 2D mesh is aligned on the field lines in the poloidal plane. Note however that our method do not require any specific arrangement for this case. The mesh of the poloidal plane consists of 3564 points while 50 poloidal planes are considered. As for the diffusion coefficients, μ_n and μ_u , we have used the values $\mu_n = 1 \text{ m}^2/\text{s}$ and $\mu_u = 0.1 \text{ m}^2/\text{s}$. Finally the value of ρ^* is set equal to $3 \cdot 10^{-4}$ resulting in a drift velocity of the order of some percents of the speed of sound, typically the ones experimentally found in the SOL. Considering the boundary conditions, on the inner surface a constant gradient of density, equal to -1 , and $u_{\parallel} = 0.5c_s$ are imposed. On the outer surface two limiters are present where Bohm boundary conditions are imposed. Bohm boundary conditions require that the flow goes out of the computational domain with a speed at least equal to the speed of sound. On the remaining of the outer surface standard slip conditions are imposed. Note that using this set of boundary conditions and a constant initial condition the resulting simulations are axisymmetric.

In the following plots the results of the simulation are shown. On the left and right side of Fig. 6 the u and n profiles of the computed solution are shown. The density results in logarithmic scale clearly show the presence of the separatrix : in the interior region the density is high while outside it quickly decreases near zero. The results for the parallel velocity clearly show how the flow is accelerated towards by the Bohm boundary conditions. Indeed in Fig. 7 the Mach number is shown. Fig. 7a shown the Mach number and again the strong acceleration due to the Bohm boundary conditions is obvious. Fig. 7b shows only the Mach number of the drift component of the velocity vector: this component is negligible almost everywhere except in the private flux region and in the weak field side where the magnetic poloidal field is stronger.

5. CONCLUSIONS

We have devised a finite volume method in cylindrical coordinates able to simulate the edge regions in tokamaks. In contrast to standard methods, this technique do not use the projected form of the equations

FIGURE 6. Computed results for the parallel velocity (left) and density (right) profiles, the results for the density are in logarithmic scale.

that have the drawback to introduce curvature source terms in the continuous system of equation. Instead, this technique applies the discretization on the vector form of the equations where a strong conservation form does exist and then project the resulting discrete system on the basis of the curvilinear system defined on each control volume. Thus numerical methods have been validated on several simple test cases and applied on a 3D simulation of a reduced MHD model using the adiabatic assumption to express the electric potential. In the future, we plan to consider more complex MHD models and apply this method to the study of MHD instabilities and micro-turbulence in edge plasmas.

REFERENCES

- [1] A. Bonnement, T. Fajraoui, H. Guillard, M. Martin, A. Mouton, B. Nkonga, and A. Sangam. Finite volume method in curvilinear coordinates for hyperbolic conservation laws. *ESAIM: PROCEEDINGS*, 32:163–176, 2011.
- [2] A. Harten, P.D. Lax, and B. van Leer. "On upstreaming differencing and Godunov-type schemes for hyperbolic conservation laws". *SIAM Rev.*, 25:35–61, 1983.
- [3] B. van Leer. "Towards the ultimate conservative difference scheme V: a second-order sequel to Godunov's method". *Journal of Computational Physics*, 32(1):101–136, 1979.
- [4] Jardins, S. "Computational methods in plasma physics". *Chapman & Hall CRC*, Computational sciences series, 2010.
- [5] Vinokur, M. "Conservation equations of gasdynamics in curvilinear coordinate systems". *Journal of Computational Physics*, 14:105–125, 1974.
- [6] S. Gottlieb and C. W. Shu. "Total variation diminishing Runge-Kutta schemes". *Mathematics of Computation*, 67:73–85, 1998.

FIGURE 7. Comparison of the Mach number for the velocity and the drift component of the velocity.