

Advanced spatial discretizations in the B2.5 plasma fluid code

Klingshirn, H.-J.^{a,*}, Coster, D.P.^a, Bonnin, X.^b,

^a*Max-Planck-Institut für Plasmaphysik, EURATOM Association, Garching, Germany*

^b*LSPM-CNRS, Université Paris 13, F-93430 Villetaneuse, France*

Abstract

Plasma wall interaction modeling for fusion devices requires consistent plasma backgrounds, which are typically provided by SOL codes like SOLPS 5 [1]. However, the computational domain of these codes typically does not cover the far scrape-off layer, resulting in large uncertainties in the modeling of the interaction between the plasma and the main chamber wall. The work presented here addresses this issue by extending the B2.5 plasma fluid code to provide plasma backgrounds up to the device wall.

Key words: SOLPS, Divertor Modelling, Edge Modelling

JNM keywords: T0100

PSI-20 keywords: B2/EIRENE, Divertor Modelling, Edge Modelling

PACS: 52.65.Kj, 02.60.Cb, 02.70.-c

* Corresponding author: H.-J. Klingshirn, Max-Planck-Institut für Plasmaphysik, Boltzmannstr 2, Garching bei Muenchen, D85748 Germany; hmk at ipp.mpg.de

1 Introduction

Understanding and predicting the plasma transport in the scrape-off layer (SOL) of magnetic confinement fusion devices is a major concern in the operation and design of experiments and future fusion power plants. The physics in the SOL region is highly complex due to the interplay of plasma and neutrals transport, atomic processes and plasma-wall interaction. Especially the latter is important due to the possibly extremely high energy fluxes deposited on the device wall, the effects of which can have a severe impact on the allowable operational regimes and the device lifetime.

During the last three decades, sophisticated computational models for the SOL plasma have been developed. For toroidally symmetric devices like Tokamaks, an established approach is to combine a 2d fluid (CFD) description for the ion species with a kinetic model for the neutrals. The B2-EIRENE code (aka SOLPS [1]) in its various versions is an example for such a coupled model.

A design decision in the original version of B2 was to limit the computational domain of the code to extend only some limited distance away from the separatrix in the radial direction (i.e. it only covers the near SOL and some annulus of the core, c.f. figure 1). For modeling power deposition in the divertor, where

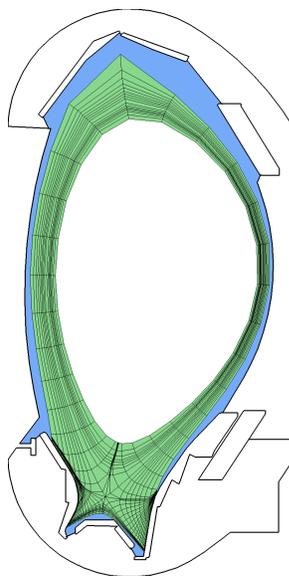


Fig. 1. The standard simulation domain for the SOLPS 5 covers the SOL (green). The region up to the wall (blue) is not covered.

the region of interest is located around the strike point, this is a valid choice which allows simplification of the code design, resulting in significant speedup of both code development and simulation times. The drawback is that no plasma solution is available for the far SOL and at the main chamber and PFR walls. The missing data can be computed using a dedicated code (see

e.g. [2]), but from a technical point of view an integrated solution is preferable.

One of the simplifications originally exploited in the B2 code family is the use of logically rectangular field-aligned curvilinear grids. The physical domain is split along the separatrix (or multiple separatrices in case of a double null topology) into SOL, core and private flux regions, which are then mapped onto rectangular regions in the computational space. Every region is discretized using a structured curvilinear quadrilateral grid which can be stored efficiently on 2-dimensional arrays. To achieve exact separation of the highly anisotropic parallel and radial plasma transport, cell edges along the poloidal direction are aligned to flux surface and radial edges are chosen to be orthogonal to them. However, because the number of grid points along a poloidal grid line is fixed, but the poloidal extent of the regions varies with the radial coordinate, the grid orthogonality condition has to be relaxed when approaching the target plates. The resulting non-orthogonal grids (c.f. figure 3a) pose challenges for the formulation of numerical schemes. This problem becomes more severe the further one moves away from the separatrix, making this type of grid unsuitable for covering the far SOL.

An approach to combine the advantages of structured grids with complex geometries that is used successfully for various CFD problems (see e.g. [3]) are “cut-cell” type grids. In a first step a standard structured grid covering the region of interest is set up, ignoring the details of the problem geometry which is typically defined by geometric surfaces. Subsequently, the intersections of the grid cells with the surfaces are computed. The resulting grid consists of standard cartesian cells in most of the domain, boundary cells with a more complex geometry at the surfaces and possibly unused cells that fall outside of the actual region of interest.

Successful application of the cut-cell approach was reported for the SOLPS 4 code [4,5]. The work presented here introduces it to the SOLPS 5 code version, with emphasis on an automatized grid generation chain. It can be separated into two areas: automatic generation of cut-cell type grids for tokamak geometries (section 2) and an upgrade of the B2.5 plasma fluid code to support the resulting more complex grids (section 3).

2 CARRE 2 grid generator

The SOLPS 5 suite of codes includes the dedicated grid generator CARRE [6]. Starting with magnetic equilibrium information reconstructed from experimental measurements and a 2d line-segment model of the device wall, it generates logically rectangular curvilinear quadrilateral field-aligned grids for SOL simulations. At the heart of the code is an iterative algorithm that optimizes the placement of grid nodes on flux surfaces based on a number of grid

quality criteria, one of which is grid orthogonality.

CARRE 2 is an extension of the original CARRE code, enabling it to create cut-cell type grids covering the entire plasma vessel up to the device wall. To achieve this the standard CARRE algorithm is embedded into a more complex algorithm (c.f. figure 2):

- (1) Use the standard CARRE algorithm to identify topology, targets and limiting surfaces.
- (2) Derive a *virtual geometry* from the device geometry.
- (3) Apply the standard CARRE algorithm to the virtual geometry to obtain an initial standard grid.
- (4) Iterate until no cells with critical geometry exist:
 - (a) Compute intersection of cell edges and real device geometry.
 - (b) Identify cells with critical geometry.
 - (c) Add or remove radial grid lines to fix the identified defective cells.
- (5) Finalization:
 - (a) Adjust geometry of cut cells so that edges align on the device structure.
 - (b) Categorize grid objects (cells, edges, nodes)

The geometry of a cell is classified as critical if it has more than four edges (i.e., it's not a quadrilateral or triangle) or if user-defined resolution constraints are violated along the poloidal direction. This mechanism can be used by the user to force the grid to a specific poloidal grid resolution at designated structure elements (e.g. the divertor targets). Grid points are added using the same optimization procedure from the standard algorithm, which ensures orthogonality of the resulting grid. Note that an important consequence of this iterative algorithm is that (other than in the original CARRE algorithm) the number of grid points in the poloidal direction is not known in advance.

The final classification step determines whether grid cells are located inside, outside or at boundary of the computational domain. For every boundary edge the index of the corresponding device structure is stored. For edges not located on a structure (e.g. the core boundary) canonical indices are derived. This information can be used later for the definition of boundary conditions.

CARRE 2 can be operated in several modes:

- (1) *Standard mode*: replicates the behaviour of the original CARRE code, with the orthogonality condition relaxed at the target plates (c.f. fig. 3a) and no far SOL coverage.
- (2) *Target mode*: the radial extent of the grid is the same as in standard mode, but cut cells are created at the target plates. The resulting grid is fully orthogonal (c.f. fig. 3b).
- (3) *Vessel mode*: the radial extent of the grid is chosen to cover the whole plasma vessel, with cut cells being generated along the device wall (c.f.

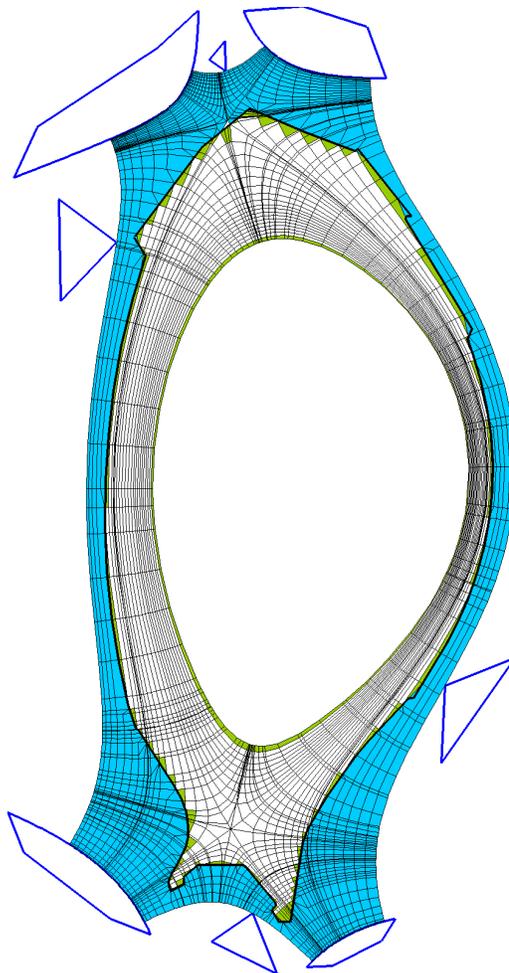


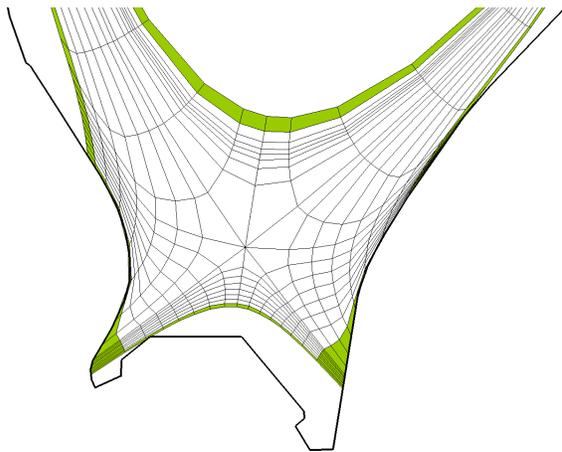
Fig. 2. Grid generation for a disconnected double null case, with the upper x-point outside of the vacuum vessel. Virtual geometry structures are shown in cyan, real geometry structures are black. Cell colors: white = internal, green = boundary, blue = external.

fig. 3c).

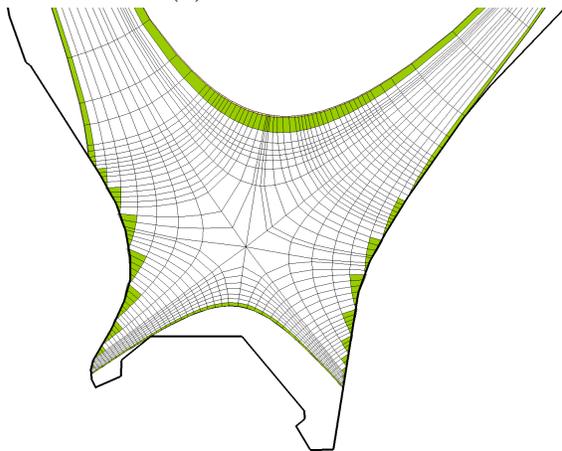
CARRE 2 is embedded in the SOLPS 5 grid generation chain (c.f. figure 5), which provides the DivGeo GUI to set up the geometry. The virtual geometry created by CARRE 2 can be imported in DivGeo to fine-tune grid generation parameters (fig. 4).

3 B2.5-extended plasma fluid code

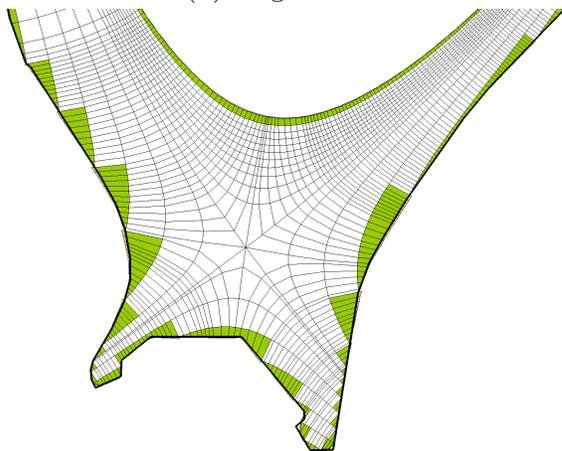
With extended grids up to the first wall now being readily available, the next step is to update the plasma and neutrals codes in the SOLPS 5 package to make use of them. While the EIRENE neutrals code already can handle arbitrary geometries with quadrilateral and triangular grids, some work was



(a) standard mode



(b) target mode



(c) vessel mode

Fig. 3. CARRE 2 operation modes

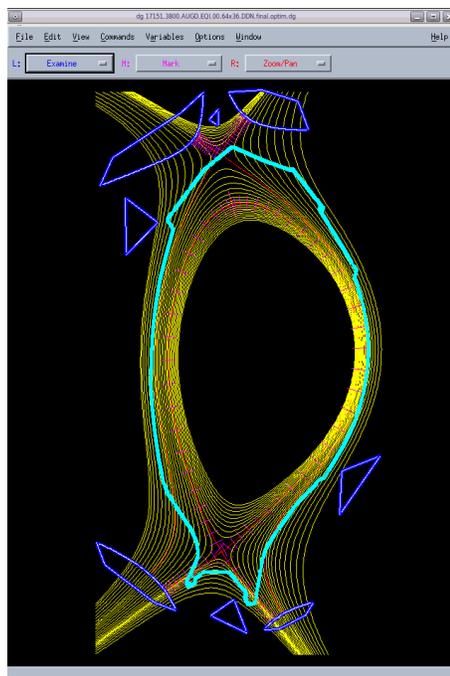


Fig. 4. Using the DivGeo geometry GUI to set up parameters for the CARRE 2 run shown in figure 2. Cyan=real device geometry, blue=virtual geometry

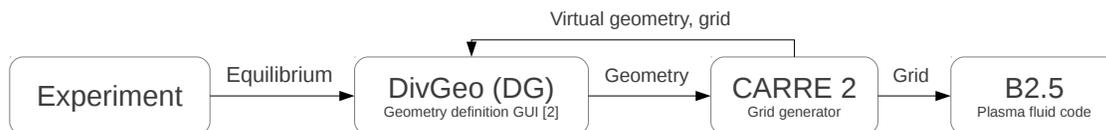


Fig. 5. The updated SOLPS 5 grid generation chain

required on the grid handling mechanics and the numerics of the B2.5 code.

Specifically, the grid import toolchain which prepares the geometry data was generalized to deal with the possibility of triangular cells resulting from the cell-cutting process. Also, due to the external cells being ignored by the PDE solver, the grid layout in computational space is more complex (fig. 7), requiring more care when setting up guard cells for the implementation of boundary conditions.

The resulting code version is referred to as *B2.5-extended*. Current work concentrates on code verification. First test runs were completed showing the correct implementation of the grid and boundary condition mechanics. In these runs, the standard B2.5 transport model (without drifts) is solved for a single ion species (D+), with Dirichlet-type boundary conditions on all boundaries. Robust convergence of the code was observed.

Figure 6 shows T_e solutions for two different grids: the run shown in fig. 6a uses the target-mode grid from fig. 3b, while the solution shown in fig. 6b was obtained on the vessel-mode grid shown in fig. 3c. Fig. 7 shows the grid layout

in computational space for the vessel-mode grid, using the standard mapping approach as described e.g. in [7].

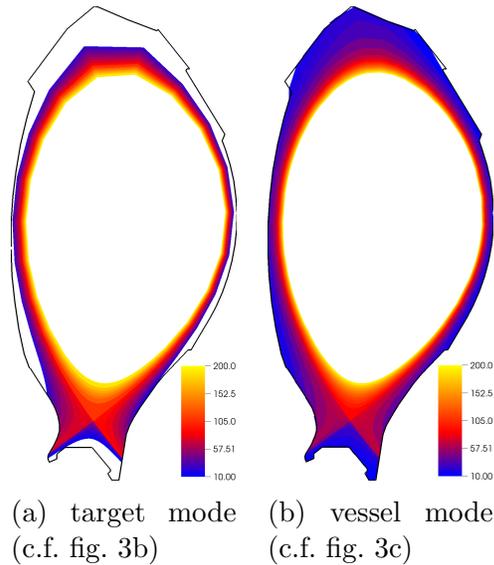


Fig. 6. B2.5-extended T_e solutions

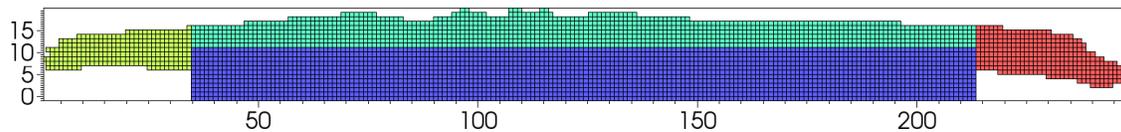


Fig. 7. Layout of the vessel-mode grid for figs. 3c and 6b in computational space. Because of external cells being ignored by the solver, the regions are no longer rectangular but ragged. Regions: blue=core, green=left divertor/PFR, red=right divertor/PFR, cyan=SOL

4 Conclusions and outlook

The SOLPS 5 scrape-off layer modeling package is being extended to provide consistent plasma backgrounds up to the first wall. To achieve this, the CARRE grid generator and the associated grid generation tool chain were updated to support automated generation of suitable grids. Furthermore the B2.5 plasma fluid solver was extended to make use of the resulting more complex grids, and first verification runs were performed.

The next steps are the implementation of a generalized sheath boundary condition as described in [5] and a detailed verification of the code with the built-in fluid neutrals model. Of special interest here is the comparison of standard and target-mode cut-cell grids to quantify the effect of grid non-orthogonality. Once this is done the coupling to the EIRENE neutrals has to be enabled for

the whole domain, eventually resulting in an updated code package that can be subjected to a validation effort.

References

- [1] R. Schneider, X. Bonnin, et al., *Contributions to Plasma Physics* 46 (2006) 3.
- [2] F. Subba, X. Bonnin, D. Coster, and R. Zanino, *Computer Physics Communications* 179 (2008) 194.
- [3] M. Aftosmis, M. Berger, and G. Adomavicius, *AIAA paper* 2000-808 (2000).
- [4] W. Dekeyser, M. Baelmans, D. Reiter, P. Börner, and V. Kotov, *Journal of Nuclear Materials* (2010) 584.
- [5] M. Baelmans, P. Börner, W. Dekeyser, and D. Reiter, *Nuclear Fusion* 51 (2011) 083023.
- [6] R. Marchand and M. Dumbery, *Computer Physics Communications* 96 (1996) 232.
- [7] H.-J. Klingshirn, *Adaptive grids and numerical fluid simulations for scrape-off layer plasmas*, PhD thesis, Technische Universität München, 2010.