

Presentation of the New SOLPS-ITER Code Package for Tokamak Plasma Edge Modelling^{*})

Xavier BONNIN, Wouter DEKEYSER, Richard PITTS, David COSTER¹⁾,
Serguey VOSKOBOYNIKOV²⁾ and Sven WIESEN³⁾

ITER Organization, Route de Vinon-sur-Verdon, CS 90 046, F-13067 St-Paul-lez-Durance, France

¹⁾*Max-Planck-Institut für Plasmaphysik, D-85748 Garching-bei-München, Germany*

²⁾*Peter the Great St. Petersburg Polytechnic University, 195251 Saint Petersburg, Russian Federation*

³⁾*Forschungszentrum Jülich GmbH, Institut für Energie- und Klimaforschung - Plasmaphysik, D-52425 Jülich, Germany*

(Received 31 March 2016 / Accepted 18 April 2016)

We present in this paper the code package SOLPS-ITER, initially introduced by S. Wiesen *et al.* [J. Nucl. Mater. **463**, 480 (2015)], dedicated to simulations of plasmas in the edge region of fusion devices. This package brings together previously existing SOLPS implementations and aims to become the new standard SOLPS version. We summarize the benchmarking work done to ensure backward compatibility with previous work, with a strong requirement on maintaining the viability and usability of the already extensive database of SOLPS runs used for the ITER divertor design and edge plasma physics studies worldwide over the years. The SOLPS-ITER package includes not only the plasma (B2.5) and neutral (Eirene) transport solvers, but also a large set of software tools for input file build-up, conversion of old runs, inline run analysis, and post-processing, all within a standardized portable run environment and version control system. Ongoing and planned upgrades to the code, such as extending the computational domain to the full vacuum vessel wall and a new graphical user interface, are also discussed.

© 2016 The Japan Society of Plasma Science and Nuclear Fusion Research

Keywords: SOLPS-ITER, B2.5-Eirene, plasma edge, plasma modelling, plasma code

DOI: 10.1585/pfr.11.1403102

1. Introduction

Simulations of ITER divertor performance, geometry optimization and estimates of key related parameters such as fuelling throughput have historically relied on the SOLPS (Scrape-Off Layer Plasma Simulation) plasma edge modelling tool, more specifically its versions 4.0, 4.2 and 4.3, developed mostly in collaboration with Forschungszentrum-Jülich [1] (FZJ). However, the majority of the SOLPS code base development has been performed by research groups outside the ITER Organization (IO), most notably at the Max-Planck Institut für Plasmaphysik (IPP) (both Garching and Greifswald branches), Université Paris 13 and St. Petersburg Polytechnical State University, leading ultimately to versions 5.0 [2], 5.1 [3] and 5.2 [4], respectively.

While the work at the IO and FZJ focused mainly on neutral transport physics [1, 5–9], the developments performed elsewhere have been concerned with extending and refining the physics capabilities of the plasma model. For example, the St. Petersburg group has improved the numerical solution of the drift terms and the electric potential equation [4, 10]. These enhancements have reached a level of maturity at which it makes sense to merge them

all. A decision to do this was taken at the IO in 2012 and work was commissioned under contract to couple the St. Petersburg version of the B2.5 plasma fluid solver with the newest version of the Eirene neutral transport code from FZJ [5] to yield a new code version, named SOLPS-ITER. The results of this activity were reported in [11]. Since then, further effort at the IO has been devoted to the inclusion of the refinements present in the 5.0 and 5.1 versions which were not all captured in the initial coupling exercise. The run environment for SOLPS-ITER has also been extended to the rest of the SOLPS code suite (mesh generation and I/O processing) and the code is now distributed by means of a dedicated repository maintained by the IO, including state-of-the-art version control and issue management. Whilst the first paper [11] introducing the new code package concentrated on a description of general convergence and physics benchmarking issues, the present article aims to describe in more detail the physical implementation of the code, including run environment and workflow, together with a look ahead to ongoing and future refinements to the package envisaged at the IO.

This paper is organized as follows. In Sec. 2, we present the general run environment under which the SOLPS-ITER code package is distributed, as well as the overall workflow used for producing SOLPS runs. In Sec. 3, we summarize briefly the additional efforts that

author's e-mail: xavier.bonin@iter.org

^{*}) This article is from 15th International Workshop on Plasma Edge Theory in Fusion Devices.

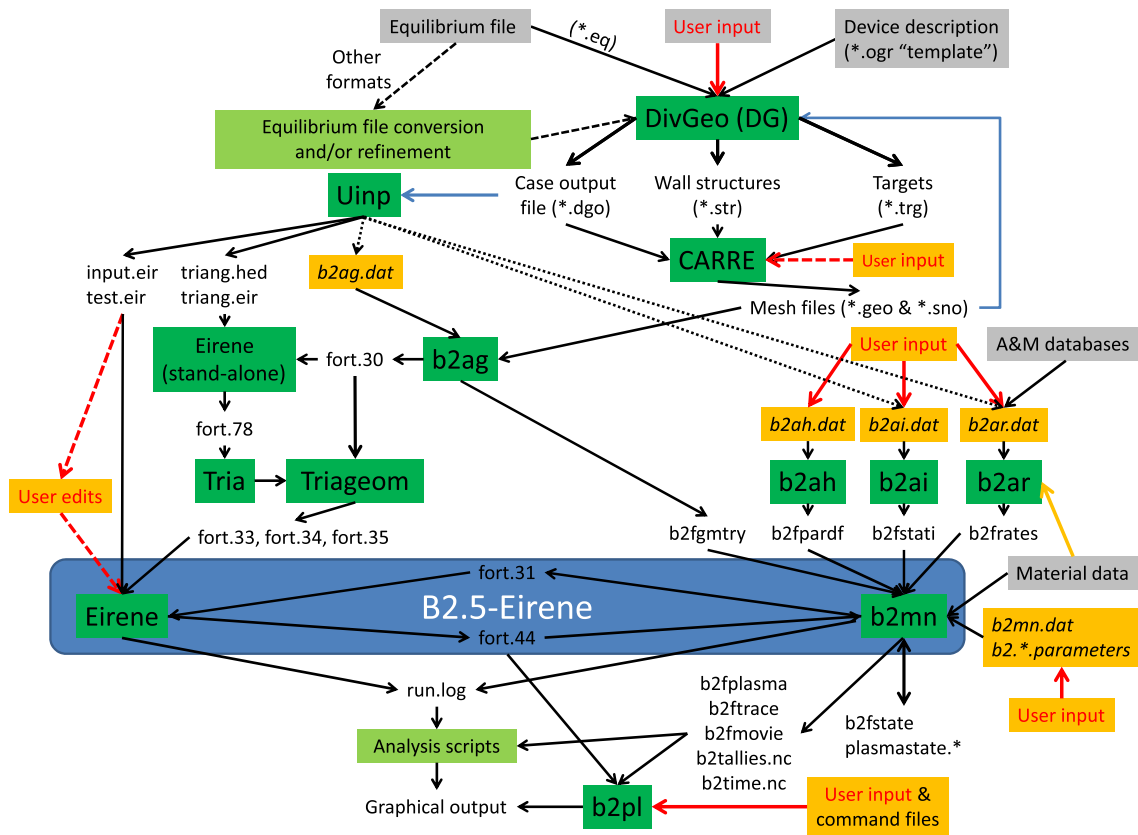


Fig. 1 The SOLPS-ITER workflow. The main code B2.5-Eirene is contained within the blue box. Each of the green boxes is a separate executable. Grey boxes represent data that is provided by external databases. Orange boxes represent steps the new user interface discussed in Sec. 4.2 is meant to facilitate.

have occurred since the work reported in [11] to ensure backward compatibility with previously existing SOLPS runs performed with the various previous code versions and other code improvements. In Sec. 4, we discuss the ongoing work and plans for new features being added to SOLPS-ITER. Finally, we give some concluding remarks in Sec. 5.

2. The SOLPS-ITER Run Environment and Workflow

Once the initial code write-up and benchmark activities described in [11] were completed, the plasma solver B2.5-Eirene needed to be embedded into a comprehensive run environment, similarly to those in place for previous SOLPS implementations. Indeed, a full SOLPS distribution must include additional tools for input file preparation, inline run analysis and post-processing, accompanied by the appropriate documentation.

A basic SOLPS-ITER run, being built from scratch, is obtained by following the workflow presented in Fig. 1 from top to bottom, where the main plasma solver B2.5-Eirene, which is the coupling of the two standalone executables “b2mn” and “Eirene”, is represented by the blue box. Everything above that box is input preparation (which

need of course not be repeated if one is restarting from an imported run, apart from some input parameter changes for physics analysis scans). Below that blue box, one ends with the current graphical post-processor “b2plot” (or “b2pl”).

The main steps required in building up a SOLPS-ITER run (or set of runs) are as follows:

- *Geometry definition:* the DivGeo utility is used to establish the location of the vacuum vessel walls, their properties, pumping locations, the topology of the equilibrium, plasma species present, etc. . .
- *Plasma mesh generation:* the Carre mesh generator [12] provides for a structured quadrangular mesh, to be used by the plasma solver, which covers an area going from the edge of the confined region to the line of first tangency with a limiting (non-divertor) wall structure.
- *Input file generation:* using the data provided above, the Uinp utility is used to create the corresponding Eirene (and some of the B2.5) input files.
- *Neutral mesh generation:* with the help of these input files, the Tria and Triageom programs create an unstructured triangular grid covering the area gridded by Carre plus the outer SOL domain up to the actual

vacuum vessel wall.

- *Physics input*: other plasma parameters are translated by means of the “b2ag” (geometry), “b2ah” (transport and boundary conditions), “b2ai” (initial state) and “b2ar” (atomic physics tables) into the internal format used by B2.5. These are supplemented by a list of run switches and additional namelists provided as stand-alone human-readable input files.

With the input files established, one can then launch runs of either Eirene or B2.5 in stand-alone mode, or coupled B2.5-Eirene runs and plot and analyze the results by means of the b2plot utility. An important feature of SOLPS-ITER is the existence of converter utilities which translate the input and output files from previous SOLPS versions into the format expected for SOLPS-ITER. In this way, continuation of older runs can proceed in a semi-automatic fashion with minimal user interaction.

The SOLPS-ITER distribution, as provided by a GIT repository maintained at the IO¹, includes all the source code and documentation associated with these various executables. Because this package is made available to a wide variety of users operating in different computing environments, the source code used must adhere to strict Fortran and C standards and the compilation files must allow for this diversity. This is achieved by determining, at session start-up, the current location and compiler choice and passing that information using environment variables at compilation time. Also at compile time, the code version number and GIT hash are embedded into the executable, and they are printed in the output files, to provide for traceability and reproducibility of the results.

Because the DivGeo, Carre, B2.5 and Eirene programs also exist as stand-alone units outside of the SOLPS-ITER package, they are included within it as self-contained GIT submodules, which can be cloned independently. These repositories are also linked to the SOLPS and Eirene repositories, hosted by IPP-Garching and FZJ, respectively, to ensure quick propagation of code changes from one version to another. For each of these submodules and for the SOLPS-ITER package as a whole, an issue reporting system is in place, allowing users to signal code bugs, point to missing information in the documentation or accompanying data files, or suggest new features. Similarly, contributors can submit suggestions of new source code, compiler flags, or documentation complements by means of a Web-based interface hosted as part of the repository. Major announcements concerning the code are disseminated by means of a SOLPS-ITER users mailing list².

Once all this architecture was in place, the SOLPS-ITER package was distributed to all interested parties among the ITER member research institutions. A launch workshop was held at the IO headquarters in Cadarache in April 2015 to present the new package, followed by addi-

tional training sessions and videoconferences held to discuss current issues with users around the world. The material from these meetings is stored on a dedicated restricted website.

The next step is the implementation of a library of “use cases” designed as reference starting points for runs on a variety of devices and geometries. This feature will make use of new Interface Data Structures (IDSs), which have been designed for plasma edge codes. These dedicated data structures, similar to the Consistent Physical Objects (CPOs) proposed by the earlier EU Integrated Tokamak Modelling Task Force (ITM TF) framework [13], will allow for standardized archival and retrieval of SOLPS-ITER run results, as well as the incorporation of SOLPS-ITER (or at least B2.5-Eirene) as an actor into wider integrated modelling physics workflows within the ITER Integrated Modelling Analysis Suite (IMAS) framework [14].

3. SOLPS-ITER Benchmarking

Because the benchmarking effort presented in [11] concentrated on obtaining, as far as possible, identical output for ITER runs using either SOLPS4.3 or SOLPS-ITER, it did not test many of the new features which are present in the SOLPS5.0/5.1 and 5.2 versions. As part of the deployment activities of the SOLPS-ITER package, these tests have now been performed. Indeed, several necessary code modifications were implemented for that purpose and sets of run switches have been identified which allow users to continue SOLPS5.0/5.1 or 5.2 existing runs with SOLPS-ITER while keeping an identical solution behaviour. Similarly, the set of switches necessary to keep a SOLPS4.3 solution unchanged and reproduce the published benchmark cases from [11] have been documented.

The benchmarks from [11] were based on the B2.5 code base from SOLPS5.2 (as of mid-2013) and a recent Eirene version. This Eirene version, however, differed somewhat from the one installed and used at ITER since 2008. Therefore, prior to the release of SOLPS-ITER, these code bases needed to be brought up-to-date, for those parts that had not yet been added during the benchmarking exercise. On the Eirene side, in addition to the continuing code improvements submitted by the development team at FZJ, this mostly concerned radiation transfer [6], treatment of chemical sputtering [1], double-null topology support [15], and new diagnostics implemented for the purpose of ITER divertor design work.

Moreover, because the SOLPS5.0 and 5.2 code bases had themselves also diverged in 2008, many recent improvements and features from 5.0 and 5.2 had to be reintroduced into SOLPS-ITER. These include, among many others, the treatment of additional external species provided by an outside code (see point 2 in Sec. 4.3), the generalization of Eq. (2) of [11] following [16], new feedback schemes [17], and graphical post-processing.

¹<https://git.iter.org/>, access restricted to registered users.

²solps-iter@iter.org, membership upon request.

4. Work and Plans towards Future SOLPS-ITER Improvements

This section reviews some of the features which we are planning to incorporate into SOLPS-ITER and for which some work has already begun.

4.1 Extended grids

A major shortcoming of current versions of the SOLPS code is the restriction of the plasma fluid simulation domain to points within the “first tangency line” intersecting a solid (non-divertor) surface. As a result, a number of physics issues important for a more complete model of the boundary plasma in today’s devices and ITER cannot be adequately captured by the code. A key example is the phenomenon of main chamber recycling, identified in recent years as a strong player affecting tokamak particle balance and the edge SOL plasma in current machines (see e.g. [18]). A related issue is the strong local plasma-surface interactions expected in ITER as a result of the high triangularity baseline magnetic equilibrium in which a secondary X-point is located close to the first wall in the upper regions of the main chamber [19]. The enhanced recycling driven by far SOL contact with the main walls also affects the local density profile, and hence the efficiency of ion cyclotron resonance heating (ICRH), one of the three main additional heating schemes foreseen on ITER.

To properly reproduce these and other relevant phenomena, it is thus necessary to extend the B2.5 simulation grid as much as possible to encompass the entire vacuum vessel volume outside the magnetic separatrix. Previous work in this direction has already been performed using the SOLPS4 [20, 21] and SOLPS5.0 [22] code bases. In the current version of SOLPS-ITER, discretization up to the real vacuum vessel wall is only performed for the calculation of neutral trajectories within Eirene, using an unstructured triangular grid. What is now needed is to apply the experience gained in the previous extensions [20–22] and transpose them to the SOLPS-ITER code base. This work will of course need to be validated via strict benchmarks, following the procedure established in [21]. For instance, when this work was done for SOLPS5.0, the proposed upward-pointing trapezoid test case from [21] was duplicated with a downward-pointing geometry, as well as a rectangular slab geometry. In all three cases, the problem posed was left-right symmetric and this symmetry was recovered in the solutions to a numerical accuracy of 10^{-13} . We also demonstrated that, for both trapezoid cases, the solution was independent of the labelling convention used for the slanted edges of the domain. Beyond these numerical checks, however, the sheath boundary conditions model must also be extended to be able to deal with cases where the field lines are at grazing incidence with the wall (see e.g. [23, 24]) and ways must be found to stabilize the solution for very short flux tubes in remote areas or near corners of the vacuum vessel.

4.2 New user interface design

The SOLPS package is notorious for being quite complicated to master for new users. Motivated by a desire to lower this barrier to entry and thus broaden the potential user base, the IO has devoted resources to produce a new graphical user interface with three principal aims:

- 1 Provide a “dashboard” utility to monitor ongoing runs, ascertain their status, continue stopped runs, and archive completed ones.
- 2 Provide a standard graphical tool to compare the results of various runs with each other and with experimental data, making use of the Interface Data Structures (IDS) infrastructure being implemented at ITER. The current available post-processing tool “b2plot” permits the results of only a single run to be inspected.
- 3 Provide a point-and-click utility to help in the production of the input files necessary to prepare a run, and check these files for sanity and consistency.

4.3 Additional physics

The SOLPS-ITER package remains of course a tool for state-of-the-art edge plasma physics research and, as such, will continue to be augmented with new physics features and analysis functions both by the IO development team and the worldwide SOLPS user community. At the moment, the following new features are either in development or being considered:

- 1 Within B2.5, a module exists to compute the surface temperature and its in-depth profile for target and wall elements as a function of the incident heat load and cooling scheme [25]. The results from this simple model are currently being calibrated against full multi-dimensional thermal calculations for the case of ITER divertor target monoblocks.
- 2 Heavy impurities such as tungsten are cumbersome to follow within the fluid framework of B2.5 [26]. However, with the use of tungsten as divertor and wall armour material increasingly being employed on present devices, as well as exclusively in the divertor for ITER [27], it is essential that SOLPS-ITER provide some means to include it. A promising approach being pursued is the coupling of B2.5-Eirene with kinetic trace impurity following codes such as IMPGYRO [28].
- 3 A very valuable validation exercise, reported in [29, 30], matching the code output to a detailed two-point model analysis of the SOL and divertor plasma, was performed for SOLSP4.3. Several groups have reported interest and are devoting manpower to reproducing this analysis with the SOLPS-ITER code base.
- 4 With the introduction of metallic divertors, there is a need for a plasma edge radiator to dissipate the SOL plasma heat load before it reaches the divertor targets. Currently, nitrogen and neon are the favoured

candidates. However, the case of nitrogen presents additional difficulties due to its associated molecular physics, both the break-up of the N_2 molecule as the gas is entering the plasma, and the possible formation of ammonia (NH_3) [31]. This latter issue is of particular concern for ITER since any tritiated ammonia exhausted from the plasma chamber will need to be separated and treated appropriately by the tritium plant. In current radiative divertor modelling efforts (see e.g. [32]), nitrogen is considered as a monatomic gas and its molecular chemistry is neglected. Thus, development of a consistent chemical model for nitrogen molecules will be necessary to improve the simulations of such regimes.

- 5 There is also interest in being able to use SOLPS-ITER in an interpretive mode, in which the solution would be constrained to fit a given experimental data set while computing the remaining quantities for a fully consistent plasma solution.

5. Conclusions

In this paper, we have presented the new SOLPS-ITER code package for plasma edge simulations in magnetically confined fusion devices. The core component of this package is the B2.5-Eirene coupling obtained from SOLPS5.2 and a modern Eirene version. After careful benchmarks against previous SOLPS4.3 results from the ITER divertor design database [11], this core code has been brought up-to-date with the most recent developments from the Eirene and B2.5 codes, augmented with a complete run environment, including conversion utilities from previous SOLPS versions, pre- and post-processing tools, and is being distributed through a central repository maintained at the IO. SOLPS-ITER can thus now be used to continue existing SOLPS5.0/5.1 and 5.2 runs without modification of the plasma solution, or with introduction of new physics effects imported from other SOLPS versions. The package is available to the wider plasma R&D community of the ITER members and continues to be improved following contributions from the IO development team and its user base. A demonstration of the capabilities of the new code package, in the context of a study of the Alcator C-Mod divertor, is presented in the companion paper [33].

6. Acknowledgement

The views and opinions expressed herein do not necessarily reflect those of the ITER Organization. Development of the SOLPS-ITER package has been a community effort and the authors wish to acknowledge here the many suggestions for code fixes and feature improvements provided by users from all ITER member parties since its initial distribution. A full list of these can be found by consulting the SOLPS-ITER repository.

©2016 ITER

- [1] A.S. Kukushkin, H.D. Pacher, V. Kotov, G.W. Pacher and D. Reiter, *Fusion Eng. Des.* **86**, 2865 (2011).
- [2] R. Schneider, X. Bonnin, K. Borrass, D.P. Coster, H. Kastelewicz *et al.*, *Contrib. Plasma Phys.* **46**, 3 (2006).
- [3] X. Bonnin, A.S. Kukushkin and D.P. Coster, *J. Nucl. Mater.* **390-391**, 274 (2009).
- [4] V. Rozhansky, E. Kaveeva, P. Molchanov, I. Veselova, S. Voskoboynikov *et al.*, *Nucl. Fusion* **49**, 025007 (2009).
- [5] V. Kotov, D. Reiter and A.S. Kukushkin, Numerical study of the ITER divertor plasma with the B2-Eirene code package, Juel-Report 4257 (2007).
- [6] V. Kotov, D. Reiter, A.S. Kukushkin, H.D. Pacher, P. Börner and S. Wiesen, *Contrib. Plasma Phys.* **46**, 635 (2006).
- [7] D. Reiter, V. Kotov, P. Börner, K. Sawada, R.K. Janev *et al.*, *J. Nucl. Mater.* **363-365**, 649 (2007).
- [8] V. Kotov, D. Reiter, A.S. Kukushkin and H.D. Pacher, *Phys. Scripta* **T138**, 014020 (2009).
- [9] A.S. Kukushkin, H.D. Pacher, V. Kotov, D. Reiter, D. Coster *et al.*, *Nucl. Fusion* **45**, 608 (2005).
- [10] V. Rozhansky, P. Molchanov, I. Veselova, S. Voskoboynikov, A. Kirk *et al.*, *Plasma Phys. Control. Fusion* **55**, 035005 (2013).
- [11] S. Wiesen, D. Reiter, V. Kotov, M. Baelmans, W. Dekeyser *et al.*, *J. Nucl. Mater.* **463**, 480 (2015).
- [12] R. Marchand and M. Dumbery, *Comput. Phys. Commun.* **96**, 232 (1996).
- [13] G.L. Falchetto, D. Coster, R. Coelho, B.D. Scott, L. Figini *et al.*, *Nucl. Fusion* **54**, 043018 (2014).
- [14] F. Imbeaux, S.D. Pinches, J.B. Lister, Y. Buravand, T. Casper *et al.*, *Nucl. Fusion* **55**, 123006 (2015).
- [15] M. Himabindu, A. Tyagi, D. Sharma, S.P. Deshpande and X. Bonnin, *Phys. Plasmas* **21**, 022504 (2014).
- [16] D.P. Coster, X. Bonnin, A.V. Chankin, H.-J. Klingshirn, C. Konz *et al.*, paper TH/P4-3, 22nd IAEA Fusion Energy Conference, Geneva, Switzerland (2008).
- [17] D.P. Coster, *J. Nucl. Mater.* **415**, S545 (2011).
- [18] B. LaBombard, J.W. Hughes, N. Smick, A. Graf, K. Marr *et al.*, *Phys. Plasmas* **15**, 056106 (2008).
- [19] R.A. Pitts, S. Carpentier, F. Escourbiac, T. Hirai, V. Komarov *et al.*, *J. Nucl. Mater.* **415**, S957 (2011).
- [20] W. Dekeyser, M. Baelmans, D. Reiter, P. Börner and V. Kotov, *J. Nucl. Mater.* **415**, S584 (2011).
- [21] M. Baelmans, P. Börner, W. Dekeyser and D. Reiter, *Nucl. Fusion* **51**, 083023 (2011).
- [22] H.-J. Klingshirn, D.P. Coster and X. Bonnin, *J. Nucl. Mater.* **438**, S856 (2013).
- [23] R.H. Cohen and D.D. Ryutov, *Contrib. Plasma Phys.* **44**, 111 (2004).
- [24] M. Shoucri, H. Gerhauser and F.H. Finken, *Phys. Plasmas* **16**, 103506 (2009).
- [25] D.P. Coster, X. Bonnin and M. Warrier, *Phys. Scr.* **T124**, 9 (2006).
- [26] X. Bonnin and D. Coster, *J. Nucl. Mater.* **415**, S488 (2011).
- [27] R.A. Pitts, S. Carpentier, F. Escourbiac, T. Hirai, V. Komarov *et al.*, *J. Nucl. Mater.* **438**, S48 (2013).
- [28] S. Yamoto, Y. Homma, K. Hoshino, Y. Sawada, X. Bonnin *et al.*, *J. Nucl. Mater.* **463**, 615 (2015).
- [29] V. Kotov, D. Reiter, D.P. Coster and A.S. Kukushkin, *Contrib. Plasma Phys.* **50**, 292 (2010).
- [30] V. Kotov and D. Reiter, *Plasma Phys. Control. Fusion* **51**, 115002 (2009).

- [31] V. Rohde, M. Oberkofler and the ASDEX Upgrade Team,
J. Nucl. Mater. **463**, 672 (2015).
[32] L. Aho-Mantila, X. Bonnin, D.P. Coster, C. Lowry, M.

- Wischmeier *et al.*, J. Nucl. Mater. **463**, 546 (2015).
[33] W. Dekeyser, X. Bonnin, S. Lisgo, R.A. Pitts, C. Boswell
et al., Plasma Fusion Res. **11**, 1403103 (2016).