# A NEW SIMULATION FRAMEWORK BASED ON THE KEPLER AND SCICOS OPEN-SOURCE SOFTWARE FOR THE DESIGN AND QUALIFICATION OF TOKAMAK CONTROL ALGORITHMS: FIRST TEST CASE RESULTS

Oliviero Barana[1], Cédric Boulbe[2], Sylvain Brémond[3], Simone Mannori[4], Philippe Moreau[5], Nathalie Ravenel[6]
and the EFDA ITM Task Force contributors

*[1,3,6]CEA,IRFM: Saint-Paul-lez-Durance, France, 13108, oliviero.barana@cea.fr*
*[5]CEA,IRFM: Saint-Paul-lez-Durance, France, 13108, philippe.jacques.moreau@cea.fr*
*[2]Université de Nice Sophia-Antipolis: Nice, France, 06108, cedric.boulbe@unice.fr*
*[4]INRIA Rocquencourt: Le Chesnay, France, 78153, simone.mannori@gmail.com*

*Plasma control is recognized to be a crucial issue for the achievement of ITER objectives. One of the most challenging tasks for the preparation of the ITER operation will therefore be the design and qualification of a variety of control algorithms. This highlights the need for a simulation platform capable of supporting the design, integration and test of advanced control algorithms on complex physics models. With this aim, a generic multi-purpose "flight" simulator (GMFS) is being developed at IRFM (Institut de Recherche sur la Fusion par confinement Magnétique), CEA Cadarache, France.*

*The GMFS is based on Kepler, a free interdisciplinary open-source Java software. Kepler will be used as a simulation platform to test and improve control algorithms before their actual use in the real control system. The physics and engineering codes complementary to the control algorithms will be supplied by the EFDA Integrated Tokamak Modelling Task Force (ITM-TF). The GMFS will be benchmarked, at the beginning, on the Tore Supra Tokamak.*

*In this paper we will report on a test case suitable to demonstrate the feasibility of a part of GMFS, namely the development of workflows where to create and verify ITER plasma boundary feedback control algorithms. It consists of: a) derivation of a linear plasma response model; b) design of a control diagram under the ScicosLab/Scicos open-source software; c) porting of the diagram under Kepler; d) substitution of the Kepler controller with a controller generated by a special Scicos extension; e) substitution of the simplified static linear model with the free-boundary equilibrium code CEDRES++.*

*The test case demonstrated the feasibility of employing Kepler, ScicosLab/Scicos and other expressly-made codes in view of the conception of valuable instruments for the active control of ITER and it can be considered as a first step in this direction.*

## I. INTRODUCTION

ITER[1,2] will be the most complex and powerful tokamak device ever built in the field of research on nuclear fusion by magnetic confinement, with a typical flat-top current of about 15 MA. Its main heating power will be supplied by α particles generated by fusion reactions but, in order to reach this condition, 73 MW of additional power from different sources will be needed.[3]

Even if the ITER design has attained a certain degree of maturity, many technical risks still remain, part of them having recently come to light thanks to physics results from existing machines.[4] Therefore, due to the large amount of power involved and to the consequent danger of damaging the machine, several operational constraints will have to be taken into account and a key role will be played by the active plasma control with which to reach the target performance.

The ITER plant control system (PCS)[5] will basically have three purposes: a) to ensure reliable ITER operation; b) to fully explore the ITER operational space for physics experiments using advanced control techniques that employ a number of actuators comparable to present-day machines to control an increased number of parameters; c) to guarantee machine protection in conjunction with the ITER central interlock system (CIS). The main PCS physics and operation requirements are: plasma equilibrium control; plasma kinetic control; advanced operation control (profile control); active control of magnetohydrodynamic (MHD) instabilities; disruption avoidance and mitigation; event handling. The PCS will interface to CIS and to the general CODAC (Communication, Data Acquisition and Control) infrastructure and its network, with an estimated number of signals equal to about 5000 (data rates of about 30 MByte/s). While the supervisory control system will transmit the reference waveforms and the initial set of algorithms and parameters to the PCS, when a discharge starts the PCS will have full control over all the necessary plant systems and orchestrate the pulse. Any necessary reconfiguration of plant systems during a discharge will be triggered by the PCS.

In order to satisfy all the requirements, careful design, qualification and optimization of control algorithms will be needed. Sophisticated algorithms, for example for controlling the MHD instabilities or plasma profiles like the current density one, will have to be implemented and carefully tested, the current experiments being an effective test bed where to check them.

It is in this context that plasma-discharge simulation frameworks acquire more and more importance. Reliable, comprehensive and fast simulation environments can help developing effective real-time algorithms, studying machine-protection issues, forecasting plasma behaviors, validating pre-programmed parameters and post-processing real data for improvements and tuning.

This paper deals with the new simulation framework which is being developed at IRFM (Institut de Recherche sur la Fusion par confinement Magnétique), CEA Cadarache, France, and is structured as follows: while chapter two describes the simulation framework, chapter three reports the results of the first test case developed within this framework; discussion and future developments are approached in the last chapter.

## II. THE GENERIC MULTI-PURPOSE "FLIGHT" SIMULATOR

### II.A. Rapid Survey of Simulators

Several "flight" simulators exist nowadays all around the world, but none of them seems completely to satisfy all the requirements the new IRFM simulator will have to fulfill. It is worth noting that all the simulators mentioned hereafter are based on Matlab/Simulink™; this product is very good and worldwide recognized, but it is very expensive and this fact could be a problem for associations whose budget is limited.

At DIII-D a very effective modeling/simulation framework[6] is integrated to the DIII-D PCS environment and it is used for development, validation and implementation of algorithms. It can perform "hardware-in-the-loop" (HIL) simulations and it can be used also for other tokamaks like KSTAR and EAST, but for this it needs a software version of the PCS.

DINA-CH[7] is the tokamak full-discharge simulator developed at TCV. It has reached outstanding results in terms of modeling, but it seems nevertheless limited in what concerns the interface to the real PCS.

XSC (eXtreme Shape Controller) Tools[8] are a set of graphic tools created for the design and validation of XSC, the new JET plasma shape controller. This set of tools, which appears to be very interesting and efficient, it is currently not yet machine-independent, even if an effort to extend it also to other tokamaks is *en route*. Even if it is capable of providing the code ready to be deployed in a controller, it does not seem to supply the possibility to test the system in HIL mode.

A good example of development of HIL simulations is the project concerning the rapid prototyping of the ITER central safety system[9]. This work is anyway based on a mock-up of, and not on, the whole control system.

Finally, a promising framework for ITER shape controller and transport simulations can be found in [10]. Based on CORSICA,[11] it is nevertheless not comprising the HIL mode.

### II.B. General Description

The IRFM simulation framework is known as generic multi-purpose (plasma discharge) flight simulator[12] (GMFS). It will be based on Kepler,[13] a free interdisciplinary open-source Java software which saw the light in the aim to help scientists develop models, improve their understanding and share results. In the proposed framework Kepler will be used as a simulation platform to test and improve control algorithms before their actual use in the real control system. The physics and engineering codes complementary to the control algorithms and supplied to the simulation platform will come from the results of the effort launched within the EFDA[14] Integrated Tokamak Modelling Task Force[15] (ITM-TF). ITM-TF was set up in 2004 with the long-term purpose of providing European laboratories with a suite of codes necessary to prepare and analyze future ITER discharges. GMFS can be considered generic, differently from other existing simulation tools which are basically machine-dependent, because it will use a set of ITM-TF standardized tools benchmarked, if possible, on any of the current tokamak experiments. GMFS can be considered multi-purpose because it will not only allow discharge simulations in a full-simulation mode, but it will also be used to check the control algorithms in the HIL mode, where the real control system takes part in the tests as well. GMFS will be benchmarked at least on Tore Supra, the IRFM tokamak.

### II.C. Basic Requirements

The GMFS basic requirements are the following:

- Possibility to pre-execute discharges in simulation mode in order to check control parameters and algorithms;
- Availability of an interface between GMFS and the control system, in order to be able to execute a plasma discharge with parameters already used in a simulation;
- Availability of a comprehensive library of physics codes, easily updateable, allowing the quick implementation of new control schemes;

- Availability of feedback control algorithms based on complex physics models, in order to have advanced control methods permitting the execution of high-performance plasmas in steady-state regimes;
- Accessibility to engineering models, strictly connected to the machine protection system, through which to implement reliable interlock schemes which should allow the execution of safe machine operations.

## II.D. Architecture

The block diagram of a pulse control chain is shown in Fig. 1. A certain degree of similarity can be noticed between the white blocks, which concern a real experiment, and the grey blocks, related to GMFS. While the Integrated Simulation Editor (ISE) is the GMFS equivalent of a Pulse Scheduler, the Models block comprehends all the engineering and physics codes that replicate, as far as possible, the real behavior of a tokamak and that are available in Kepler. The Control Toolbox block is constituted by all the tools available for the development of the control algorithms and it can be used both for the actual control system and in GMFS. A simulated pulse involves the switches S1, S2, S4, S6 and S7, all with their selector in the A position. In the very useful HIL mode, which permits to test the hardware without real discharges, the simulated controller is substituted by the real controller and the diagnostic equipment is linked to the Models block thanks to the Synthetic Diagnostics block. In this mode the switch/selector chain is the following: S1/(A or B) – S2/B – S3/A – S4/B – S6/B – S5/A – S7/B. Other more or less different configurations are of course possible.

## II.E. Development Environment And Data Access

People working in the ITM-TF project share a common development environment, which is generally know as “Gateway”.[16] This facility, located in Portici, near Naples, Italy, has a computing performance of about 1 TFlop/s, with a shared storage data area of about 100 TBytes; sixteen nodes are reserved for High-Performance Computing (HPC) resources, while three nodes (equipped with Scientific Linux and 16 GBytes of RAM each) serve as front-end to allow users remotely accessing the Gateway resources. Kepler as well as ScicosLab/Scicos[17] (see later), the physics codes and the whole ITM-TF data infrastructure, reside on the Gateway.

In order to share information among the different codes in a consistent way, a new data structure has been formulated inside ITM-TF.[18] XML schemas are used to describe an entire fusion device with a tree structure. Each sub-tree corresponds to a so-called Consistent Physical Object (CPO), addressed as a whole entity through UAL (Universal Access Layer), the ITM-TF data access layer.[19] The current UAL implementation is based on MDSplus.[20]

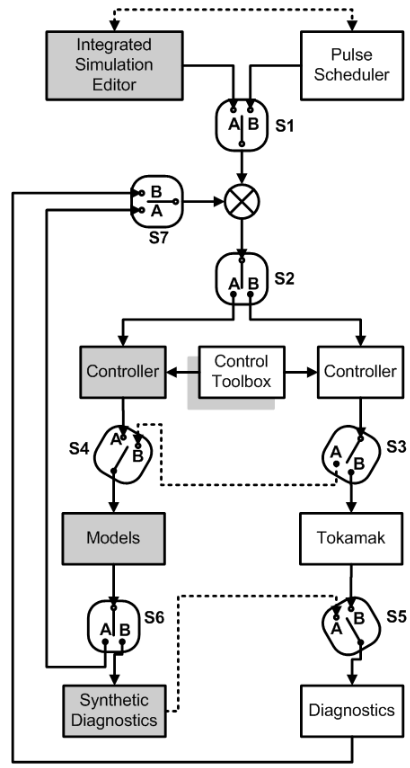


Fig. 1. Block diagram of a pulse control chain. The white blocks concern a real experiment, the grey ones the simulation. Blocks linking the two sides are shadowed. The switches allow the selection of different configurations.

## III. THE TEST CASE: ITER PLASMA SHAPE CONTROL

### III.A. Introduction

The test case has been developed in the framework of the study of ITER scenarios where plasma equilibrium feedback control is mandatory (e.g. scenario design with constraints on poloidal-field (PF) coils). The test case aims at having a working example, in fully simulated mode, of a feedback control of the ITER plasma shape. The control scheme envisages at controlling the distance between the plasma last closed flux surface and eight fixed points on the first wall (eight so-called gaps) using a classical multi-input multi-output proportional-integral-derivative (PID) controller, twelve PF coils as actuators and a tokamak model issued from the free-boundary equilibrium code CEDRES++.[21] The test-case reference equilibrium is represented in Fig. 2 together with the eight gaps; the reference plasma current is $I_P$ = 15 MA.

Actually, the gap#1 is fixed because of the limiter and therefore the gaps to be controlled are seven.

The controller has been designed using a version of the ScicosLab/Scicos package expressly adapted to the ITM-TF environment. Scicos, which is being used as Control Toolbox (see Fig. 1) on Tore Supra, has been chosen since it is open-source and, like Matlab/Simulink, can automatically generate C code. In order to design a controller under Scicos, a linear model is needed in place of CEDRES++.

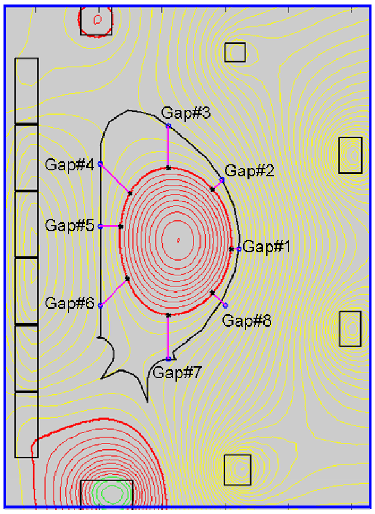


Fig. 2. The test-case reference equilibrium together with the position of the eight gaps.

### III.B. Linear Model Identification

Starting from the currents in the PF coils, CEDRES++ has been used to derive a linear model, for the gap calculation around the reference equilibrium, made of a static and of a dynamic part. The static part is given by:

$$\delta gaps = M_{GAP} \cdot \delta I_{PF}\,, \qquad (1)$$

where $M_{GAP}$ is a 7x12 matrix and $I_{PF}$ the current in the PF coils. The 12x12 mutual inductance matrix $M_{FLUX}$ of the system, including the plasma contribution, has been calculated by means of:

$$\delta Flux = M_{FLUX} \cdot \delta I_{PF}\,, \qquad (2)$$

where δFlux is the flux variation in the PF coils. It is then possible to derive the dynamic part of the model, which is given by the circuit equations

$$\delta V_{PF} = R \cdot \delta I_{PF} + d/dt(\delta Flux), \qquad (3)$$

where R is a diagonal matrix of resistances. Using (2) in (3), $I_{PF}$ can be expressed as a function of $V_{PF}$, the PF-coil voltage:

$$\frac{d}{dt}(\delta I_{PF}) = -M_{FLUX}^{-1} \cdot R \cdot \delta I_{PF} + M_{FLUX}^{-1} \cdot \delta V_{PF} \ (4).$$

### III.C. Simulation Results Under ScicosLab/Scicos

The block diagram of the ITER gap feedback control workflow under Scicos is shown in Fig. 3. The circuit equations have been implemented in the simulation using a linear state-space system (LSSS) block. The targets and the computed values are actually variations with respect to the starting point and not absolute values.

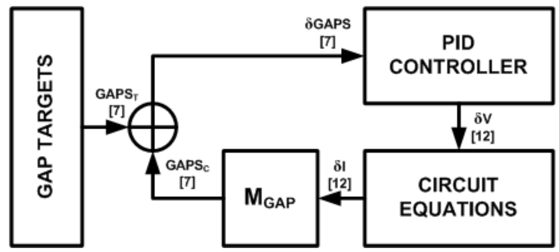


Fig. 3. Block diagram of the ITER gap feedback control workflow under Scicos. T stands for Target and C for Current.

The results of the simulation are displayed in Fig. 4 (sampling time of 1 ms). Once the results have been considered satisfying, the C code related to the controller was generated.

### III.D. Simulation Under Kepler

Normal blocks in Kepler are called “actors”. Each workflow has typically a higher level supervisor, called “director”, whose purpose is to manage its workflow. Actors can be composite, i.e. each actor can contain other actors and directors in a nested way. Since Kepler is based on Java and the code generated by Scicos is in C, a specific tool, called *fc2k*,[22] has been developed inside ITM-TF to wrap the C code with Java and automatically generate a working Kepler actor. The ITER plasma shape control test case was developed in three steps.

*III.D.1. Step 1: exact replica of the Scicos workflow*

The first step aimed at exactly reproducing the Scicos results using only actors available in Kepler, in order to get acquainted with the new platform and to validate it. In particular, custom actors replicating the same linear model as Scicos had to be developed. The final workflow could perfectly reproduce the Scicos simulation.

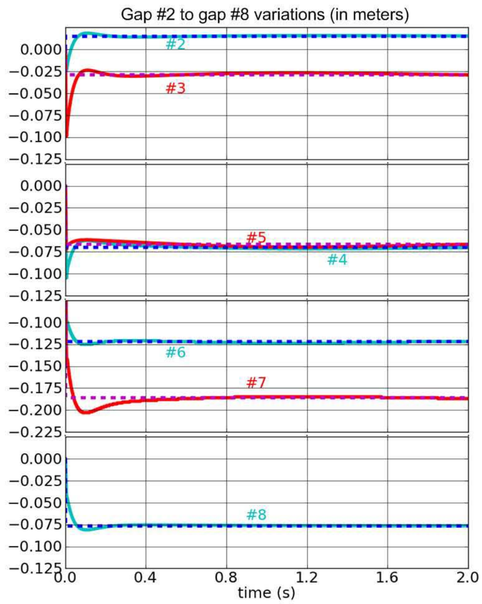


Fig. 4. Simulation of the gap active control under Scicos. The dashed lines are the targets.

*III.D.2. Step 2: introduction of the Scicos controller*

Step 2 consisted in introducing the PID-controller actor generated by *fc2k* inside the workflow, without substituting the static linear model of Eq. (1) with the CEDRES++ model. This workflow, which is shown in Fig. 5, is the standard ITM-TF one and consists basically of four composite actors, dealing respectively with an initialization phase, a main sequence, a time update and an exit phase. The main-sequence actor embodies the most important blocks (PID controller, system dynamic-model block, etc.). Fig. 6 shows the comparison between the Scicos and Kepler results on the gap active control; the agreement is very good, the differences being basically caused by a different way of implementing the LSSS (i.e. the dynamic part of the model) block.

*III.D.3. Step 3: introduction of the CEDRES++ actor*

The CEDRES++ code has recently been added to the library of available physics codes under the ITM-TF environment and a version of CEDRES++ suitable to be used by *fc2k* has been supplied too. The workflow with Kepler is the same as Fig. 5. In the initialization phase data are retrieved from the UAL database using the ITM-TF actor *ualinit*. In the exit phase data are stored in the UAL database using the ITM-TF actor *ualcollector*. Ad-hoc actors have been produced under ITM-TF in order to extract single signals from the CPOs (*ualdemux*) or to update CPOs with single signals (*ualmux*). This feature has been used to interface the controller actor issued from Scicos with the remaining part of the Kepler workflow. The controller actor does not use CPOs, since it must compatible with real plasma control systems; on the contrary, the CEDRES++ actor requires the compatibility with the ITM-TF data structure, i.e. it must exchange data using the CPO format.

The CEDRES++ actor was not sufficient to substitute the static linear model actor with a more effective physics model. It has been also necessary to integrate it with a specific C++ code with which to compute the gaps using the CEDRES++ output. The results of the complete simulation of the ITER plasma shape control are represented in Fig. 7. The agreement with Figg. 4 and 6 is good. The larger errors are basically due to the use of a physics model more complex than the Scicos one and to uncertainty in the calculation of the gaps caused by the mesh granularity used in CEDRES++.

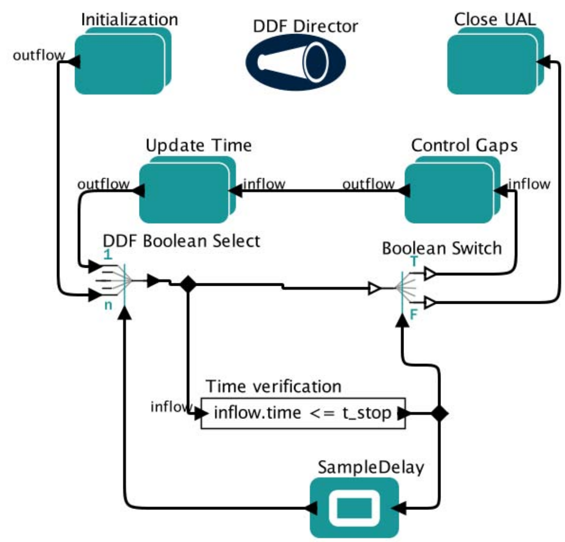


Fig. 5. The standard ITM-TF workflow.

## IV. DISCUSSION AND FUTURE DEVELOPMENTS

The test case described in this paper demonstrates basically two things: a) the first effective use of GMFS as platform to simulate feedback control scenario in view of ITER; b) a successful use of the ITM-TF resources. Yet there are a few things that must be improved. In particular: a) CEDRES++ is not adapted to be executed hundreds or thousands of times in a control loop, since under the Gateway a CEDRES++ iteration takes about 10 s; b) the CEDRES++ memory management was not optimized; an undetected memory leak of 19 MBytes was quickly running the Gateway resources out; the complete simulation was only possible after a partial reduction of the leak down to 1.2 MBytes. These two issues actually will concern all the ITM-TF codes, which will need an optimized programming and a strict benchmarking before their effective use for the study of feedback control scenarios.

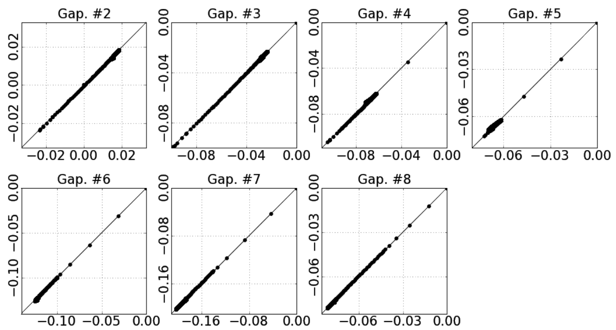


Fig. 6. Comparison between Kepler (y axes) and Scicos (x axes) gap active control results. The Kepler physics model is the linear one.

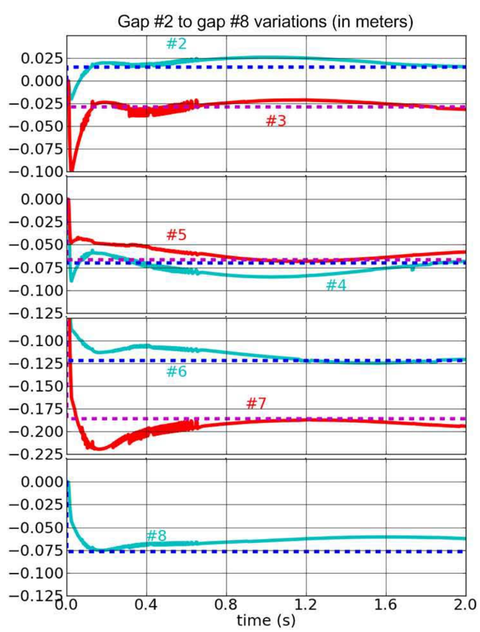


Fig. 7. Results of the complete simulation of the ITER plasma shape control with CEDRES++.

A new Kepler workflow structure, especially adapted to control issues, is under development. It will exploit the Finite State Machine (FSM) concepts. FSMs look appealing for feedback control design and event handling due to their versatility and easy design. Future developments also envisage simulating gap control using an advanced[23] version of CEDRES++ coupled with a transport code (e.g. Cronos[24]).

## REFERENCES


1. R. AYMAR et al., *Plasma Phys. Control. Fusion* **44**, 519 (2002).
2. PROGRESS IN THE ITER PHYSICS BASIS, *Nucl. Fusion* special issue **47**, S1 (2007).
3. J. JACQUINOT et al., *Fusion Eng. Design* **84**, 125 (2009).
4. N. HOLTKAMP, *Fusion Eng. Design* **84**, 98 (2009).
5. A. WINTER et al., *Proc. of the 37th EPS Conference on Plasma Physics*, Dublin (Ireland), 21-25 June 2010.
6. J.A. LEUER et al., *Fus. Eng. Design.* **74**, 645 (2005).
7. J. B. LISTER et al., *Fusion Eng. Design* **74**, 633 (2005).
8. G. DE TOMMASI et al., *IEEE Trans. Plasma Sci.* **35**, 709 (2007).
9. G. AMBROSINO et al., *IEEE Trans. Plasma Sci.* **38**, 1662 (2010).
10. T. A. CASPER et al., *Fusion Eng. Design* **83**, 552 (2008).
11. CORSICA. Online at: https://fusion.gat.com/THEORY/caltrans/.
12. P. MOREAU et al., *Proc. of the 26th Symposium on Fusion Technology*, Porto (Portugal), 27 September – 01 October 2010.
13. Kepler. Online at: https://kepler-project.org/.
14. EFDA. Online at: http://www.efda.org/.
15. A. BECOULET et al., *Comp. Phys. Comm.* **177**, 55 (2007).
16. The EFDA ITM Gateway. Online at: http://www.efda-itm.eu/.
17. The ScicosLab/Scicos project. Online at: http://www.scicoslab.org/.
18. B. GUILLERMINET et al., *Fusion Eng. Design* **83**, 442 (2008).
19. G. MANDUCHI et al., *Fusion Eng. Design* **83**, 462 (2008).
20. MDSplus. Online at: http://www.mdsplus.org/.
21. J. Blum and J. Le Foll., *Comput. Phys. Rep.* **1**, 465 (1984).
22. N. RAVENEL et al., *Fusion Eng. Design* **85**, 482 (2010).
23. P. HERTOUT et al., *Proc. of the 26th Symposium on Fusion Technology*, Porto (Portugal), 27 September – 01 October 2010.
24. V. BASIUK et al., *Nucl. Fusion* **43**, 822 (2003).