

FUSION PLASMA SIMULATION IN THE INTERACTIVE GRID

Francisco CASTEJÓN

*Laboratorio Nacional de Fusión—Asociación Euratom/Ciemat
28040 Madrid, Spain*

✉

*BIFI: Instituto de Biocomputación y Física de Sistemas Complejos
Universidad de Zaragoza
50009 Zaragoza, Spain*

José Miguel REYNOLDS, Fermín SERRANO, Ruben VALLES

*BIFI: Instituto de Biocomputación y Física de Sistemas Complejos
Universidad de Zaragoza
50009 Zaragoza, Spain*

Alfonso TARANCÓN, José Luis VELASCO

*BIFI: Instituto de Biocomputación y Física de Sistemas Complejos
Universidad de Zaragoza
50009 Zaragoza, Spain*

✉

*Departamento de Física Teórica
Universidad de Zaragoza
50009 Zaragoza, Spain
e-mail: rvalles@unizar.es*

Abstract. Within the framework of the int.eu.grid project, IVISDEP, which stands for Interactive Visualizer of an Integrator of Stochastic Differential Equations for Plasmas, has been successfully ported and executed. It is an example of integration of many of the innovative features that this grid infrastructure offers. The applica-

tion itself simulates and visualizes the evolution of the plasma inside a stellarator fusion device. The followed approach implies the simulation of a great number of trajectories of particles, whose positions are calculated among a large number of nodes in the Grid. The graphical interface allows the user to see and modify the physical and computational parameters of the simulation interactively, making this application a valuable tool for researchers in plasma physics.

Keywords: Langevin, stellarator, mpi, interactivity, visualization, opengl

1 INTRODUCTION

The int.eu.grid (Interactive European Grid) project started on 1st May, 2006. The objective of the project is the deployment of an advanced Grid-empowered infrastructure oriented to support the execution of demanding interactive applications.

The described model faces the challenges which rise when Grid-enabling an interactive application and unveils the int.eu.grid e-Infrastructure as a powerful framework for the deployment of demanding interactive applications.

Several innovative features mainly regarding grid middleware were to be developed in the int.eu.grid project. Due to the computer, graphical and interactivity requirements, IVISDEP fits as a good example of integration of all of them.

2 ISDEP: THE APPLICATION COMPUTING CORE

2.1 Scientific Interest

Fusion offers the prospect of a long-term safe and environment-friendly energy source. It is particularly attractive because it uses abundant and deslocalized fuel: water and lithium. The basic idea is to obtain energy from the union (fusion) of light atom nuclei. The mass of the components after the fusion is less than before and the exceeding mass m is transformed into kinetic energy following the famous expression $E = mc^2$ where c is the speed of light. In order to overcome the electrostatic repulsions between atoms, high energies are needed. This makes necessary to enormously increase the temperature of the fuel, leading it to a plasma state. This brings the problem of maintaining such a hot plasma without melting the container. The most promising way is to generate a “magnetic bottle” by means of a set of coils and to contain the fuel without touching it. For more details about the physics of magnetically confined plasmas, see [1].

The device which is studied by ISDEP, named TJ-II, is a medium size stellarator that operates at CIEMAT in Madrid since 1997 [2]. Some of its parameters are: average major radius of 1.5 m, maximum minor radius of 0.2 m, an average magnetic field of 1 T and pulse length from 0.2 s to 0.5 s. The magnetic configuration is produced mainly by two sets of coils: the hard core (HC) and the toroidal field

coils (TF). The hard core (HC) consists on a circular poloidal field coil and a helical coil closely wound around the former. The TF set consists of 32 TF plane circular coils whose centres are located on a helix that winds around the HC [3]. The intricacy of the coil configuration produces a complex behaviour extremely rich and interesting for research.

Active research is being carried out in order to study best configuration of coils, how to inject energy, and other physical and engineering parameters of the system.

2.2 The ISDEP code

One way to get an approximation to the problem is computer simulation of the behaviour of the reactor. The Fusion Plasma Application is based on ISDEP, a C-language-programmed computing core, highly optimized for the solving of the ion collisional transport in TJ-II.

Estimating the collisional transport in TJ-II means to solve the integro-differential equation for the distribution function f of the ions in the plasma.

$$\frac{\partial f}{\partial t} + \vec{\nabla}_{\vec{r}_{\text{GC}}} \cdot (\vec{u}f) + \frac{\partial}{\partial \lambda}(a_{\lambda}f) + \frac{\partial}{\partial x^2}(a_{x^2}f) = \mathcal{L}f \quad (1)$$

Here \vec{r}_{GC} denotes the position of the ions and λ and x^2 their velocity. Under several non-restrictive assumptions (guiding-centre approximation, long-range Coloumb collisions, see [4] for more details) this equation describes the evolution of the plasma inside the device: the magnetic and electric field and the collective effects of collisions between particles are included in the equation via the functions a_{λ}, a_{x^2} and the second-order differential operator \mathcal{L} . Its detailed functional form is given in [4].

This problem is non-linear, and of high complexity, since \mathcal{L} depends on f . The *Langevin approach* followed by ISDEP is to linearize the equation, ignoring this latter dependence. The evolution of a subset of the plasma (*test ions*) inside a plasma considered as a stationary thermal bath is then studied.

Now, Equation 1 is equivalent to solve a closed set of five coupled stochastic differential equations (SDE):

$$\frac{d\vec{r}_{\text{GC}}}{dt} = \vec{a}_{r_{\text{GC}}}, \quad (2)$$

$$\frac{d\lambda}{dt} = [a_{\lambda} + a_{\lambda}^{\text{Ito}}] + b_{\lambda}\xi_{\lambda}, \quad (3)$$

$$\frac{dx^2}{dt} = [a_{x^2} + a_{x^2}^{\text{Ito}}] + b_{x^2}\xi_{x^2}. \quad (4)$$

where ξ_{λ} and ξ_{x^2} are independent *white noises*. The terms $a_{x^2}^{\text{Ito}}$ and a_{λ}^{Ito} depend on \mathcal{L} .

The problem under study becomes then more intuitive: solving this SDE means to calculate a large number of trajectories of the *test ions* in the presence of magnetic and electric fields (described by a_{λ}, a_{x^2}) and collisions with a background plasma of a given temperature and density ($a_{x^2}^{\text{Ito}}, a_{\lambda}^{\text{Ito}}, b_{x^2}$ and b_{λ}). This latter term is aleatory

(note the white noises in Equation 4), and requires the implementation of a pseudo-random number generator.

In order to calculate the ion trajectories, ISDEP code uses a numerical algorithm (developed by Kloeden and Pearson [7]) of the Runge-Kutta type. The basis of this numerical scheme is as follows: given the position and velocity at time t_0 , the forces acting over the ion are computed, and one can obtain the position and velocity of the ion in a later time $t_0 + dt$. Iterating this numerical scheme about 10^6 times, one obtains the (discrete) trajectory. When a great number of these trajectories (typically of the order of 10^5 to 10^6) have been calculated, a careful statistical treatment allows to calculate relevant magnitudes of the plasma, such as temperatures, pressures and fluxes of heat and particles.

The 3-dimensional magnetic configuration necessary for calculating the trajectories is read from a grid, and interpolated in simulation time. The remaining parameters, namely the electric field, the density and the ion temperature, are also read from input files. They can be independently modified or even “switched on and off”. In that way, one can estimate the separate influence of each of them on the behaviour of the plasma, or study the evolution of the plasma under changes of the background conditions.

From the physical point of view, the main point of the Langevin approach is that it starts from basic physics without many doubtful assumptions. Moreover, it allows a three-dimensional study of the plasma, which is further powered by the visualization tool. From the computational point of view, it is perfectly suited for grid computing, since the trajectories are independent, and can be calculated each in one different node. The code ISDEP has already been successfully used in several studies: describing the ion collisional transport in TJ-II [4], estimating the plasma heating under special confinement conditions [8] and search for a flux-expansion divertor for TJ-II [9].

3 ANTECEDENTS

3.1 IVISDEP: Graphical Application

Not only graphical, but three-dimensional information, such as a map of the collisions of the particles with the vacuum chamber of the fusion device, is sometimes crucial in this kind of studies.

IVISDEP started to be developed as an added value software tool for the analysis phase of the data simulated in plasmas. At first, it was a 3D opengl visualizer, whose main function was to represent batch generated data in a 3D model; this way the numerical analysis could be contrasted with it allowing to understand different behaviours in plasmas. In a next step, ISDEP computing core was modified and a kind of API was produced to let it communicate with the graphical interface interactively. In that way the simulation could be steered by changing its parameters without having to restart it from the beginning.

This Graphical Interface is in charge of calling the ISDEP computing core and receiving the calculated data in each time step to make the final visualization of each particle.

This graphical application is programmed in the object-oriented language C++, but also includes C bindings since the whole computing core uses this programming language.

It uses Fox Toolkit library as GUI development framework and makes use of GLUT and Mesa (a free software/open source library, fully software-based graphics API that is code-compatible with OpenGL) to represent the data graphically.

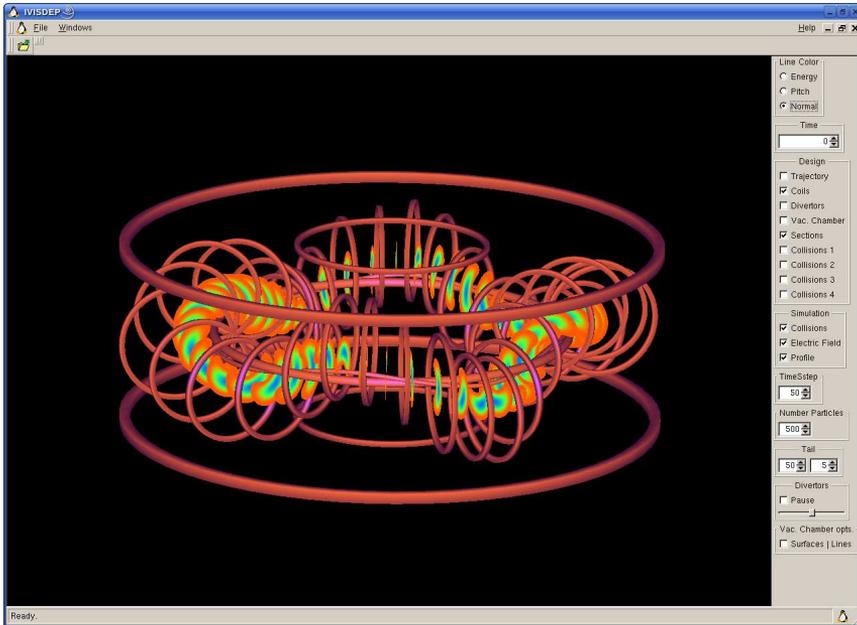


Fig. 1. IVISDEP view with plasma inside a stellarator fusion device

The application was from the beginning developed with the aim to be simple enough for new users to start working with it, at the same time as it would include interesting features to be useful. Several initial parameters can be selected easily from the GUI. Moreover, some of them, interactive and visual, can be changed. The new results are obtained in just the next iteration of the simulation.

3.2 ISDEP Executed in EGEE

The computing core has already been adapted to the grid. It has been running over EGEE since 2006 where it has generated important results. Some of them were already presented in the Second EGEE User Forum [10]. The great computing power provided by EGEE has been crucial for these studies, as acknowledged in [4, 8, 9].

Nevertheless, there was the need to join together the computing power of the grid and the visualization parts of the program. Due to the importance of being able to see how the trajectories behave during the simulation, to change related parameters, to steer it and to obtain the results graphically, the computing core and the visualization application were adapted to the interactive environment of int.eu.grid. The aim is to offer a useful and attractive framework for the fusion researchers and to reflect the potential power of interactive grid computing.

4 ADAPTATION OF IVISDEP TO INT.EU.GRID FRAMEWORK

4.1 Use Case

During the first months of the project, the definition of the grid infrastructure and the requirements for the interactive middleware started to be clarified. In addition, the fusion application was analyzed as a whole by experts in each field to agree the use case.

The user should be able to launch and to interact with the application, viewing the results of this interaction in a reasonable time, running the application in the Worker Nodes (WNS) of the Grid.

In this new interactive grid scenario, the main objective is to distribute the whole number of trajectories to be calculated among as much WNs as possible. This way, we can compute a greater number of trajectories at once, and obtain more detailed results. This approach is possible due to the assumption of no interaction among the different trajectories. This issue is already taken into account in the resolution of the SDE thanks to the definition of background ion distribution, as described in Section 2.2.

4.2 Architecture

The main scheme of the fusion application to be ported to int.eu.grid had a big change of structure. The main pieces of the integration of the application are the following: MPI [11], RAS server, Migrating Desktop (MD) [12], Glogin [13] and Gvid [14]. As can be seen, such a complex use case has involved the integration of several specific tools.

In the int.eu.grid scenario, the application is started from MD, which is a roaming virtual Java desktop to facilitate the access to the grid. It is the entering point to the grid and at the same time is the layer where the final visualization is achieved. Gvid is the plug-in which makes possible the visualization inside the Java window and Glogin is the channel through which the whole communication between the RAS server and the grid directly takes place.

Currently, the application needs to be steered remotely from the Gvid plug-in in the MD. Therefore, the graphical interface part has been modified, dropping FoxToolkit to become a GLUT compliant application, in order to be able to re-

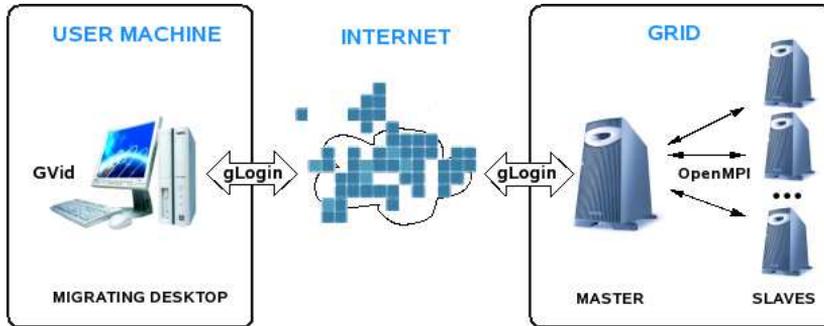


Fig. 2. System architecture overview

ceive callbacks and changes of parameters during the simulation, using the remote callbacks implemented inside a Remote Glut library.

The three main components of Gvid make possible the remote visualization of the application. It has to be compiled to use these Gvid and Remote Glut libraries in order to perform an offscreen rendering, that is encoded as an mpeg video and transmitted over the network to the visual java plugin, where the output adapters are able to decode the video and capture interaction events from the user.

The ISDEP computing core was also restructured to distribute the computation among the WNs. This new distributed system was based on MPI because of the fact that it provided the features needed to manage the data calculated in the different machines.

The new scheme of the fusion application fits a Master—Slaves one, in which the Master is in charge of the visualization issues and gathering, in a synchronized way, the data calculated by the Slaves. The latter are fully dedicated to the calculation of new positions of the particles and to sending the data to the Master to be visualized. The ISDEP algorithm was improved to distribute the computation among the Slaves by calculating in each step the next position for all particles, while in the ancient schema the trajectories were calculated sequentially.

Furthermore, we can distinguish two kinds of interactive parameters:

- The graphical parameters: These are the ones that affect only the 3D representation of the scene. They can be rotated, zoomed, and allow to visualize different 3D models related to the physical structure of the simulated fusion device. They are captured by Gvid adapters and sent through Glogin to the Master process that receives them and change the scene representation.
- The simulation interactive parameters: They make possible to change between different simulation profiles that have been sent as input parameters. The user can choose, from the Java plugin inside MD, to simulate a different profile of parameters while the previous simulation is already running. It is possible thanks to the MPI broadcast that the Master process does to the Slaves once it receives the remote command of changing the simulation behaviour. Once the Slaves

read the new input profile, they continue the simulation in a synchronized way as before.

Summing up, the trajectories are calculated in the Slaves and collected by the Master, which creates the OpenGL graphics sent through Glogin from the Grid to the MD in the user machine. In the other direction, the user interacts with Gvid plug-in, and these events are sent via Glogin to the Master node, which is in charge of notifying these changes to the Slaves using MPI messages to adapt their calculations.

This whole infrastructure working together is what makes the application capable of being executed in an interactive grid environment, taking advantage of the great amount of computing power of the resources.

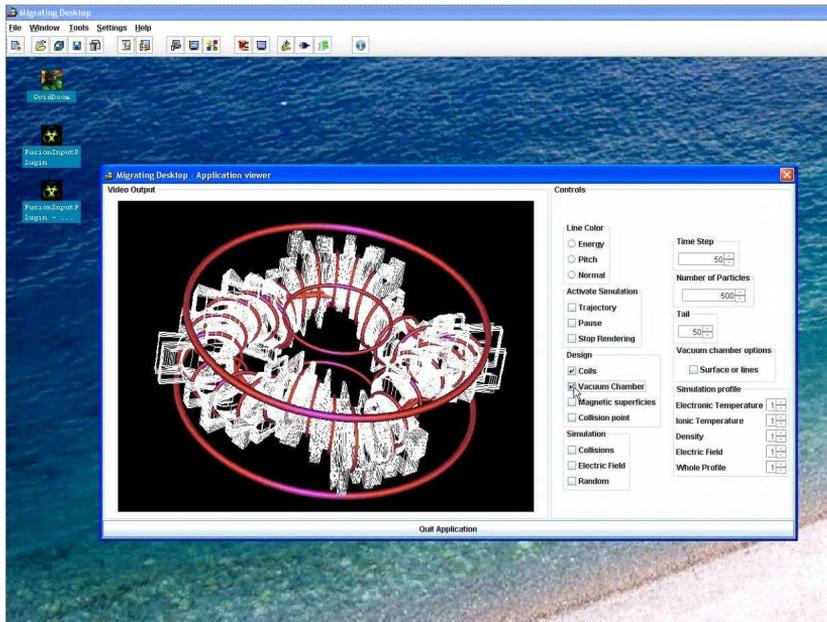


Fig. 3. IVISDEP running on int.eu.grid and visualized inside Migrating Desktop

4.3 Job Grid submission

The features and adaptations explained in the two previous sections show how the user can visualize and interact with the application once it is running on the infrastructure, but the very first step is submitting it as an interactive job.

As explained before, the Migrating Desktop is a virtual and user friendly tool that is available to all the researchers to send jobs to the Grid and also to manage their stored data. An input Java plugin for IVISDEP has also been developed

to define the input parameters of the first simulation submitted to the Grid and also the input files required. The user has also a generic bundle of tabs to define from graphical menus the description of the jdl job file, that will be used by the CrossBroker [15] (the brain of the grid infrastructure) to decide where to execute the described job.

5 CONCLUSIONS

IVISDEP has been successfully integrated in the int.eu.grid framework during the first year of the project. The adaptations and the improvements done in the application have made it able to run interactively in the grid joining all the middleware pieces.

The quality of the result has been recognized by the global Grid community, as it was awarded during the Manchester User Forum 9th–11th 2007.

The work in the application continues both to include new features and to prove that interactivity in a high layered infrastructure like the GRID is possible with the suitable tools properly integrated.

The final objective of this application is to become a framework able to simulate different topologies of fusion devices like ITER, adapted for the different lines of investigation in the field of Plasma Fusion and to make it accessible as many researchers as possible to take advantage of it.

REFERENCES

- [1] BOOZER, A. H.: Physics of Magnetically Confined Plasmas. Reviews of Modern Physics, Vol. 76, 2005, No. 4, pp. 1071–1141.
- [2] TJ-II GROUP: TJ-II Project: A Flexible Heliac Stellarator. Fusion Technology, Vol. 17, 1990, No. 1, pp. 131–139.
- [3] GUASP, J.—LINIERS, M.: Nuclear Fusion, Loss Cone Structure for Ions in the TJ-II Helical Axis Stellarator Part I: Properties without a Radial Electric Field, Vol. 40, 2000, No. 3, pp. 397–409.
- [4] CASTEJÓN, F.—FERNANDEZ, L. A.—GUASP, J.—MARTIN-MAYOR, V.—TARANCÓN, A.—VELASCO, J. L.: Ion Kinetic Transport in the Presence of Collisions and Electric Field in TJ-II ECRH Plasmas. Plasma Physics and Controlled Fusion, Vol. 49, 2007, No. 6, pp. 753–776.
- [5] KLOEDEN, P. E.—PLATEN, E.: Numerical Solution of Stochastic Differential Equations. Springer Verlag, 1992.
- [6] ARNOLD, L.: Stochastic Differential Equations: Theory and Applications, John Wiley and Sons, 1974.
- [7] KLOEDEN, P. E.—PEARSON, R. A.: The Numerical Solution of Stochastic Differential Equations. J. Austral. Math. Soc., Ser. B, 1992.
- [8] CASTEJÓN, F.—FERNANDEZ, J.—MARTIN-MAYOR, V.—TARANCÓN, A.—VELASCO, J. L.—ESTRADA, T.: Ion heating in Transitions to CERC in the Stel-

- larator TJ-II. 34th European Physical Society Conference on Plasma Physics, 2007, Warsaw (Poland).
- [9] CASTEJÓN, F.—LÓPEZ-FRAGUAS, A.—TARANCÓN, A.—VELASCO, J. L.: Searching for a Flux-Expansion Divertor in TJ-II. 17th Toki Conference and 16th Stellarator/Heliotron Workshop, 2007, Toki (Japan).
- [10] CASTEJÓN, F.—FERNANDEZ, L. A.—GUASP, J.—MARTIN-MAYOR, V.—TARANCÓN, A.—VELASCO, J. L.: Grid Computing for Kinetic Transport in Magnetic Fusion Devices. Second EGEE User Forum 9th–11th, 2007, Manchester (UK).
- [11] GABRIEL E. *et al.*: Open MPI: Goals, Concept, and Design of a Next Generation MPI Implementation. Lecture Notes in Computer Science (LNCS), Vol. 3241, Springer, 2004, pp. 97–104.
- [12] KUPCZYK, M.—LICHWAŁA, R.—MEYER, N.—PALAK, B.—PŁÓCIENNIK, M.—WOLNIEWICZ, P.: “Applications on Demand” as the Exploitation of the Migrating Desktop. Future Generation Computer Systems, Vol. 21, 2005, No. 1, pp. 37–44.
- [13] ROSMANITH, H.—KRANZLMÜLLER, D.: glogin – A Multifunctional Interactive Tunnel into the Grid. In: R. Buyya (Ed.), Proc. Grid 2004, 5th IEEE/ACM Intl. Workshop on Grid Computing, IEEE Computer Society, ISBN 0-7695-2256-4, Pittsburgh, PA, USA, pp. 266–272, 2004.
- [14] POLAK, M.—KRANZLMÜLLER, D.: Interactive Videostreaming Visualization on Grids. In: J. Dongarra, B. Tourancheau (Eds.), Cluster and Computational Grids for Scientific Computing, Future Generation Computer Systems, Special Section, Elsevier Science, Amsterdam, The Netherlands, Vol. 24, 2008, No. 1, pp. 39–45.
- [15] HEYMANN, E.—SENAR, M. A.—FERNÁNDEZ, E.—FERNÁNDEZ, A.—SALT, J.: Managing MPI Applications in Grid Environments. In: Grid Computing, Second European Across Grids Conference, AxGrids 2004, Lecture Notes in Computer Science, Vol. 3165, Springer, 2004, pp. 42-50.



Ruben Valles PÉREZ studied computer engineering at CPS (Centro Politecnico Superior de ingenieros) in Zaragoza. He started to work at BIFI in 2005 collaborating in different projects related to software development and at the beginning of 2006 he joined the cluster and grid computing research group. He is currently in charge of the int.eu.grid project at BIFI and is also the site administrator of the grid node. He also works on related grid projects such as EGEE II.