241

ASTRA-Matlab Integration for the Control of Tokamaks

Goretti SEVILLANO, Izaskun GARRIDO, Aitor J. GARRIDO

Department of Automatic Control and Systems Engineering
EUITI of Bilbao University of the Basque Country
Plaza de la Casilla, 3 48012 Bilbao, Spain
e-mail: mgsevillano001@ikasle.ehu.es

Abstract. The current increment in energy consumption has renewed the interest in the development of alternatives to fossil fuels. In this regard, the interest in solving the different control problems existing in nuclear fusion reactors like Tokamaks has been intensified. The aim of this manuscript is to show how the ASTRA code, which is used to simulate the performance of Tokamaks, can be integrated into the Matlab-Simulink tool in order to make easier the development of suitable controllers for Tokamaks. As a demonstrative case study to show the feasibility and the goodness of the proposed integration, a modified anti-windup PID-based controller coupled with an optimization algorithm for the loop voltage has been implemented. This integration represents an original and innovative work in the Tokamak control area and it provides new possibilities for the development and application of advanced control schemes to the standardized ASTRA code.

Keywords: plasma physics, Tokamak control, modelling and simulation, Matlab.

1. Introduction

As a consequence of the current worldwide increasing demand of energy together with the growing concern over the increasing pollution, CO_2 emissions and climate change the research in new energy sources, alternatives to fossil fuels, has gain more relevance in the last years (Weisz, 2004; Shell International, 2001). Between the unconventional resources that may help to meet this energy demand nuclear fusion has emerged as a promising source, what has led to an increasing interest in the control of plasma in fusion processes. In recent years, substantial efforts and resources have been dedicated for the development of clean nuclear technology based on fusion processes. As a result of these efforts several research papers have been published, especially in the field of Control Engineering applied to fusion processes (Ariola and Pironti, 2008; Pironti *et al.*, 2005; Schultz *et al.*, 2006).

Thus, although the controlled fusion is still a technological challenge, fusion reactors present important advantages over other existing energy sources (ITER, Internet). The first advantage that can be remarked is the existence of fuel supply for several thousand years. Besides, unlike other energy supplies, the product of fusion reaction, helium

doesn't contribute to air pollution, greenhouse effect or acid rain. Furthermore, contrary to fission, fusion doesn't produce long-lived radioactive isotopes and it is a safe reaction, since there is no risk of a large scale nuclear accident (Pironti *et al.*, 2005).

Nuclear fusion is based in the interaction of two light nuclei which fuse into a heavier and more stable nucleus producing a large amount of energy (Pironti *et al.*, 2005). Those light nuclei are contained into an ionized gas called plasma (Garrido *et al.*, 2007). This plasma can be confined using electromagnetic forces generated by external magnetic fields, which is known as magnetic confinement. Nowadays, the most promising magnetic confinement system is the Tokamak (Wesson, 2004). A Tokamak basically is a toroidal device (see Fig. 1) that confines the hot plasma using a helical magnetic field (Pironti *et al.*, 2005; Rodriguez, 2004; Garrido *et al.*, 2008).

Most of the controllers presently working on Tokamaks have been designed by decoupling the controls as much as possible. These controllers are usually based on proportional integral derivatives (PIDs) with a multiloop structure, which are partly designed on the basis of simple models of the system to be controlled and they are usually fine-tuned during Tokamak operation. In the past this has often meant that the effect on a particular plasma parameter from coils other than the coil being used to control that parameter was ignored. Having always in mind the different time scales, recent efforts seek to design more complex controllers focusing the attention on these effects (Wesson, 2004).

In this manuscript we describe an integration of the ASTRA transport code within the Matlab-Simulink tool, with emphasis on its application for rapid controller prototyping. The resulting code is able to run on a standard PC with Matlab-Simulink under the Linux Operating System. In order to demonstrate the capabilities and performance of this environment to quickly implement different controllers, a case study consisting of a modified anti-windup PID-based controller for a trajectory tracking problem, tuned using a combined Ziegler–Nichols method jointly with a gradient descent-based optimization
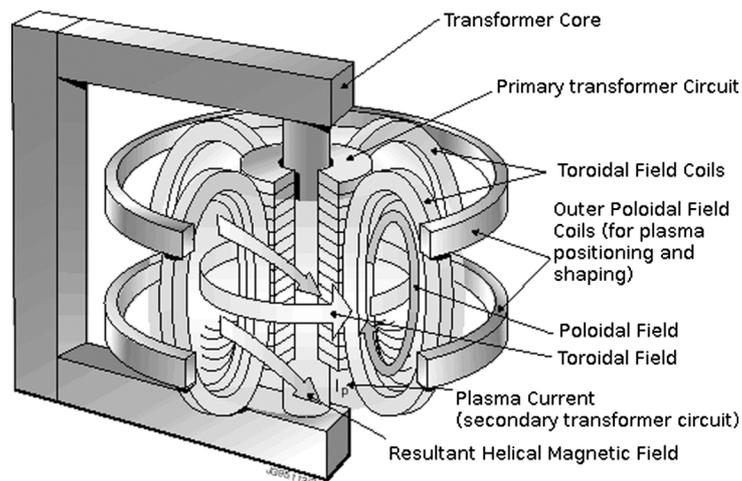

Fig. 1. Scheme of a typical Tokamak.

technique provided by Simulink, will be presented. The choice of this kind of control has been determined by the fact that, despite all the advances in science and technology in recent decades in the field of automatic control, the fact is that PID-based controllers remain the most widely used schemes in industrial process control. In this sense, it can be remarked that based on a survey of over eleven thousand controllers in diverse industries, 97% of regulatory controllers utilize PID feedback Desborough, 2001, so that nowadays most of the feedback loops are controlled using this algorithm or some variant thereof. There is therefore an unquestionable acceptability of this kind of controllers and despite its longevity this trend is unlikely to change in the immediate future.

The rest of the paper is organized as follows. In Section 2, the ASTRA code used for modelling the Tokamak performance is presented and the equations used for constructing the model are stated. Next, in Section 3, the proposed ASTRA-Matlab integration is described. In the following Section 4, a model for the plasma flux and the corresponding control are presented followed by some simulation results comparing different controlled cases which have been obtained using the combination of the implemented ASTRA-Matlab integration with the proposed PID-based anti-windup control scheme implemented as a demonstrative case study. Finally, in Section 5, some concluding remarks are presented.

2. ASTRA Code

2.1. *General Description of the ASTRA Code*

The ASTRA (Automatic System for Transport Analysis) is a tool for generating computer codes for the solution of transport problems in magnetically confined plasmas. ASTRA is a standard tool for generating computer codes to solve transport problems in magnetically confined plasmas (Pereverzev and Yushmanov, 2002). It is a useful tool for the study of transport mechanisms in reactor-oriented facilities of Tokamaks. ASTRA solves coupled time-dependent 1-D transport equations for particles, heat and current and 2-D MHD (Magnetohydrodynamic) equilibrium self consistently with realistic Tokamak geometry (Na, 2003).

In order to achieve higher effectiveness, the flexibility provided by ASTRA allows the user to customize the code. This flexibility relies on the wide choice of standard relationships, functions and subroutines representing various transport coefficients, equilibrium solvers, methods of auxiliary heating (e.g., NBI) and other physical processes and data treatment in the Tokamak plasma. Another interesting feature of ASTRA is that it generates interactive codes. This means that the user in addition to observe the time evolution of plasma parameters during the program execution, can also interrupt the execution or change the data presentation and control parameters influencing the course of modelling (Pereverzev and Yushmanov, 2002).

Therefore, the ASTRA code is considered a transport code with a flexible programming system capable of creating numerical codes for predictive or interpretative transport modelling for stability analysis and for processing experimental data.

2.2. *ASTRA Background Equations and Formulae*

In the ASTRA code, the magnetic system is considered toroidally symmetric and two coordinate systems are used: a cylindrical coordinate system (r, φ, z) with the polar axis coinciding with the major axis of the torus and another coordinate system (a, θ, ζ) associated to the magnetic geometry of the Tokamak where a denotes the radial variable which is an arbitrary label of a magnetic flux surface (see Fig. 2), θ is the poloidal angle and the toroidal angle is chosen $\zeta = -\varphi$ (Pereverzev and Yushmanov, 2002). Taking into account that the local flux g can be expressed by (1) and the function of magnetic surface $F(a)$ has to satisfy the diffusion equation (2), where two functions of a single argument a given by (3) and (4) have to be introduced.

$$g(a, \theta) = F(a)\tilde{v}(a, \theta) - \tilde{D}(a, \theta)\nabla F(a), \tag{1}$$

$$\frac{\partial F}{\partial t} = \frac{\partial}{\partial V}\left(\langle(\nabla V)^2\tilde{D}\rangle\frac{\partial F}{\partial V} - F\langle\nabla V \cdot \tilde{v}\rangle\right) + S(a), \tag{2}$$

$$D(a) = \langle(\nabla a)^2\tilde{D}\rangle/\langle(\nabla a)^2\rangle, \tag{3}$$

$$v(a) = \langle\nabla a \cdot \tilde{v}\rangle/\langle|\nabla a|\rangle. \tag{4}$$

Considering those definitions independent of the choice of the magnetic surface label a it is possible to rewrite (2) in the form used in ASTRA given by (5). Using those definitions the total flux and the average flux density on a magnetic surface can be expressed by (6) and (7).

$$\frac{\partial F}{\partial t} = \frac{\partial a}{\partial V}\frac{\partial}{\partial a}\left[\frac{\partial V}{\partial a}\langle(\nabla a)^2\rangle\left(D\frac{\partial F}{\partial a} - \frac{\langle|\nabla a|\rangle}{\langle(\nabla a)^2\rangle}vF\right)\right] + S(a), \tag{5}$$

$$\Gamma(a) = \frac{\partial V}{\partial a}\left(\langle|\nabla a|\rangle vF - \langle(\nabla a)^2\rangle D\frac{\partial F}{\partial a}\right), \tag{6}$$

$$\gamma(a) = vF - \frac{\langle(\nabla a)^2\rangle}{\langle|\nabla a|\rangle}D\frac{\partial F}{\partial a}. \tag{7}$$
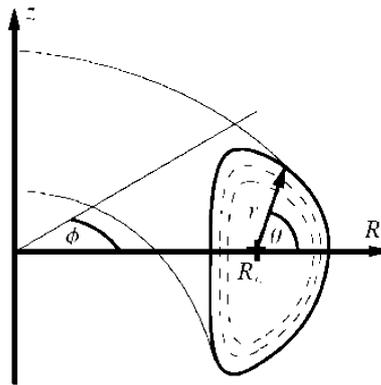

Fig. 2. Axisymmetric configuration of the plasma (dashed lines represent the magnetic surfaces).

The magnetic field can be then obtained by (8) and the current density by (9), where Ψ and I are defined by (10) and (11) respectively, being R_0 the distance from the axis of the torus to a fixed point in the plasma and B_0 the vacuum magnetic field at the point where $r = R_0$.

$$B = I\nabla\zeta + \frac{1}{2\pi}[\nabla\Psi \times \nabla\zeta], \tag{8}$$

$$j = -\frac{\nabla\zeta}{2\pi\mu_0}r^2\mathrm{div}\frac{\nabla\Psi}{r^2} + \frac{1}{\mu_0}[\nabla I \times \nabla\zeta], \tag{9}$$

$$\psi = -\Psi = \frac{1}{2\pi}\int_V B\cdot\nabla\theta\,d^3x = \int_{S_\theta} B\cdot dS_\theta, \tag{10}$$

$$I = R_0 B_0 - \frac{\mu_0}{2\pi}\int_{S_\theta} j\cdot dS_\theta. \tag{11}$$

Although functions Ψ and I are surface functions, which means that they depend on space coordinates through the variable a, since they describe an evolving plasma they also depend on time, that is why each of them can be used as radial coordinate instead of a (Pereverzev and Yushmanov, 2002). Once at this point, it is convenient to introduce another two surface functions defined by the following equations: toroidal magnetic flux Φ and the effective minor radius ρ.

$$\Phi = \int_{S_\zeta} B\cdot dS_\zeta = \frac{1}{2\pi}\int_V \frac{I}{r^2}\,d^3x, \tag{12}$$

$$\rho = \sqrt{(\Phi/(\pi B_0))}. \tag{13}$$

The plasma equilibrium in a Tokamak is determined by the Grad–Shafranov equation (14), where $p = p(\rho, t)$ represents the plasma pressure with the contribution of all plasma species and I is the diamagnetic current previously defined in (11).

$$\Delta^*\psi = r^2\mathrm{div}\frac{\nabla\psi}{r^2} = -4\pi^2\left(\mu_0 r^2\frac{\partial p}{\partial\psi} + I\frac{\partial I}{\partial\psi}\right). \tag{14}$$

The ASTRA code presents a special notation with the aim of simplifying the notation which is explained in detail in Pereverzev and Yushmanov (2002). It also uses the transport equations shown in Table 1 that may be expressed in terms of thermodynamic forces taken as derivatives with respect to ρ, which makes possible to write the equilibrium equation (14) in terms of the functions provided by transport equations as:

$$\Delta^*\psi = 2\pi\mu_0 R_0\left[\frac{J}{\langle B^2/B_0^2\rangle}\left(j_{||} + \frac{R_0}{B_0\rho\mu}\frac{\partial p}{\partial\rho}\right) - \frac{r^2}{B_0 R_0\rho\mu}\frac{\partial p}{\partial\rho}\right]. \tag{15}$$

Table 1

Transport equations in ASTRA code

1. Equation for the electron density n_e, where Γ_e is the electron flux through a flux surface $\rho = \text{const}$ and S_e represents the source of electrons

$$\frac{1}{V'}\left(\frac{\partial}{\partial t} - \frac{\dot{B}_0}{2B_0}\frac{\partial}{\partial \rho}\rho\right)\left(V'n_e\right) + \frac{1}{V'}\frac{\partial}{\partial \rho}\Gamma_e = S_e.$$

2. Equation for electron temperature T_e, where q_e is the electron heat flux through a flux surface $\rho = \text{const}$ and P_e represents the energy of the source of electrons

$$\frac{3}{2}\left(V'\right)^{-5/3}\left(\frac{\partial}{\partial t} - \frac{\dot{B}_0}{2B_0}\frac{\partial}{\partial \rho}\rho\right)\left[(V')^{5/3}n_eT_e\right] + \frac{1}{V'}\frac{\partial}{\partial \rho}\left(q_e + \frac{5}{2}T_e\Gamma_e\right) = P_e.$$

3. Equation for ion temperature T_i, where $n_i = n_e/Z_i$ and $\Gamma_i = \Gamma_e/Z_i$ and with q_i as the ion heat flux through a flux surface $\rho = \text{const}$ and P_i representing the energy of the source of ions

$$\frac{3}{2}\left(V'\right)^{-5/3}\left(\frac{\partial}{\partial t} - \frac{\dot{B}_0}{2B_0}\frac{\partial}{\partial \rho}\rho\right)\left[(V')^{5/3}n_iT_i\right] + \frac{1}{V'}\frac{\partial}{\partial \rho}\left(q_i + \frac{5}{2}T_i\Gamma_i\right) = P_i.$$

4. Equation for the poloidal flux ψ, where $\sigma_{||}$ represents the conductivity and by considering $\vec{j}_{BS}$ and $\vec{j}_{CD}$ as the bootstrap current density and the density of the current driven by external sources

$$\sigma_{||}\left(\frac{\partial \psi}{\partial t} - \frac{\rho\dot{B}_0}{2B_0}\frac{\partial \psi}{\partial \rho}\right) = \frac{J^2R_0}{\mu_0\rho}\frac{\partial}{\partial \rho}\left(\frac{G_2}{J}\frac{\partial \psi}{\partial \rho}\right) - \frac{V'}{2\pi\rho}(j_{BS} + j_{CD}).$$

3. ASTRA-MATLAB Integration

The motivation of this work is to help developing different controllers for Tokamak reactors in a more convenient and efficient way. Keeping this objective in mind, two widely extended tools have been combined. The ASTRA standard transport code, which reliability as Tokamak simulator is widely recognized, has been embedded in the well-known Matlab-Simulink software, as shown in Fig. 3. On the one hand, such integration composes a very valuable tool for control design, since researchers can develop and implement controllers for diverse Tokamak models using the capabilities and advantages of the Simulink environment in an easy and intuitive way. On the other hand, there already exist different control codes that have been developed in Matlab using parameter estimation or other system identification techniques (Zutautaite-Seputiene *et al.*, 2010; Atanasov and Ichtev, 2010; Vörös, 2010), with accurate Tokamak descriptions but lacking the capabilities of the transport code provided by ASTRA, which could benefit from this tool by developing coupled model control systems. Moreover, embedding ASTRA transport code into Matlab allows inexpensive upgrades such as the implementation of high per-
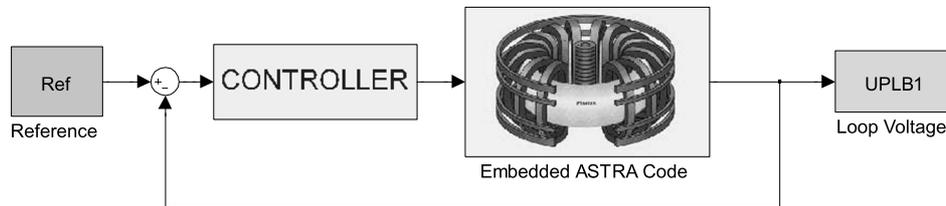

Fig. 3. Simulink closed loop system diagram with embedded ASTRA code.

formance compilation if needed (Chang, 2010). In order to prove its performance, the proposed integration is validated in the next section. For this purpose, a Simulink loop voltage control is implemented as a case study so that the loop voltage is regulated by using the plasma current that has previously been obtained by ASTRA solving the Grad-Shafranov equation at each time step.

One of the advantages of ASTRA over other codes lies in that it allows the user not only to check its parameters at execution time, but also to save and modify them. Therefore, in order to achieve the integration of the ASTRA into Matlab the first issue to deal with was the need to automate not only the ASTRA execution but also to make the necessary modifications so as to keep the user interaction capabilities. Thus, two interface modules had to be programmed: ASTRA-Matlab data conversion and its corresponding inverse transformation. Besides, it was also necessary to create a C-module linked to the ASTRA code to automate the communication between both codes without the intervention of the user. This software development, which scheme may be observed in Fig. 4, represents the main part of the work and it is composed of the following elements:

- *ASTRA execution automation*. This module integrates the ASTRA Init and Interrupt routines by incorporating source code that implements specific functions to automate each ASTRA iteration. This module adds the necessary source code to
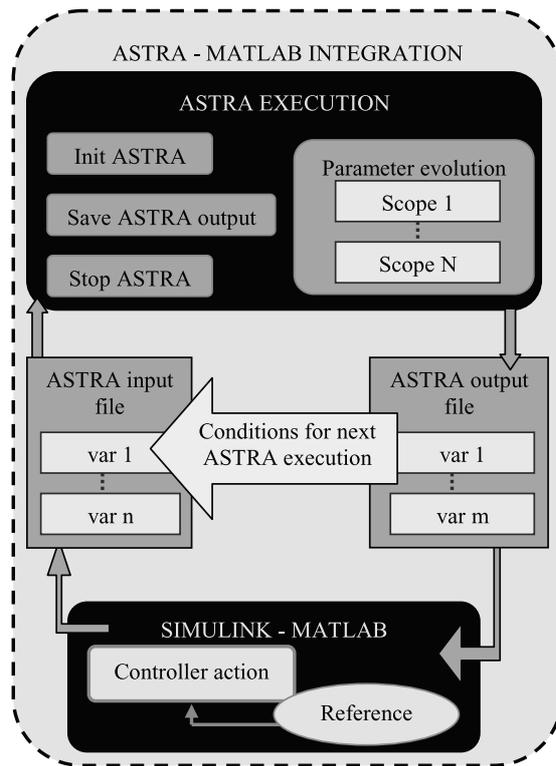

Fig. 4. Software flow diagram of ASTRA automation.

ASTRA so as to automate the output and to modify its input specifications according to the output generated from Simulink without user intervention.

- *ASTRA output and input automation*. Besides the possibility of interaction between the user and ASTRA represents one of the advantages of this code, this module takes care of modifying automatically the initial and final values for each iteration and it also keeps the possibility of the user interaction preserving all ASTRA capabilities.

- *Parameter exchange between different environments*. An intermediate module transfers the output variables from the scope of the ASTRA module to the Simulink environment. This interface module only handles communication parameters (number and value of inputs, outputs, sampling time, etc.). Several procedures were explored to integrate data into the Simulink environment such as C-mex S-function and Perl scripts (Wall *et al.*, 2000). The choice of an adequate method was determined by the amount of data to be converted, since the computational speed that an S-function provides is easily overcome by the simplicity of Perl scripts when the data communication flow is limited. In this module the data conversion from ASTRA scope to Simulink environment and its inverse transformation is also carried out.

Once this set of modules has been implemented, all these functions related to the ASTRA operation and the interconnection between ASTRA and Matlab are collected in a unique Simulink block (see Fig. 3). The features of Simulink programming tool make possible to embed ASTRA as a Matlab-Simulink block that may be easily combined it with other subroutines and control schemes without the need of further modifications. Figure 5 shows time evolution of some of the reprentative parameters (electron density
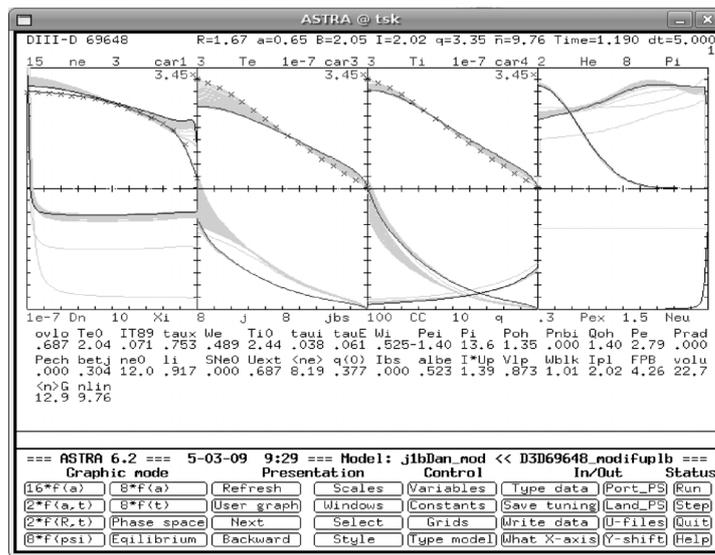

Fig. 5. Plasma parameter evolution of ASTRA – simulink integration.

(n_e), electron temperature (T_e), ion temperature (T_i), current density $(j, \ldots)$ of the plasma in a Tokamak that can be observed thanks to the wide capabilities offered by the interactive graphical user interface.

This work is even more relevant when considering there already exist several Matlab codes lacking current profiles but integrating diverse Tokamak models that may be used to provide the boundary conditions to the transport code of the type shown in Fig. 6 which, in this particular case, have been derived from the parameters of the input file of our case study. Therefore, as indicated before, the benefits of this work are twofold: on the one hand this integration is a valuable tool for testing different controls for the currents, which can be easily tested in Matlab via Simulink, before their implementation in real time over the system, and on the other hand, an inner second control loop may be implemented coupling ASTRA with other Tokamak models so as to extend the study and control to other variables as, for example, the vertical displacement of the plasma (Na, 2003).

Please, note that the objective of the proposed ASTRA-Matlab integration is not to develop real-time control code to be directly applied to the Tokamak. Instead, the aim of the mentioned integration is to serve as a valuable tool that may be helpful in the development and testing of different kind of controllers for these devices. Once established, these controllers may be transferred to the real system whose plant was represented by the ASTRA code during the previous simulation stage.
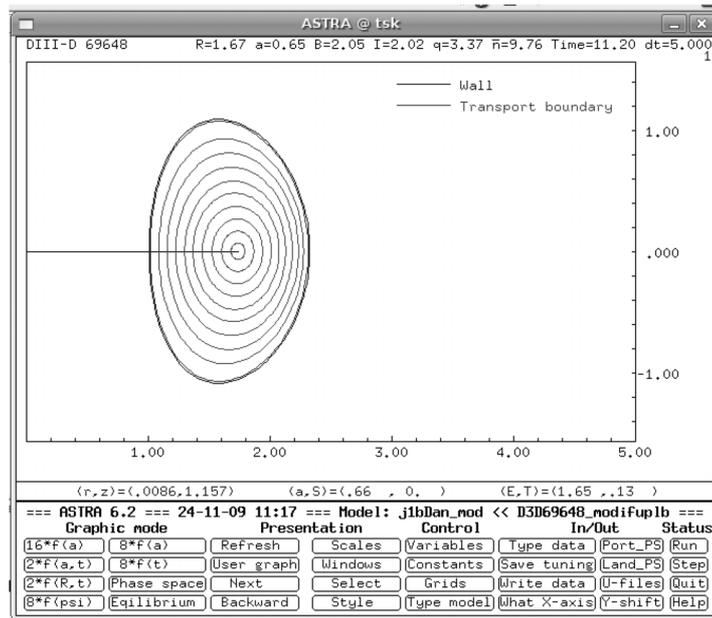

Fig. 6. Plasma boundary from our ASTRA-Matlab case study provided by the interactive graphic interface.

4. Case Study: Loop Voltage Control

In order to show how to apply rapid control prototyping with the proposed ASTRA-Matlab integration, a demonstrative case study consisting of a compensator that controls the loop-voltage of typical plasma has been considered. In order to built the model for a typical plasma and be able to carry out the simulations, the ASTRA code required some input files which were provided by the Spanish Fusion National Laboratory at CIEMAT (Centre for Energy, Environment and Technology Research). In particular, the compensator has to regulate the non-inductive current that the actuators (usually LHCD – Lower Hybrid Current Drive or ECRH – Electron Cyclotron Resonance Heating) must provide based on the error signal, $e(t) = v_0(t) - v_{loop}(t)$, where $v_0(t)$ and $v_{loop}(t)$ are the reference and output loop voltage signals respectively. For this purpose a modified anti wind-up PID-based controller with an auto-tuning rule coupled with an optimal control strategy to govern the plasma current profile has been implemented.

PID (Proportional-Integral-Derivative) control is one of the earlier control strategies and its use is widely extended (Desborough, 2001; Knospe *et al.*, 2006; Johnson and Moradi, 1995; Aström and Hägglund, 2005). At this respect, it may be recalled the last work of Åström and Miller (2009) "Feedback Systems" where it is stated that over 95% of all industrial control problems are solved by PID control. Besides, since the final objective of the proposed control system is to be implemented over a real plant, one of the requisites is to maintain the loop voltage control as simple as possible while meting the system performance requirements. In this sense, other control schemes were initially considered, as adaptative and learning controllers (De la Sen and Almansa, 2002; Liutkevicius, 2003; Kaminskas and Liutkevicius, 2002; 2005), ANN-based (Amundarain *et al.*, 2009; Ibeas and de la Sen, 2004, Garliauskas, 2010) or robust sliding-mode controllers (Barambones and Garrido, 2007), but they were finally discarded for the shake of implementation reliability.

One of the issues present when dealing with PID-based controllers is that their performance may be affected due to, among other factors, an inadequate tuning of the controller parameters (Aström and Hägglund, 2005). Hence, although auto-tuning is a very desirable feature and almost every PID controller implemented in a real plant provides it nowadays; it is always necessary further fine tuning in order to achieve the desired performance. In this sense, the integration proposed in this paper allows the engineer to take advantage of the many different tuning algorithms already implemented in the Matlab-Simulink tool that allows obtaining the best parameter values of the PID with the help of the optimization toolbox. In this particular case, the implemented tuning method consists of two steps: An auto-tuning based on a Ziegler–Nichols method and, a secondary fine tuning using optimal control based on a gradient descent technique provided by Simulink.

Thus, the ASTRA-Matlab integration allows using the Simulink features for the tasks of designing and tuning the compensator, so that the general procedure consists of linealizing the plant model in order to apply an initial tuning and then, evaluating the performance of the system, fine tuning the controller parameters using optimal control over the closed-loop system. Once the performance meets the requirements, it is brought back

into the original diagram and tested against the non-linear plant. Due to the differences that always exist between the model and the aforementioned real plant, the values of controller parameters obtained using auto-tuning methods must be usually readjusted over the real system. Nevertheless, results showed that the proposed tuning procedure was easily performed and provided an adequate response within the operation range of the system.

In particular, the case study that has been considered is a typical PID-based trajectory tracking problem whose aim is to govern the loop-voltage ($v_{loop}(t)$) by means of the plasma current ($I_p(t)$), so that the variable controlled is the loop-voltage and the manipulated variable is the plasma current regulated by the control actuators. Thus, the reference value for the loop voltage and the feedback of the system output value generate the error signal defined by $e(t) = v_0(t) - v_{loop}(t)$, which will serve as input to the controller. Then, the control signal $u(t)$ drives the LHCD and ECRH control actuators that will provide the necessary current to acquire the demanded loop voltage. In this way, the modulation of the plasma current adequately adjusts the loop voltage of the Tokamak at each time instant.

The general expression for a traditional PID, including a proportional action modulated by an integral action to eliminate the steady-state error and the derivative action to stabilize the system is given by:

$$u_{presat}(t) = K_p e(t) + K_i \int e(t)\,dt + K_d \frac{de(t)}{dt},$$ (16)

whose Laplace transform can be expressed as:

$$U_{presat}(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s\right) E(s).$$ (17)

Nevertheless, since the control actuators (LHCD and ECRH) are subject to saturation, it was necessary to consider an integral wind-up effect. It must be considered that these additional plasma heating actuators present physical limits, so that once they have reached saturation, increasing the magnitude of the control signal further has no effect. In this case, there exists a difference between the desired and the measured output loop-voltage values, and the resulting error will cause a continuous increment in the integral term. When the error term changes its sign, the integral term starts to unwind, and this can cause long time delays and possible instability. When this happens, the feedback loop is broken and the system runs in open loop because control actuators remain at their limits independently of the loop voltage output values as long as the actuators remain saturated.

In order to avoid this phenomena, the modified anti-windup controller considered (see Fig. 7) whose Laplace transform equation is given by (18) includes an extra feedback path that is generated by measuring the actual actuators output $u(t)$ subject to saturation, so that the saturation error signal $e_{sat}(t)$ is defined as the difference between the output of the controller $u_{presat}(t)$ and the output of the actuator as shown in (19). This corrective
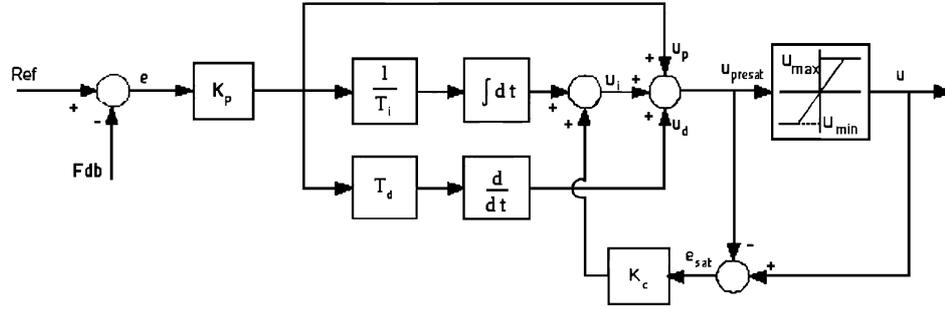
Fig. 7. Detailed scheme of the modified anti-windup controller.

error signal is fed to the input of the integrator through the gain K_c, so that when there is no saturation its value is zero having no effect on the control signal, but when the actuators saturate, the signal is fed back to the integrator in such a way that the integral action $u_i(t)$ is decreased accordingly with the saturation error. This implies that controller output $u_{presat}(t)$ is kept close to the actuators saturation limit and the integral windup is eliminated.

$$U_{presat}(s) = K_p E(s) + K_p \frac{1}{T_i s} E(s) + K_c E_{sat}(s) + K_p T_d s E(s), \tag{18}$$

$$E_{sat}(s) = U(s) - U_{presat}(s), \tag{19}$$

$$U_{presat}(s) = \frac{K_p}{1 + K_c} \left(1 + \frac{1}{T_i s} + T_d s \right) E(s) + \frac{K_c}{1 + K_c} U(s). \tag{20}$$

As it has been indicated, the controller tuning is performed in two steps, allowing simultaneous quasi-optimization of the PID parameters subject to the actuator constraints, based on a first parameter approximation using Ziegler–Nichols over the closed loop system. In particular, Ziegler–Nichols tuning rules provide a practical and systematic way of tuning PID loops that afford initial quarter decay ratio (QDR) responses that may be readjusted to achieve the required performance (Johnson and Moradi, 1995; Aström and Hägglund, 2005; Ziegler and Nichols, 1942; Zhong, 2006; VanDoren, 2003; Xue *et al.*, 2008). These tuning rules are integrated into Simulink as a choice to automatically determine the controller. Besides, it is used the aforementioned Optimization Toolbox by means of the Response Optimization Algorithm within Simulink Control Design (Matworks webminars, Internet) to adequately refine the tuning. For the case-study at hand, the values obtained for the controller parameters defined in (17) are shown in Table 2 – Controller 1.

For the case study considered, a reference tracking problem has been chosen, so that one of the requirements to be satisfied is to cancel the steady-state error, which can be defined by (21) where $e(t)$ is the tracking error and $E(s)$ represents its Laplace transform. Therefore, once the controller parameter values have been obtained as indicated, the closed-loop response of the system is analyzed (see Fig. 8).

Table 2

Case study controllers

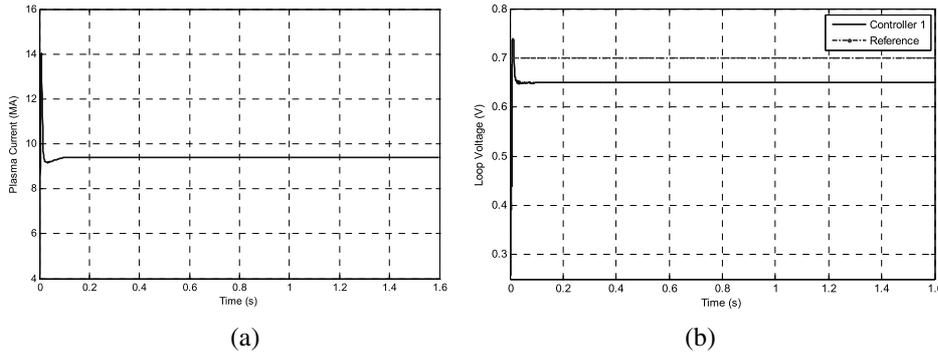| Controller 1	$K_p = 11.3$	$K_i = 2507$	$K_d = 0.018$
Controller 2	$K_p = 6.3$	$K_i = 5478.3$	$K_d = 0.0018$

Fig. 8. (a) Manipulated variable (plasma current) for Controller 1. (b) Controlled variable (loop voltage) for Controller 1.

$$e_{ss} = \lim_{t \to \infty} e(t) = \lim_{s \to 0} sE(s). \tag{21}$$

According to the time evolution of the controlled variable (loop voltage) shown in Fig. 8 (b), it can be concluded that although the system response is sufficiently fast, the steady-state error present on it is undesirable, as well as the perceived overflow. Therefore, the controller gains values must be readjusted using the described fine tuning procedure in order to cancel the steady-state error and reduce the overflow, even if this readjustment can result in a slowdown of the system response. The resulting controller new values are shown in Table 2 – Controller 2. As it may be observed, the controller integral gain (K_i) has been increased achieving the cancellation of the steady-state error and the value of the proportional gain (K_p) has been reduced, leading to a reduction in the overflow magnitude. The simulation results for the system response obtained with the readjusted controller are shown in Fig. 9. In particular, it can be concluded from Fig. 9(b) that the steady-state error of the system response has been eliminated and that the response overflow has been reduced.

Finally, in Fig. 10 the simulation results for the tracking error obtained for the system using both controllers defined in Table 2 are presented. Comparing the red dashed line, which represents the tracking error for the Controller 1, with the solid blue line, which represents the same error for the Controller 2, it can be observed that the aforementioned steady-state has been cancelled thanks to the fine tuning procedure applied. Besides, it is also noticeable that the undesirable overflow presents a peak reduction. Therefore, it can be concluded that the characteristics of the system response have been improved thanks
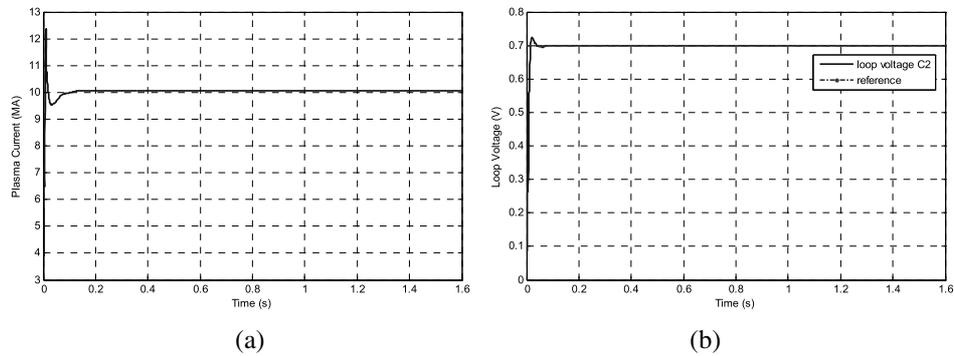
Fig. 9. (a) Manipulated variable (plasma current) for Controller 2. (b) Controlled variable (loop voltage) for Controller 2.
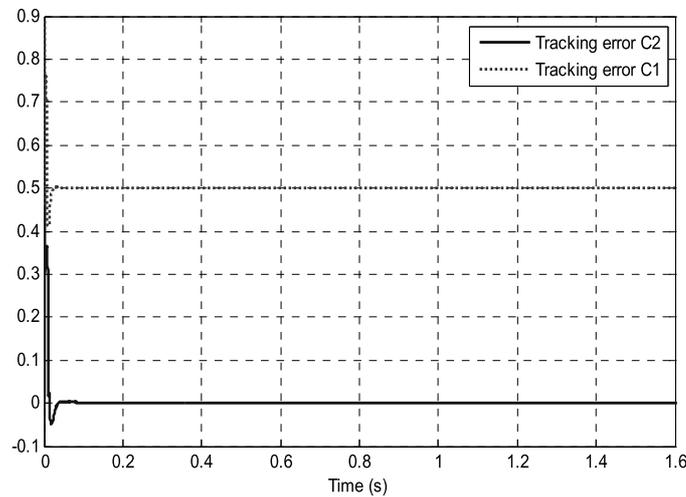

Fig. 8. Tracking error for the system response.

to the combined tuning procedure, allowing an adequate loop voltage reference tracking response.

5. Conclusions

The aim of this manuscript has been to propose the integration of the standardized AS-TRA transport code for Tokamaks within the Matlab-Simulink tool focusing on its application for rapid controller prototyping. The proposed ASTRA-Matlab integration using the Simulink as a programming tool in which the ASTRA code has been embedded within a control block, is not only very helpful for the development and application of advanced control schemes to the standardized ASTRA code for Tokamaks but it also paves the way

for further integration of ASTRA with models lacking the transport code but with accurate Tokamak description that will enable the development of control systems for nested variables.

With the purpose of demonstrating the capabilities and the performance of the code resulting from the proposed ASTRA-Matlab integration a case study consisting of a modified anti-windup PID-based controller has been analyzed. In order to avoid the problems that usually appears in PID-based controllers due to an inadequate tuning of the controller parameters, an auto-tuning algorithm based on Ziegler–Nichols method and a fine tuning using optimal control based on a gradient descent technique, both available within the Matlab-Simulink tool, have been employed. The example considered, consisting of a traditional reference tracking problem, has been used to illustrate how the proposed integration can be successfully applied to design loop voltage controllers using the plasma current as the manipulated variable which is regulated so as to achieve the desired performance of the system.

References

Amundarain, M., Alberdi, M., Garrido, A.J., Garrido, I. (2009). Neural control of the Wells turbine-generator module. In: *Proceedings of the 48th IEEE Conference on Decision and Control* (CDC2009). Shangai, China, pp. 7315–7320.

Ariola, M., Pironti, A. (2008). *Magnetic Control of Tokamak Plasmas*. Springer, Berlin.

Aström, K.J., Hägglund, T. (2005). *Advanced PID Control*. Instrument Society of America.

Åström, K.J., Miller, R. (2009). *Feedback Systems*. Princeton University Press, Princeton.

Atanasov, N., Ichtev, A. (2010). Closed-loop system identification with modifications of the instrumental variable method. *Informatica*, 21(2), 159–174.

Barambones, O., Garrido, A.J. (2007). Adaptive sensorless robust control of AC drives based on sliding mode control theory. *International Journal of Robust and Nonlinear Control*, 17(9), 862–879.

Chang, T. Y. (2010). An computation-efficient generalized group-oriented cryptosystem. *Informatica*, 21(3), 307–319.

De la Sen, M. Almansa, A. (2002). Adaptive stable control of manipulators with improved adaptation transients by using on-line supervision of the free-parameters of the adaptation algorithm and sampling rate. *Informatica*, 13(3), 345–368.

Desbourough, L., Miller, R. (2001). Increasing customer value of industrial control performance monitoring-Honeywell's experience. In: *Proceedings of the 6th Int. Conf. on Chemical Process Control*, Tucson, Arizona, pp. 172–192.

Garlauskas, A. (2010). An expansion of the neural network theory by introducing hebb postulate. *Informatica*, 21(3), 339–348.

Garrido, A., Garrido, I., Barambones, O., Alkorta, P. (2007). A survey on control-oriented plasma physics in tokamak reactors. In: *Proceedings of the 5th IASME/WSEAS Int. Conf. on Heat Transfer, Thermal Engineering and Environment*, Athens, Greece, pp. 284–289.

Garrido, A., Garrido, I., Barambones, O., Alkorta, P., Maseda, F.J. (2008). Simple linear models for plasma control in tokamak reactors. In: *Proceedings of the Int. Conf. on Control, Automation and Systems*, Seoul, Korea, pp. 2429–2432.

Global Business Environment Shell International (2001). *Energy Needs, Choices and Possibilities*. Scenarios to 2050.

Ibeas, A., de la Sen, M. (2004). Representations of multi-model based controllers by using artificial intelligence tools. *Informatica*, 15(3), 337–362.

ITER Organization [Internet]. `http://www.iter.org/`.

Johnson, M.A., Moradi, M.H. (1995). *PID Control*. Springer, London.

Kaminskas, V., Liutkevicius, R. (2002). Admics. *Informatica*, 13(3), 287–298.

Kaminskas, V., Liutkevicius, R. (2005). Learning fuzzy control of nonlinear processes. *Informatica*, 16(4), 571–586.

Knospe, C., Li, Y., Ang, K.H., Chong, G.C.Y., Kristiansson, B., Hara, S. *et al.* (2006). Special section on advances in pid control – various authors. *IEEE Control Systems Magazine*, 26(1), 30–91.

Liutkevicius, R. (2003). Coupled adaptive fuzzy control of nonlinear, time-varying plant. *Informatica*, 14(3), 323–336.

Na, Y.S. (2003). Modelling of current profile control in tokamak plasmas. PhD thesis, University of Munich, Munich.

Pereverzev, G.V., Yushmanov, P.N. (2002) *ASTRA. Automated System for Transport Analysis*. Max-Planck-Institut für Plasmaphysik.

Pironti, A., Walker, M., Ariola, M., Beghi, A., Cenedese, A., Ambrosino, G. *et al.* (2005). Special section on control of fusion-various authors. *IEEE Control Systems Magazine*, 25(5), 24–92.

Rodriguez, S. (2004). Estudio de Medios de Control de ELMS en Tokamaks. Proyecto fin de Carrera. Univ. Carlos III, Madrid (in Spanish).

Schultz, K.R., Walker, M., Humphreys, D.A., Mazon, D., Sartori, F., Lister, J.B. *et al.* (2006). Special section on control of fusion. Part II. Various authors. *IEEE Control Systems Magazine*, 26(2), 30–91.

The MarthWorks [Internet]. Mathworks webminars. Control System Toolbox. Available from: `http://www.mathworks.com/products/control/demos.html?file=/products/demos/shipping/control/GSSISOTool.html`.

VanDoren, V.J. (2003). Loop tuning fundamentals. *Control Engineering*.

Vörös, J. (2010). Recursive identification of systems with noninvertible output nonlinearities. *Informatica*, 21(1), 139–148.

Wall, L., Christiansen, T., Orwant, J. (2000). *Programming Perl*. 3rd edn. O'Reilly, Sebastopol.

Weisz, P.B. (2004) Basic choices and constraints on log-term energy supplies. *Physics Today*, 57, 47–52.

Wesson, J. (2004). *Tokamaks*. 3rd edn. Clarendon Press, Oxford.

Xue, D., Chen, Y.Q., Atherton, D.P. (2008). PID controller design. Linear feedback control analysis and design with Matlab. Society for Industrial and Applied Mathematics, pp. 183–235.

Zhong, J. (2006). PID Controller Tuning: A Short Tutorial. Mechanical Engineering. Purdue University. Available from: `http://saba.kntu.ac.ir/eecd/pcl/download/PIDtutorial.pdf`.

Ziegler, J.B., Nichols, N.B. (1942). Optimum settings for automatic controllers. *ASME Transactions*, 54, 729–768.

Zutautaite-Seputiene, I., Augutis, J., Telksnys, L. (2010). Parameters estimation in modelling of gas-gap in rbmk type reactor using Bayesian approach. *Informatica*, 21(2), 295–306.

G. Sevillano received the MSc degree in electronic engineering in 2006 from the University of the Basque Country (UPV-EHU) in Spain. She is currently a PhD student in the Automatic Control and Systems Engineering Department (UPV-EHU). Her present research interest areas are the control applications for nuclear fusion processes and reactors.

I. Garrido received a MSc in applied mathematics from the University of the Basque Country (UPV-EHU) in Spain, a MSc in numerical analysis and programming, and in 1999 a PhD in finite elements, both of them from Dundee University (UK). She has held several positions at PIK (Germany), ZIB (Germany) and UiB (Norway). She has been an invited researcher in institutions such as Stanford University or LLNL (USA). Since 2004 she holds a position as associate professor of systems and control engineering at the Automatic Control and Systems Engineering Department (UPV-EHU), where she is also vice dean of research and international relations. She has over 100 publications and has served as reviewer in international journals and conferences. She has supervised several PhD theses and her present main research interest areas are numerical simulation and control applied to wave power generation plants, and fusion.

A.J. Garrido received the MSc degree in applied physics in 1999, the MSc degree in electronic engineering in 2001, and the PhD degree in control systems and automation in 2003, all of them from the University of the Basque Country. Since 2000 he has held several researches and teaching positions at the Automatic Control and Systems Engineering Department of the University of the Basque Country at Bilbao, Spain, where he is currently associate professor of systems and control engineering. He has more than 100 papers published in the main international conferences of the area, book chapters and JCR(ISI)-indexed journals, has served as reviewer in several international indexed journals and conferences, and has supervised several PhD thesis. His present main research interest areas are the applied control of dynamic systems, especially induction machines and its application to wave power, and nuclear fusion processes.

ASTRA–MATLAB sujungimas į visumą tokamakams valdyti

Goretti SEVILLANO, Izaskun GARRIDO, Aitor J. GARRIDO

Smarkiai išaugus įprastinių energijos šaltinių vartojimui svarbu naudoti ir alternatyviąsias energijos rūšis. Šiuo požiūriu intensyvinamas sprendimas ir įvairių valdymo problemų, egzistuojančių branduoliniuose reaktoriuose, tokiuose kaip tokamakai. Šio darbo tikslas parodyti, kaip ASTRA kodas, taikomas branduolinės reakcijos procesų vyksmui tokamakuose modeliuoti, integruojamas su Matlabo Simulinku, siekiant padaryti paprastesniu reguliatorių, tinkamų tokamakams valdyti, projektavimą. Rezultatai gauti diegiant modifikuotą PID reguliatorių, susietą su tam tikru optimizavimo algoritmu, patvirtina šio projekto tikslingumą ir jo taikymą, kuriant aukštesnio tipo valdymo schemas standartizuotam ASTRA kodui.