

Previous Experience & Proposal for Future Work

Gianmaria DE TOMMASI – CREATE/Università di Napoli Federico II – Italy

Email: detommas@unina.it

Thanks to Marcello Cinque, Sara Dubbioso, Federico Fiorenza and Daniele Ottaviano

Plasma Control ISFN Meeting - ITER HQ, 23 Sep 2025



DIPARTIMENTO DI INGEGNERIA ELETTRICA
E DELLE TECNOLOGIE DELL'INFORMAZIONE

My background

- 2001 MSc in Electronic Engineering in 2001 - University of Naples Federico II
- 2002-2005 PhD in Control Engineering - University of Naples Federico II (. . . and at JET with the PPCC group)
- 2002-. . . Member of the CREATE Consortium (research grants & contracts for JET, EAST, JT-60SA, DTT, . . .)
- 2009-. . . Various contracts with ITER (Central Safety System, Central Interlock System, Plasma Control System)



Topics

- 1 Development of real-time software framework for fusion devices
- 2 Design and deployment of plasma (magnetic) control systems
- 3 System-engineering approach for PCS design

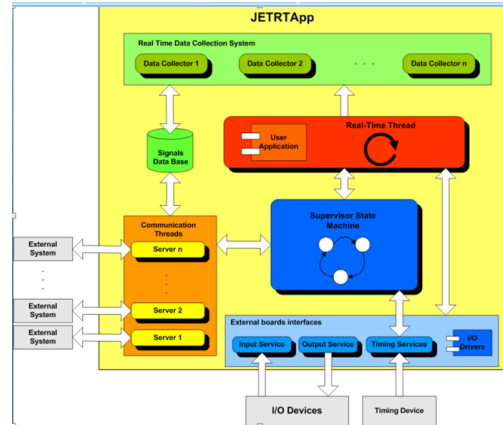
- 1 RT software frameworks – Past experience (very long ago)
- 2 RT Frameworks – Recent & possible future works
- 3 Plasma control – Past experience
- 4 Plasma control – Recent & possible future work
- 5 System-engineering: the ITER experience & possible future works

- The JETRT framework was developed in 2002/2003 to deploy the eXtreme Shape Controller (XSC)
- JETRT was based on the cross-platform BaseLib library (developed by the JET PPCC group)



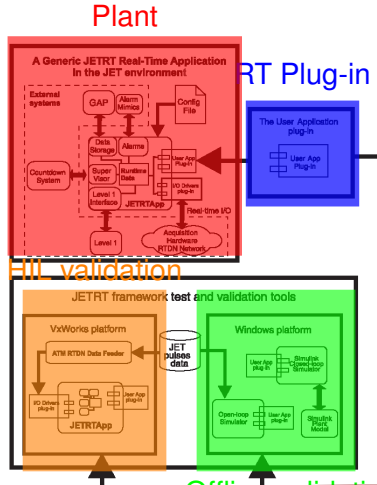
G. De Tommasi *et al.*

A flexible software for real-time control in nuclear fusion experiments
Contr. Eng. Prac., 2006



The **Real-time Application Plug-in** that can be used to:

- perform offline validation against a plant model
- perform real-time validation with hardware-in-the-loop
- run the real-time system on the plant



MARTe - The origins (2004-2005)

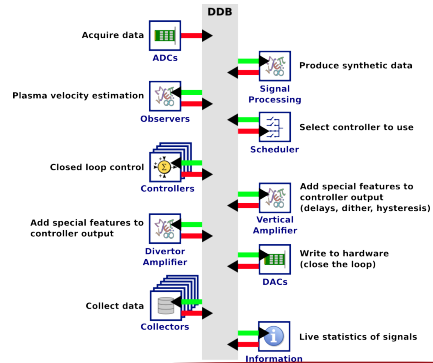


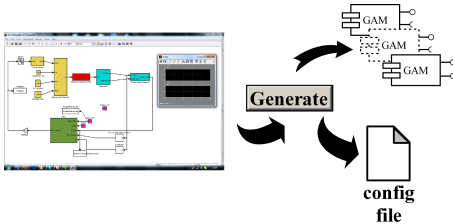
MARTe real-time thread for the JET VS5



A. C. Neto *et al.*

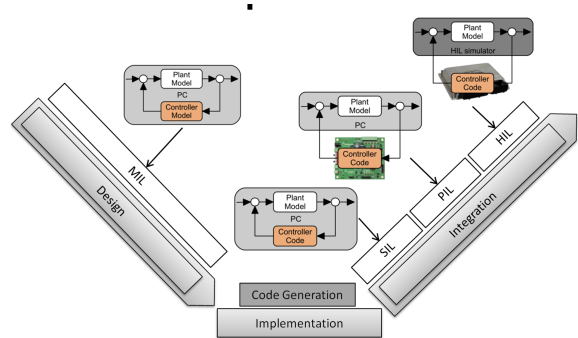
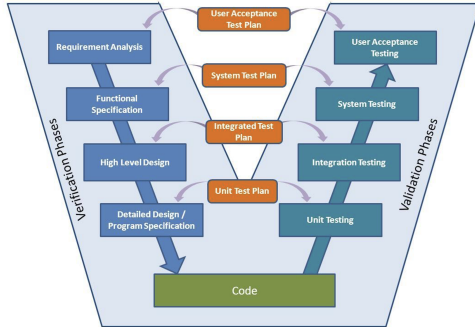
MARTe: A Multiplatform Real-Time Framework
IEEE Trans. Nucl. Sci., 2010



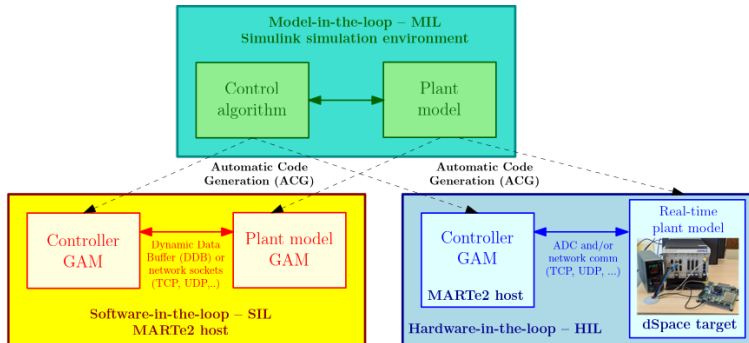


- Commonly used in many fields: rapid prototyping (dSpace, NI, Mathworks, . . .), automotive, railway, . . .
- Included in most of the RTFs for fusion devices (TCV RTF, ITER RTF, MARTe2+MDSplus)

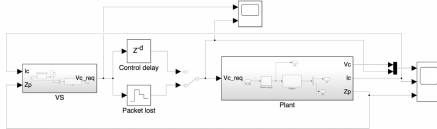
- 1 RT software frameworks – Past experience (very long ago)
- 2 RT Frameworks – Recent & possible future works
- 3 Plasma control – Past experience
- 4 Plasma control – Recent & possible future work
- 5 System-engineering: the ITER experience & possible future works



Setup for x-in-the-loop V&V



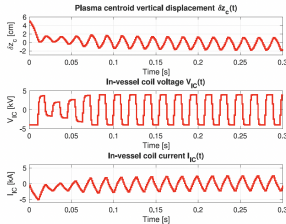
Example: the DTT VS system



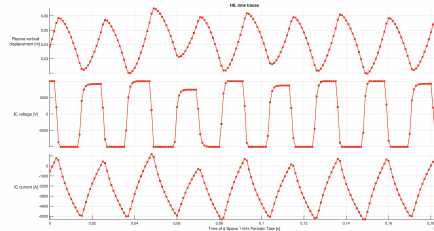
S. Dubbioso *et al.*

Rapid prototyping of control modules for the
DTT Plasma Control System

Fus. Eng. Des., 2025

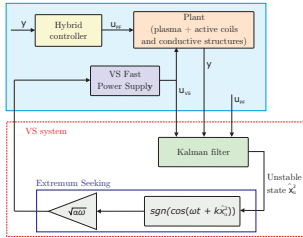


MIL simulation of a 5 cm VDE rejection with a systematic packet loss rate equal to $N_{\max}^{MIL} = 89\%$; transmission frequency is assumed equal to 1 kHz.

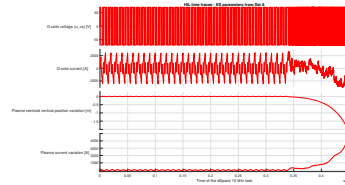


HIL simulation (SCALEXIO) of a 5 cm VDE in presence of a systematic packet loss set equal to $N_{\max}^{HIL} = 91\%$.

Example: a Extremum-seeking based VS



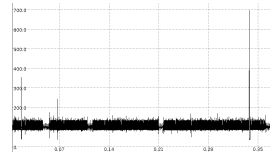
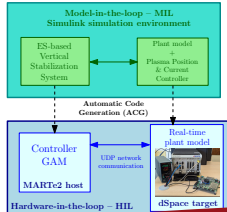
Functional architecture ES-based VS for TCV



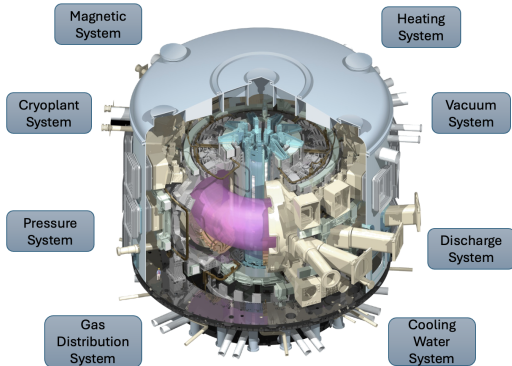
Time traces recorded during HIL simulation. **Vertical stability is lost when a delay of about 600 μ s is introduced in the control loop.**



S. Dubbioso et al.
Hardware-in-the-loop
validation of an
Extremum
Seeking-based system
for vertical stabilization
of tokamak plasmas
IFAC COSY Conf., 2025



VS thread execution time (measured in μ s) during the HIL simulation



- Can we use cloud technologies in the context of ITER to consolidate multiple isolated control systems on the same hardware?
- Possible objectives
 - Fast prototyping
 - Cost reduction
 - Heterogeneous spread redundancy
 - Diversified control & protection systems

■ Real-time containers

- Use the same abstraction everywhere
- Ensuring temporal isolation
 - Containers on co-kernels (RTAI, Xenomai)
 - Containers on partitioning hypervisors (Jailhouse)
 - Containers on MPSoCs (on RPU/FPGA via the Omnivisor)

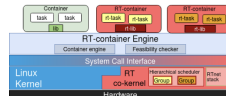
■ Memory and I/O isolation

- Reducing/controlling memory interference from co-accessing cores
- Real-time I/O access regulation

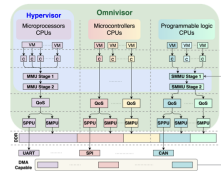
■ Criticality-aware orchestration

- Placement & monitoring driven by containers criticality and the isolation assured by nodes (**K4.0s**)
- Temporal isolation assessment
- Placement prioritization (**Preempt-K8s**)

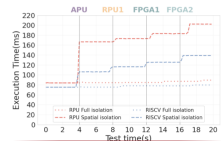
rt-containers on co-kernels



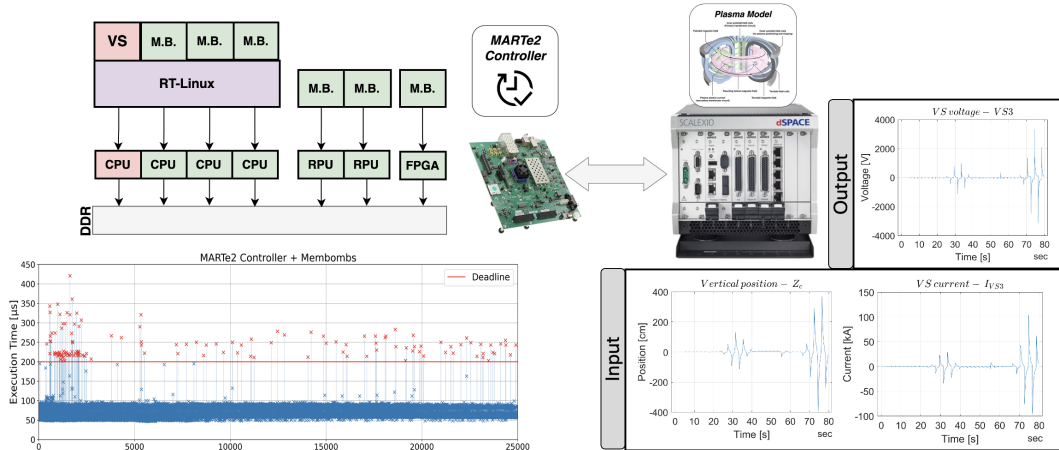
Omnivisor



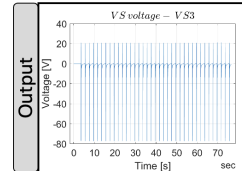
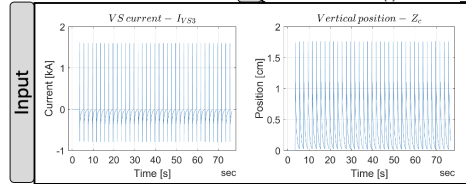
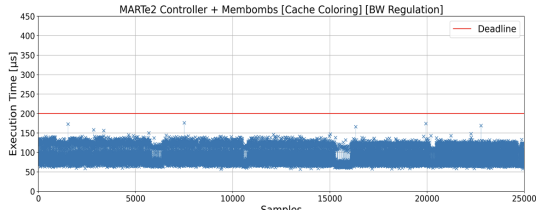
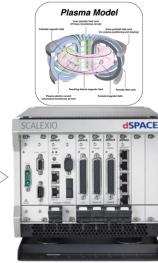
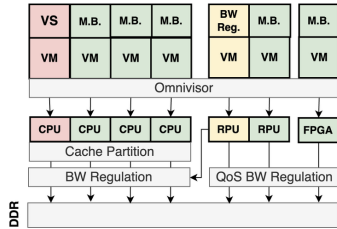
Isolation results on MPSoCs



HIL simulation of VS with co-located interference



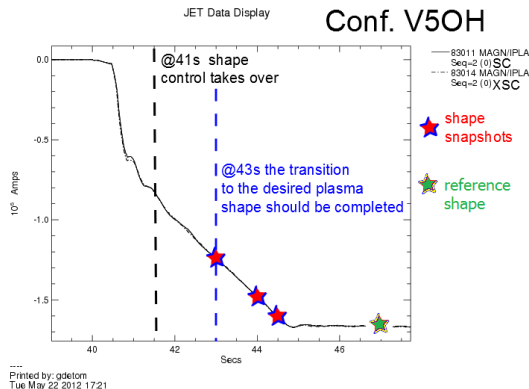
Isolation with `rt-container` and omnivisor



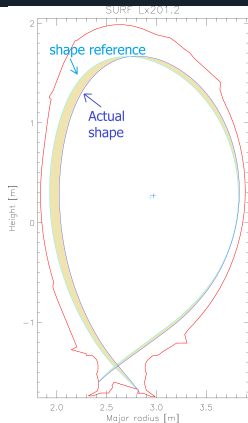
- 1 RT software frameworks – Past experience (very long ago)
- 2 RT Frameworks – Recent & possible future works
- 3 Plasma control – Past experience**
- 4 Plasma control – Recent & possible future work
- 5 System-engineering: the ITER experience & possible future works

- Design & implementation of the eXtreme Shape Controller (XSC)
- MIMO approach for plasma shape control
- It minimizes the steady-state MSE over a number of plasma shape descriptor that is greater than the number of actuators
- Exploits the SVD of a linear relationship between the PF currents and the controlled parameters
- Model-based approach that (partially) trades robustness for performance
- Originally implemented at JET but tested also on other machines (EAST, ...)

- #83011 – plasma ramp-up with standard JET SC
- #83014 – plasma ramp-up with XSC

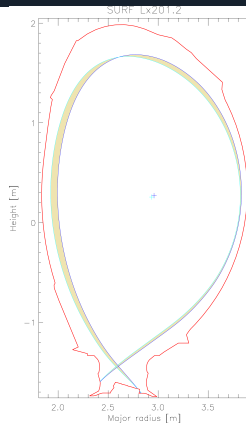


#83011 – Shape tracking during the ramp-up with SC



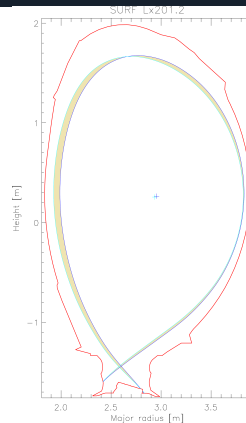
— #83011/JETPPF/EFIT/O t=43.013000
— #83011/JETPPF/EFIT/O t=47.010601

@43s



— #83011/JETPPF/EFIT/O t=44.000999
— #83011/JETPPF/EFIT/O t=47.010601

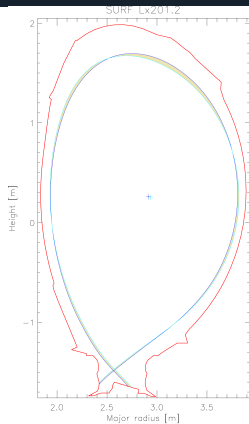
@44s



— #83011/JETPPF/EFIT/O t=44.502602
— #83011/JETPPF/EFIT/O t=47.010601

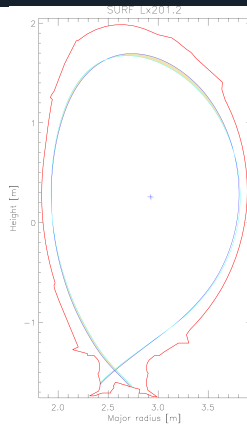
@44.5s

#83014 – Shape tracking during the ramp-up with XSC



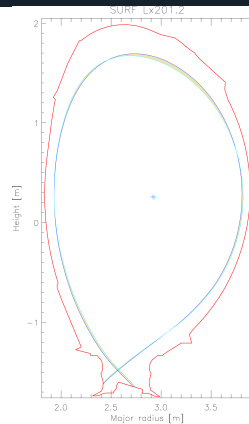
— #83014/JETPPF/EFIT/O t=43.013000
— #83014/JETPPF/EFIT/O t=47.010601

@43s



— #83014/JETPPF/EFIT/O t=44.000999
— #83014/JETPPF/EFIT/O t=47.010601

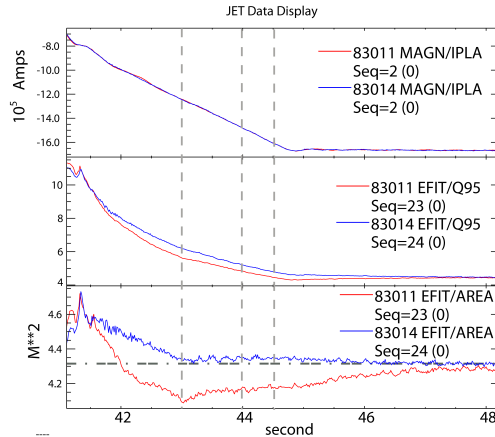
@44s



— #83014/JETPPF/EFIT/O t=44.502602
— #83014/JETPPF/EFIT/O t=47.010601

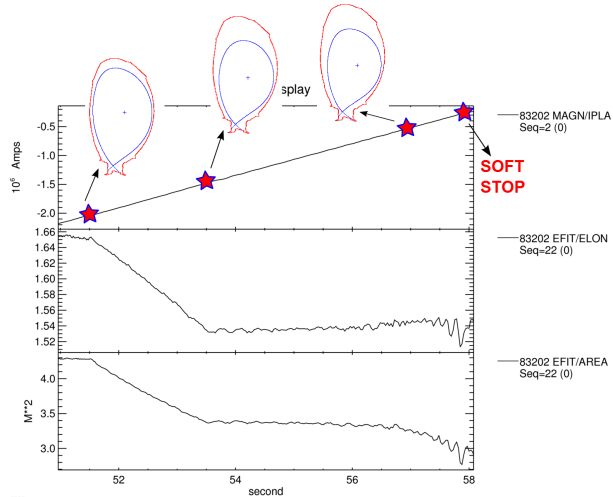
@44.5s

Plasma surface and q95



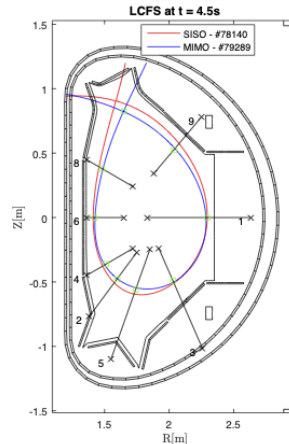
Printed by: fmavig
Wed May 23 2012 14:41

Change of elongation during the plasma current ramp-down

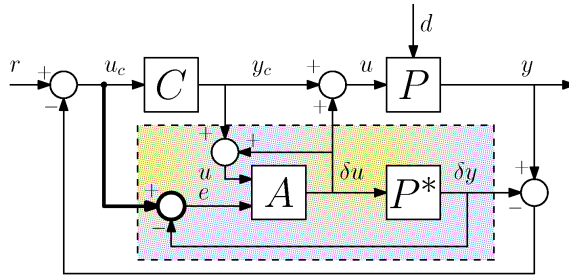


Printed by: gdetom
Wed Jun 20 2012 17:01

- Comparison between the SISO and MIMO shape controllers (pulses #78140 and #79289)
- The LCFS at $t = 4.5$ s is shown together with the control points and the target X-point position



Block diagram of the allocated closed-loop



$$P^* := \lim_{s \rightarrow 0} P(s),$$

The current allocator

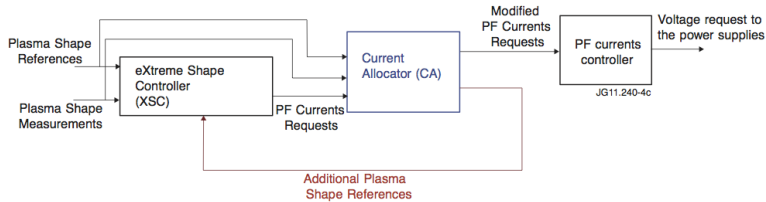
The allocator equations are given by

$$\dot{x}_a = -KB_0^T \begin{bmatrix} I \\ P^* \end{bmatrix}^T (\nabla J)^T \Big|_{(u, \delta y)}, \quad (1a)$$

$$\delta u = B_0 x_a, \quad (1b)$$

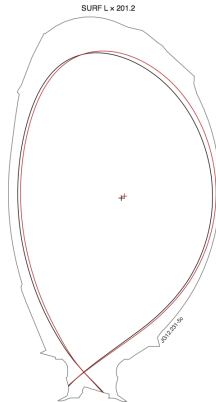
$$\delta y = P^* B_0 x_a. \quad (1c)$$

- $K \in \mathbb{R}^{n_a \times n_a}$ is a symmetric positive definite matrix used to specify the allocator convergence speed, and to distribute the allocation effort in the different directions
- $J(u^*, \delta y^*)$ is a continuously differentiable cost function that measures the trade-off between the current saturations and the control error (on the plasma shape)
- $B_0 \in \mathbb{R}^{n_{PF} \times n_a}$ is a suitable full column rank matrix

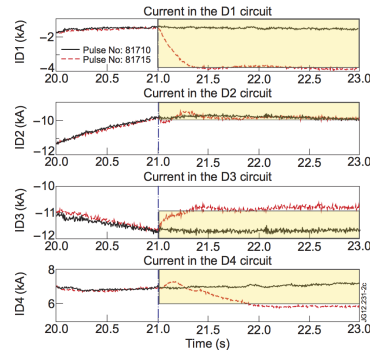


The CLA block is inserted between the *XSC* and the *Current Decoupling Controller*

Experimental results of CLA @ JET

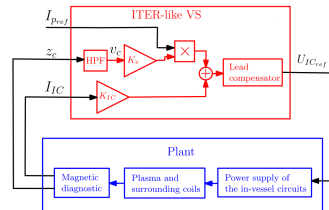
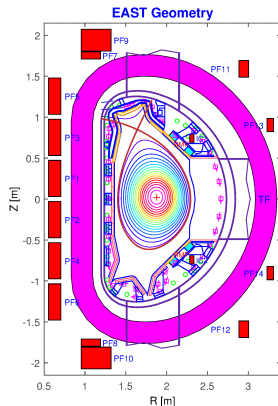


Shape comparison at 22.5 s. Black shape
(#81710 without CLA), red shape (#81715
with CLA)

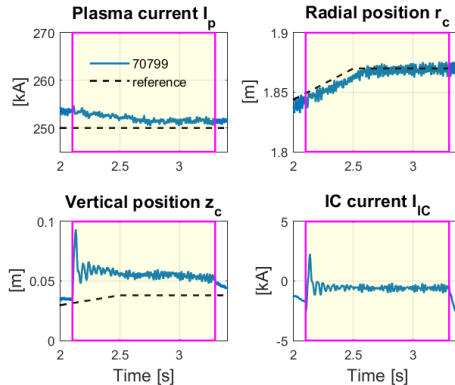


Currents in the divertor circuits. #81710 (reference pulse
without CLA) and pulse #81715 (with CLA). The shared
areas correspond to regions beyond the current limits
enforced by the CLA parameters.

ITER-like VS for the EAST tokamak

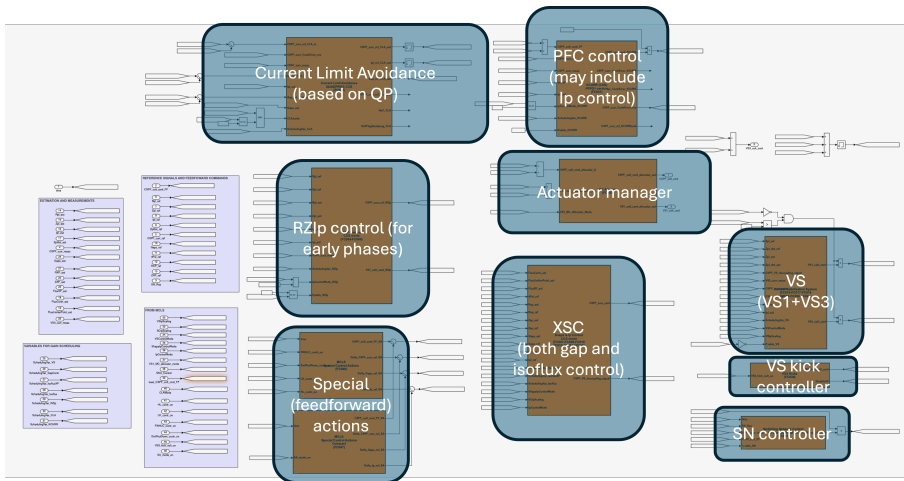


$$U_{ICref}(s) = \frac{1 + s\tau_1}{1 + s\tau_2} \cdot \left(K_v \cdot \bar{I}_{pref} \cdot \frac{s}{1 + s\tau_z} \cdot Z_c(s) + K_{IC} \cdot I_{IC}(s) \right)$$



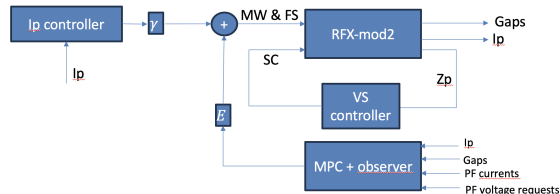
EAST pulse #70799. During this pulse the *ITER-like* VS was enabled from $t = 2.1$ s for 1.2 s, and only I_p and r_c were controlled, while z_c was left uncontrolled. This first test confirmed that the *ITER-like* VS vertically stabilized the plasma by controlling $\dot{z}_c$ and I_{IC} , without the need to feed back the vertical position z_c

The ITER Magnetic Control Architecture (for PFPO-1/SRO)



- 1 RT software frameworks – Past experience (very long ago)
- 2 RT Frameworks – Recent & possible future works
- 3 Plasma control – Past experience
- 4 Plasma control – Recent & possible future work**
- 5 System-engineering: the ITER experience & possible future works

- The VS controller stabilizes the plasma with the dedicated Saddle Coils
- I_p controller and the MPC for the shape control act on Magnetizing Windings (MW) and Field Shaping (FS) coils (actuator sharing $\rightarrow$ model-based geometric decoupling)
 - The I_p controller is a PID that acts on a control direction γ that does minimize the effect on the plasma shape
 - The MPC for the shape control acts along the *subspace* E , where $\gamma^T E = 0$
 - MPC used to not violate voltage limits



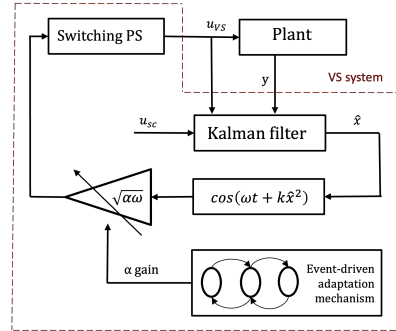
Model-free control with Extremum Seeking



- Control approach based on the minimization (via Extremum Seeking) of a candidate Lyapunov function $V(x)$

$$u(t) = \omega \cos(\omega t + kV(x))$$

- A Kalman filter is needed to estimate the plasma movement along the unstable mode $\hat{x}$
- Robustness against such Kalman filter has been observed in simulation
- Kalman filter can be replaced by a NN (trained against simulation data)
- Plan to test it on TCV (still WIP)

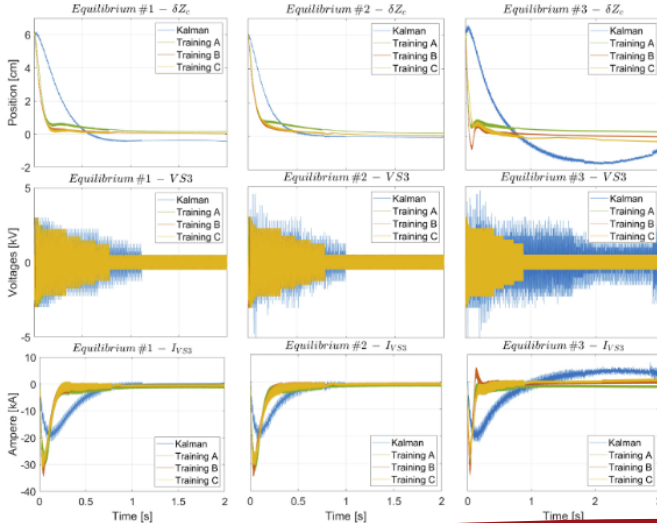


S. Dubbioso *et al.*

Model-free Stabilization via Extremum Seeking
using a cost neural estimator

Expert Sys. Appl., 2024

ITER benchmark



A RL-BASED VERTICAL STABILIZATION SYSTEM FOR THE EAST TOKAMAK

G. De Tommasi^{1,2}, S. Dubbioso^{1,2}, Y. Huang¹, Z. P. Luo¹, A. Mele¹, B. J. Xiao³

¹ Dipartimento di Ingegneria Elettrica e delle Tecnologie dell'Informazione, Università degli Studi di Napoli Federico II, via Claudio 21, 80125, Napoli, Italy

² Consorzio CREATE, via Claudio 21, 80125, Napoli, Italy

³ Dipartimento di Economia, Ingegneria, Società e Impresa, Università degli Studi della Tuscia, Viterbo, Italy

⁴ Institute of Plasma Physics, Chinese Academy of Sciences, Hefei 230021, People's Republic of China



Nuclear fusion & Tokamaks

- Nuclear fusion is foreseen as a promising source of clean and sustainable energy for the next century
- Tokamaks are experimental devices aimed at producing energy from nuclear fusion reactions that occur in a magnetically confined hot plasma
- EAST is a superconductive tokamak located in Hefei, People's Republic of China



Plasma Magnetic Control & Vertical Stabilization



- Plasma magnetic control aims at controlling the current, position and shape of the plasma column inside the vacuum vessel by means of external magnetic fields generated by the so called Poloidal Field coils (PFC)
- High performance plasmas, as the ones achieved at the EAST tokamak, have elongated poloidal cross-section which turn to be vertically unstable (like a ball on the top of a hill)
- A Vertical Stabilization (VS) system is needed to run any modern tokamak



Reinforcement Learning approach for the EAST VS



Main idea: train an agent by making it interact with a state-space linearized model of the EAST plasma and surrounding coils dynamics (RL environment), so that the agent learns how to solve the VS problem

$$\dot{A}(t) = A(t)x + B(t)u$$

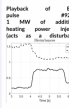
$$\dot{y}(t) = C(t)x + D(t)u$$

$$y(t) = C(t)x + D(t)u$$

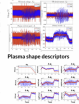
The offline training based on control of random Vertical Displacement events (VDE) in the range ± 5 cm permit to retrieve a static input-output table for the VS control system, which represents the VS control strategy

The RL-based VS agent can then be included in the whole magnetic control architecture and implement the policy to select the voltage request to the in-vessel coils V_{in} based on observation coming from the plant

Simulation results & Conclusions



Agent I/O and Plasma current



Vertical displacement events

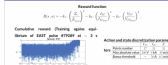


Agent I/O and Plasma current



- The preliminary assessment of the data-driven VS for the EAST proved to be robust with respect to models and scenarios not used during the training
- The robustness can be increased by considering a set of different equilibria (different plasma internal profiles, triangularity, elongation, etc.)
- however a non-tabular approach such as Deep Deterministic Policy Gradient (DDPG) must be pursued in order to contain the computational burden

Q-learning for the training of the VS agent



State: $x = [z, \dot{z}, \ddot{z}]$ and actions $u = V_{in}$ discretized space

Plasma state values: $z, \dot{z}, \ddot{z}$

State augmentation: $z, \dot{z}, \ddot{z}$

Actions:

discretized Action: $u = V_{in}$

discretization parameter: $\Delta V_{in} = 1$

along horizon: $N = 10$

Specify reward function: $R_t = -\dot{z}_t$ according to [3]

Initialize: $Q(s, a) = 0$ for all state-action pairs

Initialize: $\epsilon = 0.1$ with a random seed in the range $[-1, 1]$ cm

Initialize: $\gamma = 0.99$

Repeat: $n = 1$ to N

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table

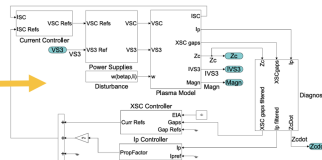
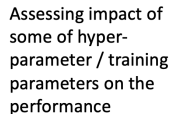
Observe: $z_t, \dot{z}_t, \ddot{z}_t$ according to the green policy applied on current Q table



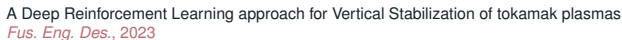
G. De Tommasi *et al.*

A RL-based Vertical Stabilization System for the EAST tokamak

American Control Conf., 2022



Hyper-parameter	Considered values	
Sampling time T_s	2.5 ms	
Episode duration T	5 s	20 s
Actor learning rate	5×10^{-4}	
Critic learning rate	10^{-3}	
Actor hidden layers $\#m$	64	128
Critic hidden layers $\#n$	32	128
Discount factor γ	0.99	
Batch size	256	
OUP variance	1840	
OUP variance decay rate	8.66×10^{-6}	3.5×10^{-7}



- Time & resource consuming activity
- A lot of trial & error
- Need to setup joint activities with big player
 - EPFL/SPC & DeepMind
 - CEA & Capgemini
- Are AI controller really outperforming model-based (adaptive control)?
- Are tokamak discharge so *unpredictable* (once the scenario is assessed)?
 - In the past we trade robustness to some degree to achieve tight control
- What if the AI controller does not work? How to re-tune it (in due time)?
- **AI & ML can be beneficial in many other ways. . .**

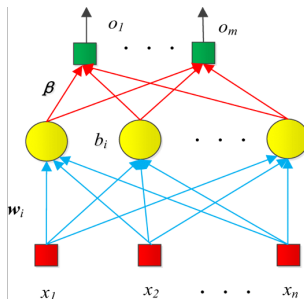
- Physics Informed Neural Networks (PINNs) can be used to solve Partial Differential Equations with initial/boundary conditions.
- PINNs may require long training times
- Given the magnetic measurements m_ψ and m_B , by solving $\Delta^* \psi(r, z) = 0$ it is possible to find the LCFS
- This equation can be solved in real-time by Physics-Informed Extreme Learning Machines (PIELMs)

- ELMs are similar to one-layer feedforward NNs and proved to have great generalization capabilities
- The structure of an ELM is the following

$$\psi_{ELM}(x) = \beta^T \sigma(Wx + b)$$

being

- β is a vector of unknown weights
- $\sigma(\cdot)$ is the activation function
- W and b are **randomly initialised**



Given a set of collocation points on the poloidal plane $\{x_i\}_{i=1, \dots, N}$, the ELM shall satisfy the GS equation at those points

$$\Delta^* \psi_{ELM}(x_i) = \beta^T \Delta^* \sigma(Wx + b) = 0$$

which is equivalent to

$$\beta^T D_\psi = 0$$

being D_ψ a matrix that can be computed [offline](#).

By exploiting similar argument the ELM can be linked to the poloidal magnetic field measures

$$\beta^T D_B = m_B$$

- PIELM can be trained to satisfy the GS equation and magnetic measurements by minimizing the cost function

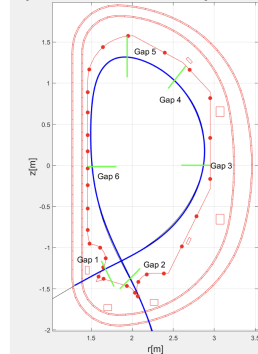
$$Loss(\beta) = \|\beta^T D_\psi\| + \|\beta^T \sigma(Wx_\psi + b) - m_\psi\| + \|\beta^T D_B - m_B\|$$

- Hence, standard least-squares methods are sufficient to train the ELM weights (pseudo-inverse) $\Rightarrow$ PIELM can be trained in real-time!
- If $\beta = E\tilde{\beta}$, with $D_\psi^T E = 0$, then it is always $\|\beta^T D_\psi\| = 0 \Rightarrow$ the PIELM satisfies the GS equation by design and the weights are trained only to fit the measures

- This method can train the ELM weights every time a new set of measurements are available
- Once trained, the ELM is not constrained by a grid of points
- Other real-time methods (XLOC, CCS, filamentary reconstruction) need to either identify the boundary locally in several regions or first identify the sources and then put together the contributions

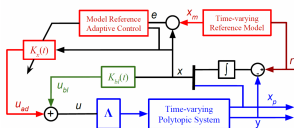
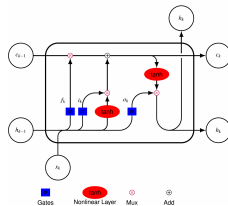
Gap	Error [cm]
1	0.1
2	0.45
3	0.48
4	0.69
5	0.11
6	0.03

Physics-Informed ELMs boundary reconstruction



The red dots represent where the solution of the GS equation is set as hard constraint. The blue line is the reconstructed one. The black comes from CREATE-NL

- Design an LSTM estimator for eddy-currents, instead Kalman filters based on linear models
- Design of adaptive and robust magnetic control components, such as a Model-Reference Adaptive Control (MRAC) for Vertical Stabilization
- Surrogate data-driven model to speed up nonlinear simulations
- Data-driven model for gyrotrons control (Andrea Antonione ECRH group)
- ...



- 1 RT software frameworks – Past experience (very long ago)
- 2 RT Frameworks – Recent & possible future works
- 3 Plasma control – Past experience
- 4 Plasma control – Recent & possible future work
- 5 System-engineering: the ITER experience & possible future works



Strategy to systematically design and deploy the ITER plasma control system: A system engineering and model-based design approach
Fus. Eng. Des., 2024

What's goin' on. . .



J. J. Norheim *et al.*

Challenges in applying large language models to requirements engineering tasks
Design Science, 2024

■ Requirements Engineering

- Assist in drafting, clarifying, and *validating* system requirements (i.e., make them *testable*)
- Detect ambiguities, inconsistencies (e.g., between SHRs and SYRs), or missing elements in requirement documents

■ Model-Based Systems Engineering

- Translate natural-language requirements into SysML/UML elements
- Generate system diagrams or design artifacts from textual descriptions



■ System Architecture & Design Support

- Propose alternative architectures based on constraints and objectives

■ Simulation & Testing

- Generate test cases/scenarios from requirements
- Automate setup assessments (in PCSSP, at least for *atomic* ones)

■ Risk & Safety Analysis

- Help identify potential hazards or failure modes from design documents

Previous Experience & Proposal for Future Work

Gianmaria DE TOMMASI – CREATE/Università di Napoli Federico II – Italy

Email: detommas@unina.it

Thanks to Marcello Cinque, Sara Dubbioso, Federico Fiorenza and Daniele Ottaviano

Plasma Control ISFN Meeting - ITER HQ, 23 Sep 2025



UNIVERSITA' DEGLI STUDI DI
NAPOLI FEDERICO II

DIPARTIMENTO DI INGEGNERIA ELETTRICA
E DELLE TECNOLOGIE DELL'INFORMAZIONE