

PAPER • OPEN ACCESS

The FreeGSNKE pulse design tool (FPDT): a computational framework for evolutive plasma scenario and control design

To cite this article: K Pentland *et al* 2026 *Plasma Phys. Control. Fusion* **68** 075035

View the [article online](#) for updates and enhancements.

You may also like

- [Tokamak discharge simulation coupling free-boundary equilibrium and plasma model with application to JT-60SA](#)
V. Ostuni, J.F. Artaud, G. Giruzzi et al.
- [Full radius integrated modelling of ohmic ramp-up at TCV including self consistent density prediction](#)
M. Marin, Y. Camenen, C. Bourdelle et al.
- [Model predictive control of KSTAR equilibrium parameters enabled by TRANSP](#)
M.D. Boyer, X. Yuan, J. Ahn et al.

Plasma Physics and Controlled Fusion



PAPER

OPEN ACCESS

RECEIVED
31 March 2026

REVISED
29 June 2026

ACCEPTED FOR PUBLICATION
15 July 2026

PUBLISHED
24 July 2026

Original content from
this work may be used
under the terms of the
[Creative Commons
Attribution 4.0 licence](#).

Any further distribution
of this work must
maintain attribution to
the author(s) and the title
of the work, journal
citation and DOI.



The FreeGSNKE pulse design tool (FPDT): a computational framework for evolutive plasma scenario and control design

K Pentland^{1,*} , N C Amorisco¹ , A Ross² , P Cavestany², T Nunn¹ , A Agnello² , G K Holt¹ ,
G McArdle¹ , C Vincent¹ , J Buchanan¹ and S J P Pamela¹

¹ United Kingdom Atomic Energy Authority, Culham Campus, Abingdon, Oxfordshire OX14 3DB, United Kingdom

² STFC Hartree Centre, Sci-Tech Daresbury, Keckwick Lane, Daresbury, Warrington WA4 4AD, United Kingdom

* Author to whom any correspondence should be addressed.

E-mail: kamran.pentland@ukaea.uk

Keywords: pulse design tool, plasma control system, Grad–Shafranov, MHD equilibria, FreeGSNKE, MAST-U

Abstract

We present the FreeGSNKE pulse design tool (FPDT), an open-source, Python-based computational framework that enables *in silico* testing and predictive design of tokamak plasma scenarios and control strategies. The FPDT couples the FreeGSNKE evolutive equilibrium solver with a virtual plasma control system (PCS) containing modular and customisable controllers. Given a set of user-defined waveforms and control parameters, the virtual PCS uses feedback and feedforward control to modulate plasma current, position, and shape, while adhering to machine safety limits on poloidal field (PF) coil currents and voltages. The resulting framework allows simulation of the controlled dynamic evolution of plasma equilibria, along with the currents in both active PF coils and passive conducting structures, under the assumption of axisymmetry. The FPDT can be used to develop plasma scenarios, test control schemes, calibrate control parameters, and perform uncertainty quantification studies, thereby reducing iterative and expensive experimental testing on a physical tokamak. The FPDT is machine-agnostic and can be customised to implement different control algorithms tailored to the specific tokamak of interest. Here, we outline the overall framework and validate its performance on plasma discharges on the MAST Upgrade tokamak in the ‘flat-top’ phase. We demonstrate excellent quantitative agreement between the FPDT simulations, the desired control waveforms, and the experimental shot data. With this extension to the FreeGSNKE open-source suite of codes we aim to encourage more reproducible and collaborative research in plasma modeling and control.

1. Introduction

In magnetic confinement fusion, plant operators must be able to rely on robust control algorithms and machine protection systems in order to safely operate, sustain, and reproduce high-performance tokamak plasma scenarios (e.g. Ariola and Pironti 2016). To do this, all tokamak devices are equipped with a plasma control system (PCS), a combination of software and hardware that receive measurements from diagnostics around the machine in order to actively control both the plant and the plasma using a number of different actuators (see, e.g. Penaflor *et al* 2008). At a minimum, the role of the PCS is to maintain the desired plasma position, shape, and current throughout all phases of a plasma ‘pulse’³—from coil magnetisation and current ramp-up, through flat-top operation, to ramp-down and termination. Beyond magnetic control, the PCS can also be responsible for coordinating systems that regulate plasma heating (e.g. via neutral beams), manage heat loads in the divertor, maintain fuel levels through gas puffing, and monitoring impurities and radiation. Some example PCSs include those at DIII-D (Penaflor *et al* 2013), TCV (Galperti *et al* 2024), and MAST-U (McArdle *et al* 2020, McArdle 2023).

³ Sometimes referred to as a ‘discharge’ or ‘shot’.

To gain deeper insight into the interaction between control actuators and plasma dynamics without physically running experiments, plasma experimentalists are increasingly turning to pulse design tools (PDTs). A PDT is a computational framework for simulating the time-dependent evolution of a tokamak plasma subject to the actions of a (virtual) PCS. As such, PDTs enable *in silico* control design and validation. Access to a PDT can support researchers in planning tokamak experiments by enabling simulations of challenging new scenarios and control algorithms, dramatically reducing the need for direct and potentially error-prone testing on the tokamak. PDTs can be used to test if a given discharge might violate machine safety limits (e.g. coil currents and voltages), as well as calibrate and optimise any parameters defining the control algorithms (e.g. gains, virtual circuits (VCs), damping, etc) across different operating scenarios and before deployment on the real PCS. They can support uncertainty quantification and its impact on PCS performance, for example, by simulating how noise and disturbances affect measurement quantities from various diagnostics (whether synthetic or not). PDTs can also be used for control room training and, if directly integrated with a PCS, can enable automated pulse testing and execution—reducing the chance of human error in the experimental pipeline.

In this paper, we focus on the magnetic control of the plasma position, shape, and current, via their actuators—the poloidal field (PF) coils. The control actions consist of active voltages applied by the power supplies to the PF coils. These voltages generate currents, which in turn produce magnetic fields that confine, shape, and stabilise both the core plasma (keeping it away from the vessel walls) and the divertor legs (managing exhaust heat loads on the divertor tiles). Using diagnostic measurements, the PCS continually adjusts these voltages in real-time (on sub- ms timescales) such that user-prescribed waveforms for plasma current, shape, and position are followed, thus maintaining a stable operating scenario. In the context of magnetic control, PDTs typically include two main components: (i) a fast⁴, accurate evolutive simulator of the coupled plasma, PF coil, and vessel currents, and (ii) a virtual PCS composed of several interlinked control algorithms. A given PCS can employ a wide range of different control algorithms, including proportional–integral–derivative (PID) controllers, model predictive control (Gerksič and De Tommasi 2016, Mele *et al* 2025), and reinforcement-learning agents (Degraeve *et al* 2022, Kerboua-Benlarbi *et al* 2024, Tracey *et al* 2024).

A vast array of evolutive free-boundary equilibrium simulators have been presented in the literature, varying in programming language, modeling assumptions, implementation choices, computational cost, open accessibility, and ease of use. An incomplete list includes: FreeGSNKE (Amorisco *et al* 2024), NICE (Gros *et al* 2025), CREATE-NL (Albanese *et al* 2015), GSevolve (Welandar *et al* 2019), DINA (Khayrutdinov and Lukash 1993), SPIDER (Ivanov *et al* 2009), CEDRES++ (Heumann *et al* 2015), FGE (Carpanese 2021, Heiß *et al* 2025), and NSFsim (Clark *et al* 2025). Similarly, a number of PDTs have been developed and successfully used to support experiments on several tokamaks. A few notable examples are: XSC Tools, used routinely on JET (Albanese *et al* 2005, De Tommasi *et al* 2007); TokSys, used on DIII-D (Humphreys *et al* 2008), MAST-U (Anand *et al* 2024, Lvovskiy *et al* 2025), and NSTX-U (Pajares *et al* 2025); MEQ, used on TCV (Carpanese 2021); Fenix, deployed on ASDEX-U (Janky *et al* 2021, Fable *et al* 2022); the PCS Simulation Platform to be deployed on ITER (Walker *et al* 2015); SOPHIA, used on ST40 (Janky *et al* 2026); and NICE, applied to WEST (Gros *et al* 2025, Nouailletas *et al* 2025). To the best of our knowledge, NICE⁵ and MEQ⁶ are the only open-source PDTs currently available to the community (with the caveat that the MATLAB code requires a licence).

These tools vary widely in their capabilities, from utilising physics models of varying complexity to offering advanced graphical or web-based user interfaces. Underlying physics models range from equilibrium-only descriptions (e.g. XSC Tools, TokSys, MEQ) to those that include current diffusion and transport models (e.g. Fenix, SOPHIA, NICE). Some tools are designed to reflect the physical setup, control, and diagnostic capabilities of a specific tokamak, while others can be suitably modified for use on different machines. In addition, certain tools employ linear or piecewise-linear model equations to reduce computational runtime and can interface directly with a physical PCS on particular tokamaks (e.g. XSC Tools, TokSys). Most PDTs focus on predictive simulations, however, inverse evolutive equilibrium tools for scenario design, such as CEDRES++ (Heumann *et al* 2015), FEEQS (Blum *et al* 2019), and GSPulse (Wai *et al* 2025), have also been proposed. We also note that integrated modeling tools—which couple physics, engineering, and control models to design and optimise entire fusion devices rather than

⁴ Relative to the time between plasma discharges.

⁵ <https://gitlab.inria.fr/blfauger/nice>.

⁶ <https://gitlab.epfl.ch/spc/public/meq>.

individual plasma pulses—do exist, but they are not designed for ‘virtual PCS’ shot design and validation simulations. Examples include PROCESS (Muldrew *et al* 2020), Bluemira (Coleman *et al* 2025), METIS (Artaud *et al* 2018), JINTRAC (Romanelli *et al* 2014), and FUSE (Meneghini *et al* 2024).

For a PDT to be useful, fundamental properties must include numerical fidelity, speed, and ease of customisation—all of which can make or break its more widespread adoption by the modeling community. For instance, ensuring that the PDT returns accurate re-simulations of past discharges (assuming it is provided with the corresponding PCS settings) is fundamental when looking to use the PDT to explore new scenarios or modifications on the default control algorithms. Computational speed is clearly advantageous, but this will vary depending on the level of physics captured by the simulator and the intended use cases, with control room use for rapid testing between plasma discharges likely being the most demanding. In order to simplify design and optimisation, control parameters and waveforms should be easily accessible and editable by users, while control algorithms should also be interchangeable and easily modifiable to accommodate different machines.

With this in mind, we present here the FreeGSNKE PDT (FPDT), a Python-based PDT built on the evolutive equilibrium code FreeGSNKE (Amorisco *et al* 2024), and released as part of the FreeGSNKE open-source suite⁷. The FPDT aims to enable *in silico* predictive design of experimental plasma scenarios and control strategies. To do this, we couple FreeGSNKE’s existing evolutive equilibrium solver with a new virtual PCS class that contains modular, documented, and customisable algorithms, each responsible for enforcing different aspects of magnetic plasma control. This extends FreeGSNKE’s previous capability of purely feedforward (FF) equilibrium simulations of plasma pulses that use time-dependent PF coil voltages recorded from, for example, past discharges or other sources. The FPDT now enables fully flexible control of plasma current, position, and shape, in which voltage actuators are driven by a combination of both FF and feedback (FB) controllers. It has a modular and transparent design, which supports implementation of different internal control algorithms tailored to specific tokamak configurations, and could potentially serve as the foundation of a more comprehensive ‘flight simulator’ tool for plasma modeling.

One of the motivations for developing this tool is to provide the community with an open-source PDT built on a contemporary software stack. The FPDT is free and open-source, with comprehensive documentation and fully accessible source code that enables scrutiny and community contributions. We hope the FPDT will enable more reproducible and collaborative research in plasma modeling and control studies (Buxton 2025, Farmakalides *et al* 2025, Kuang *et al* 2026, Pentland *et al* 2026), particularly when developing modern control approaches based on machine learning, with which FreeGSNKE integrates seamlessly (Sarkar *et al* 2026).

While the FPDT is inherently machine agnostic, the controllers we implement in the virtual PCS shown here take inspiration from those in the MAST-U PCS (McArdle *et al* 2020, McArdle 2023), which have been successfully used to demonstrate high-performance scenarios with highly shaped plasmas, advanced divertor configurations, and heat flux mitigation (Morris *et al* 2018, Kool *et al* 2025). These include PID-based algorithms to control the plasma current, vertical plasma position, and both local (e.g. midplane radii, gaps, X-point positions) and global (e.g. elongation, triangularity, squareness) shape parameters and are accompanied by algorithms to handle PF coil activation times and safety constraints (in the form of current and voltage limits). We note that we do not include actuators and subsystems required to regulate plasma heating, fueling, impurities, radiation, and detachment.

Throughout the rest of the paper, we demonstrate how the FPDT satisfies the aforementioned criteria for an effective PDT and more specifically we

- (i) describe the FPDT and the coupling between FreeGSNKE’s evolutive equilibrium solver and the new PCS class (section 2.1).
- (ii) detail the governing equations of the simulator (section 2.2) and the implementation of the control algorithms within the PCS class, linked to MAST-U (section 2.3).
- (iii) validate the FPDT simulations against challenging plasma scenarios from previously executed MAST-U pulses (section 3).

We discuss the implications of our work in section 4 and conclude by outlining the future of the FPDT.

⁷ <https://github.com/FusionComputingLab/freegsnke>.

2. The FPDT

2.1. Overview

At a high level (see the schematic in figure 1), the FPDT simulates the dynamic evolution of plasma equilibria and the associated currents in both the active PF coils and passive conducting structures using voltages generated by a virtual PCS class. In addition to the tokamak description and a set of initial conditions (i.e. a starting equilibrium), the FPDT requires user-defined reference *waveforms* for key plasma parameters (e.g. plasma current, vertical position) and controller settings that will govern the desired scenario.

During an FPDT simulation, FreeGSNKE's evolutive solver and the virtual PCS class interact during each time stepping cycle. At time t , the FreeGSNKE plasma state (see section 2.2.2) is queried and provides the virtual PCS with simulated measurements of the total plasma current, PF coil currents, and a number of plasma shape parameters. Based on the user-defined control settings and reference waveforms, these measurements are used to calculate the voltages that should be applied to the PF coils. Using these voltages and some prescribed internal plasma properties, FreeGSNKE evolves the plasma state from time t to $t + dt$, at which point the PCS is once again provided updated measurements, starting a new cycle. This closed loop is repeated until the desired end of the simulation, with users being able to extract any equilibrium-related quantities of interest at each time step.

2.2. The FreeGSNKE simulator

Here, we provide a brief overview of FreeGSNKE, its governing equations, and how it works, focusing on aspects directly relevant to the FPDT—for a more complete description, we refer the reader to Amorisco *et al* (2024).

2.2.1. Governing equations

The governing equations in FreeGSNKE describe the evolution of the plasma state together with the currents flowing in any conducting metal structures surrounding the tokamak. This is achieved by coupling the nonlinear Grad–Shafranov (GS) equation (Grad and Rubin 1958, Shafranov 1958) with a set of time-dependent circuit equations (CEs) that relate the currents in the conducting metals to the currents in the plasma itself.

Firstly, the CE governing the flow of current in the PF coils and any conducting passive structures at time t is given by

$$\mathcal{M}_{m,m} \frac{d}{dt} \mathbf{I}_m + \mathcal{M}_{m,y} \frac{d}{dt} \mathbf{I}_y + \mathcal{R}_{m,m} \mathbf{I}_m = \mathbf{V}_m, \quad (1)$$

where

- $\mathbf{I}_m(t) = (\mathbf{I}_{\text{act}}(t), \mathbf{I}_{\text{pas}}(t))$ is the vector of metal currents, including both active PF coils⁸ and passive conducting structures.
- $\mathbf{I}_y(t)$ is the vector describing the discretised current distribution of the plasma on the solution domain, which (in FreeGSNKE) is restricted to the region inside the plasma limiter (i.e. the first wall).
- $\mathbf{V}_m(t) = (\mathbf{V}_{\text{act}}(t), \mathbf{0})$ is the vector of voltages applied to the metal elements, comprising the active PF coils (whose voltages serve as the control actions supplied by the virtual PCS) and the passive structures (which receive no active voltage).
- $\mathcal{M}_{m,m}$ and $\mathcal{R}_{m,m}$ denote the mutual inductance and (diagonal) resistance matrix relevant to the metals, respectively. $\mathcal{M}_{m,y}$ denotes the inductance matrix of all metals on the discretised plasma domain⁹.

Secondly, to describe the plasma, FreeGSNKE uses a ‘lumped’ one-dimensional equation given by

$$\frac{\mathbf{I}_y^\top}{I_p} \cdot \left[\mathcal{M}_{y,y} \frac{d}{dt} \mathbf{I}_y + \mathcal{M}_{m,y}^\top \frac{d}{dt} \mathbf{I}_m + \mathcal{R}_{y,y} \mathbf{I}_y \right] = 0, \quad (2)$$

⁸ In this work, active PF coils refers to all actively controlled coils, including Ohmic coils, shape control coils, and vertical control coils.

⁹ We use bold symbols to represent vectors and calligraphic uppercase letters to denote matrices.

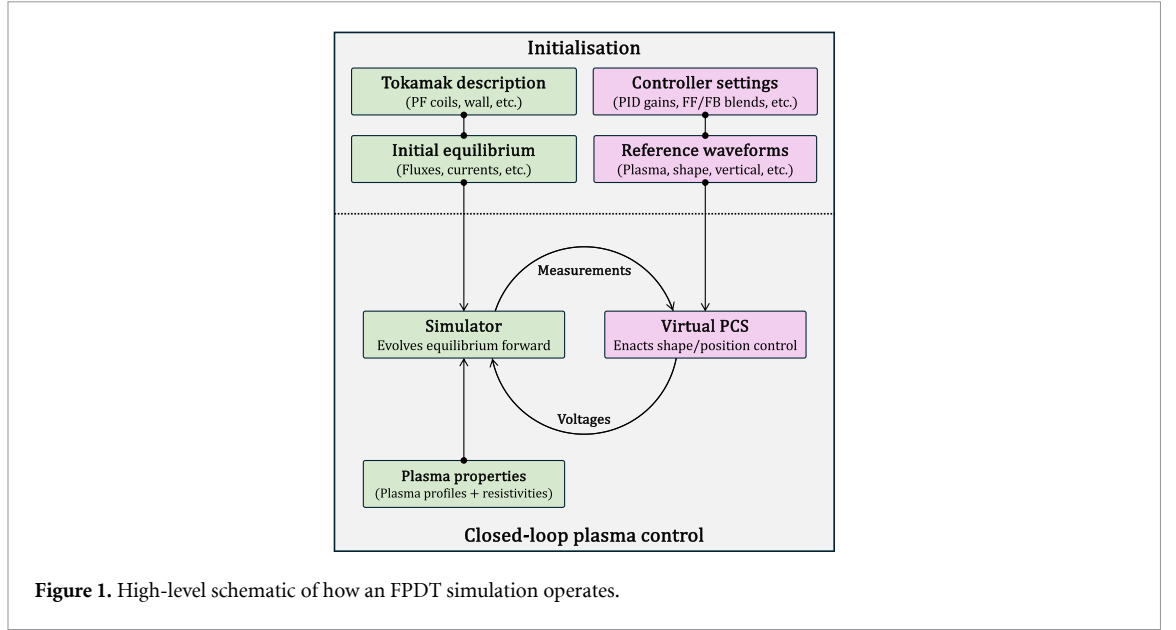


Figure 1. High-level schematic of how an FPDT simulation operates.

where

- $I_p(t)$ is the total plasma current.
- $\mathcal{R}_{y,y}(t)$ denotes the (diagonal) resistance matrix of the plasma. We define

$$\text{diag } \mathcal{R}_{y,y}(t) = 2\pi (\mathbf{R}_y \oslash \mathbf{A}_y) \rho_p(t),$$

with $\mathbf{R}_y$ being the cylindrical radii of each domain element, $\mathbf{A}_y$ their areas, and $\rho_p(t)$ the (effective) user-defined plasma resistivity—the $\oslash$ denotes element-wise division.

- $\mathcal{M}_{y,y}$ is the inductance matrix of plasma-to-plasma current elements.

Finally, the GS equation, which describes the magnetohydrodynamic equilibrium (force balance) of an axisymmetric plasma, is given by

$$\Delta^* \psi(R, Z) = -\mu_0 R J_\phi(\psi, R, Z; \boldsymbol{\theta}), \quad (R, Z) \in \Omega, \quad (3)$$

expressed in cylindrical coordinates (R, ϕ, Z) over the rectangular domain Ω . Here, we have the linear elliptic operator $\Delta^* := R \partial_R R^{-1} \partial_R + \partial_{ZZ}$, the poloidal magnetic flux ψ , the magnetic permeability of free space μ_0 , and the (toroidal) plasma current density distribution J_ϕ . This equation is coupled with a Dirichlet boundary condition on the boundary $(\partial\Omega)$ of the computational domain—see Pentland *et al* (2024), equation (3.4)

The vector $\boldsymbol{\theta}$ contains the ‘profile parameters’ that define the form of the plasma current density distribution J_ϕ inside the last closed flux surface of the plasma, in addition to the safety factor q . In particular, they define the shapes of $dp/d\psi$ and $FdF/d\psi$, where $p(\psi)$ denotes the pressure and $F(\psi)$ denotes the toroidal magnetic field profile. While many different parametrisations exist, we adopt the Lao *et al* (1985) polynomial descriptions—more details on this implementation can be found in Pentland *et al* (2024).

2.2.2. Evolved tokamak plasma state and user-defined quantities

By coupling and solving equations (1) – (3), FreeGSNKE determines the evolution of the tokamak plasma state

$$\mathbf{S}(t) = \left(\underbrace{\mathbf{I}_m(t), I_p(t)}_{\text{evolved}} ; \underbrace{\boldsymbol{\theta}(t), \rho_p(t)}_{\text{prescribed}} \right),$$

ensuring the plasma proceeds through configurations that remain in equilibrium on the Alfvén timescale (Jardin 2010). The plasma equilibrium at time t , given through the GS equation, is fully described by the set of metal currents $I_m(t)$, the total plasma current $I_p(t)$, and the profile parameters $\theta(t)$. While these quantities are sufficient for describing the instantaneous plasma state, the resistivity $\rho_p(t)$ is also required to define evolution.

The evolution of both $I_m(t)$ and $I_p(t)$ are fully determined in FreeGSNKE by (1) – (3). However, the user needs to fully define the time evolution of both $\theta(t)$ and $\rho_p(t)$ for the duration of the simulated time interval. This is because, at present, FreeGSNKE does not directly model current diffusion or self-consistently evolve the resistivity within the plasma—these features will be introduced in the future. From the tokamak plasma state $S(t)$, we can derive required quantities of interest, including any simulated measurements needed in the virtual PCS at each time step. These measurements enable the virtual PCS to provide the PF coil voltages $V_{act}(t)$ back to the simulator, which are then used in the CE (1) for evolution to the next time step—recall figure 1.

Before moving forward, we reiterate that $I_p(t)$ is evolved self-consistently by FreeGSNKE rather than prescribed through the assigned plasma profile parameters $\theta(t)$. These parameters determine the *shape* of the $dp/d\psi$ and $FdF/d\psi$ profiles, as well as their *relative* size, whereas their *absolute* size is set by $I_p(t)$ itself.

2.2.3. Solution modes

The FPDT offers three different solution modes for the system (1) – (3), namely:

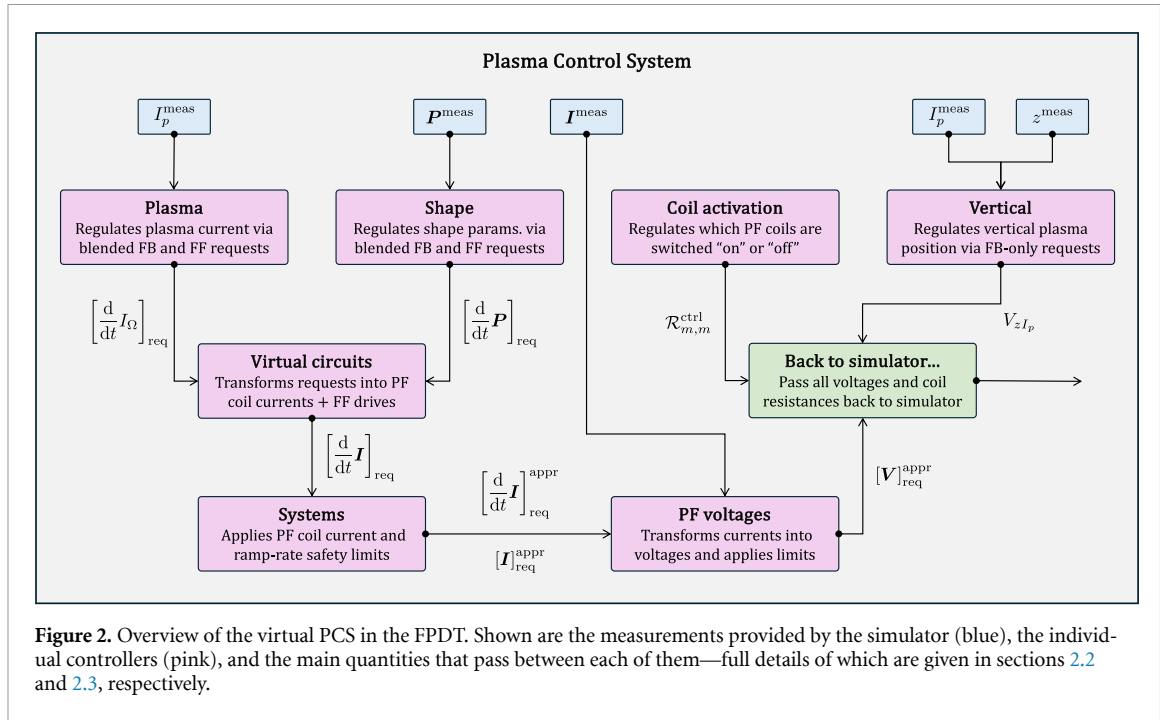
- Fully nonlinear evolution; which we refer to with NL.
- Piecewise linear evolution of the dynamics (1) – (2) coupled with fully nonlinear solution of GS (3); which we refer to with PwLD.
- Piecewise linear evolution for the dynamics (1) – (2) and GS (3); which we refer to with PwL.

A full description of the NL mode is provided in Amorisco *et al* (2024), section 4.2. It uses an implicit Euler scheme to discretise (1) – (2) in time and a staggered approach, based on the Newton–Krylov method, to solve the resulting nonlinear system with (3). Similarly, the linearisation of (1) – (2) used by both the PwLD and PwL modes is presented in Amorisco *et al* (2024), equations 28 and 29. It relies on (finite difference) Jacobians of the plasma current distribution I_y with respect to all of the independent variables at some linearisation time t_0 : the metal currents I_m , plasma current I_p , and profile parameters θ . Here, we introduce the ability to automatically update these Jacobians as the simulation progresses, which gives rise to the ‘piecewise’ aspect.

Before discussing the criteria used to trigger a re-calculation of the Jacobians, it is useful to clarify the difference between PwLD and PwL modes. In the PwLD mode, only the dynamics (1) – (2) are linearised, meaning that the nonlinear GS equation (3) is solved at every time step (based on metal currents and plasma currents obtained from the evolution of the linearised dynamics equations). Solution of the GS equation provides the virtual PCS with exact plasma descriptors for each equilibrium, for instance, vertical position and plasma shape parameters. In the PwL mode, the GS problem is also linearised, meaning that the observable plasma descriptors necessary for control are not exact but are linearised with respect to the metal currents I_m , plasma current I_p , and profile parameters θ (see appendix A for further details). Both modes are significantly less computationally demanding than the NL mode, however, their accuracy depends on the criteria used to trigger an update of the linearisations. In the PwLD mode, we use a criterion based on the evolution of the plasma current distribution: the linearisation is updated when the relative difference (in norm) between the evolving plasma current $I_y(t)$ and that at the latest linearisation $I_y(t_0)$ exceeds some threshold. In PwL mode, the full plasma current distribution $I_y(t)$ is not available, as the GS equation is not solved at all time steps. Therefore, it requires a criterion based on the relative change in the user-defined plasma descriptors. The performance of these criteria is described in section 3.

2.2.4. Tokamak description, vessel mode reduction, and time step selection

To initialise the FPDT, a tokamak geometry must be specified. At minimum this must include a set of PF coils and a limiter geometry that defines the region accessible to the plasma, and may also include passive conducting structures and diagnostics—see Pentland *et al* (2024), Lvovskiy *et al* (2025), and the FreeGSNKE documentation for more details. When building the tokamak object, FreeGSNKE calculates



$\mathcal{M}_{m,m}$ and $\mathcal{R}_{m,m}$ based on the input geometry data and resistivity values specified for the PF coils and passive structures. The user can override these calculations and provide custom resistance and mutual inductance matrices, which is often necessary when attempting a quantitative comparison with experiment¹⁰. Inductances between plasma elements and the PF coils, i.e. $\mathcal{M}_{y,y}$ or $\mathcal{M}_{y,m}$, are calculated based on geometry.

As detailed by Amorisco *et al* (2024), section 3.1, passive structure eigenmodes that evolve on exceedingly fast timescales can be omitted from the simulation to reduce computational runtime. It should be noted, however, that retaining fewer modes systematically weakens the passive vertical stabilisation provided by eddy currents and therefore reduces the vertical instability timescale associated with the equilibrium. As a consequence, the minimum viable number of modes depends on the regime being simulated; therefore, highly elongated plasmas, for instance, may require a higher number of eigenmodes to be retained¹¹. The characteristic timescale of the vertical instability is also crucial to the selection of the simulation time step dt . In order to resolve the vertical dynamics, the time step should be at least a factor 5–10 smaller than the vertical instability timescale, which is calculated automatically by FreeGSNKE’s evolutive solver upon initialisation.

Finally, in order to run the FPDT, the evolutive solver should be provided with an initial condition. In practice, one typically needs to define a full tokamak-plasma state $\mathcal{S}(t)$ at $t = t_{init}$, to allow FreeGSNKE to solve the associated static forward GS problem. When using the FPDT to simulate a variation on a previous discharge, this initial state can, for example, be built using equilibrium reconstruction data.

2.3. The PCS

The PCS class in FreeGSNKE is a collection of interchangeable and adaptable algorithms for controlling key plasma quantities via a combination of FF and FB control—see figure 2 for an illustration of the workflow. This modular structure and the customisability of each of the individual controllers makes it straightforward to adapt the system to different control requirements and different machine configurations.

2.3.1. FB and FF control

Before describing each controller separately, we briefly clarify our nomenclature regarding FB and FF control. In the FPDT, controllers use a combination of both FB and FF requests. For a generic quantity

¹⁰ In practice, purely geometric resistance and inductance values are often insufficient because, for example, they do not account for wiring between PF coils and the power supplies.

¹¹ We refer the reader to the FreeGSNKE documentation for further details: <https://docs.freegsnke.com/>.

$X = X(t)$, we define the total request as

$$\left[\frac{d}{dt} X \right]_{\text{req}} = b_X \left[\frac{d}{dt} X \right]_{\text{FB}} + (1 - b_X) \left[\frac{d}{dt} X \right]_{\text{FF}}, \quad (4)$$

where $b_X = b_X(t) \in [0, 1]$ is a ‘blend’ coefficient defining the convex combination between the FB and FF terms.

Throughout the FPDT, the FB terms (as in (4)) are provided by PID controllers:

$$\left[\frac{d}{dt} X \right]_{\text{FB}} = K_X^{\text{prop}} \delta_X + K_X^{\text{int}} \iota_X + K_X^{\text{der}} \frac{d}{dt} \delta_X, \quad (5)$$

which correct departures from a desired reference value. The proportional term is based on the error

$$\delta_X(t) = X^{\text{ref,FB}}(t) - X^{\text{meas}}(t), \quad (6)$$

and has gain $K_X^{\text{prop}} = K_X^{\text{prop}}(t)$. The user-defined reference waveform for X is denoted $X^{\text{ref,FB}}(t)$ and its real-time measurement is $X^{\text{meas}}(t)$. The integral term, assuming the simulation and PCS time step dt are the same here, uses ι_X as an approximation of the integral of the error:

$$\iota_X(t) = \iota_X(t - dt) + \delta_X(t) \, dt, \quad (7)$$

and has gain $K_X^{\text{int}} = K_X^{\text{int}}(t)$.

$$\frac{d}{dt} \delta_X(t) = (\delta_X(t) - \delta_X(t - dt)) / dt, \quad (8)$$

and has gain $K_X^{\text{der}} = K_X^{\text{der}}(t)$.

The FF term in (4) makes no reference to real-time measurements and enacts prescribed requests which are entirely user-defined.

$$\left[\frac{d}{dt} X \right]_{\text{FF}} = \frac{d}{dt} X^{\text{ref,FF}}, \quad (9)$$

or by setting the FF derivative on the right hand side directly with a waveform. Note that the FB and FF reference waveforms used in (6) and (9) may be different.

2.3.2. Plasma current controller

The plasma current controller regulates the total plasma current, ensuring that it follows the desired reference trajectory. It does so by producing a request, of the same FF and FB structure as (4), for the rate of change of current in the Ohmic PF coil(s) (denoted here as I_Ω). The FB contribution is produced by a PID controller (5), using the error

$$\delta_{I_p}(t) = I_p^{\text{ref,FB}}(t) - I_p^{\text{meas}}(t),$$

between the user-defined FB reference $I_p^{\text{ref,FB}}(t)$ and the measured plasma current $I_p^{\text{meas}}(t)$. As before, the gains used by this controller, $(K_{I_p}^{\text{prop}}, K_{I_p}^{\text{int}}, K_{I_p}^{\text{der}})$, are user-defined waveforms. The FF contribution enables the user to set a waveform for the loop voltage on the plasma $V_{\text{loop}}^{\text{FF}}(t)$. The combined FB and FF request is given by

$$\left[\frac{d}{dt} I_\Omega \right]_{\text{req}} = b_{I_p} \left[\frac{d}{dt} I_p \right]_{\text{FB}} + (1 - b_{I_p}) \frac{V_{\text{loop}}^{\text{FF}}}{\mathcal{M}_\Omega}, \quad (10)$$

where $\mathcal{M}_\Omega(t)$ is a waveform for the mutual inductance between the plasma and the Ohmic coils and b_{I_p} is the blend coefficient (recall (4)).

2.3.3. Shape controller

The shape controller regulates a set of user-specified shape parameters, that can include both core and divertor descriptors, such that they follow their specified reference trajectories. Such descriptors can take a variety of forms, including physical coordinates of specific plasma features (e.g. X-points, strikepoints, etc); poloidal flux values at given locations; and closest ‘gaps’ between, say, the plasma separatrix and a chosen point on the first wall.

The shape controller output is a request for the rate of change of each shape parameter, collected here in the vector $\mathbf{P}(t)$. Following the same structure as (4), it includes both FB and FF contributions:

$$\left[\frac{d}{dt} \mathbf{P} \right]_{\text{req}} = \mathbf{b}_P \left[\frac{d}{dt} \mathbf{P} \right]_{\text{FB}} + (1 - \mathbf{b}_P) \left[\frac{d}{dt} \mathbf{P} \right]_{\text{FF}}, \quad (11)$$

where $\mathbf{b}_P$ is a vector of blend coefficients. The FB contribution is produced by a PID controller (5) with error

$$\delta_P(t) = \mathbf{P}^{\text{ref,FB}}(t) - \mathbf{P}^{\text{meas}}(t),$$

and user-defined vectors of gains ($\mathbf{K}_P^{\text{prop}}$, $\mathbf{K}_P^{\text{int}}$, $\mathbf{K}_P^{\text{der}}$). The FB reference waveform is denoted by $\mathbf{P}^{\text{ref,FB}}(t)$ and the measurement of the shape parameters by $\mathbf{P}^{\text{meas}}(t)$. The FF contribution is as in (9) and uses a FF reference waveform $\mathbf{P}^{\text{ref,FF}}(t)$.

2.3.4. Virtual circuit controller

The VCs controller combines the requests from the plasma and shape controllers, transforming them into requests for the rate of change of currents in the PF coils using VCs.

First, let us focus on control of the magnetic topology. In order to move each shape parameter independently of all the others, a VC matrix can be used. A VC matrix $\mathcal{V}$ is defined as the pseudoinverse of the sensitivity matrix, which itself is the Jacobian of the shape parameters with respect to currents in the PF coils, for a specific equilibrium (e.g. Lvovskiy *et al* 2025). Each column of the VC matrix gives the changes in PF coil currents required to change a given shape parameter by one unit, without changing any of the other shape parameters considered (i.e. it decouples shape parameter changes from one another). VC matrices are, however, a function of the equilibrium from which they were generated and so this decoupling becomes less accurate when a VC matrix is used to control a different equilibrium from the one it was derived for. The VCs Controller therefore enables use of a schedule of VC matrices $\mathcal{V}(t)$ to be used between different times in the simulation.

These VCs enable us to transform the request (11) into a request for the rate of change of PF coil currents:

$$\left[\frac{d}{dt} \mathbf{I} \right]_{\text{req},P} = \mathcal{V}_P(t) \left[\frac{d}{dt} \mathbf{P} \right]_{\text{req}}. \quad (12)$$

Note that $\mathcal{V}_P(t)$ is the VC matrix specific to the shape parameters in $\mathbf{P}$, while below we will use $\mathcal{V}_\Omega(t)$ to denote the VC for the Ohmic circuit. We have also dropped the subscript from $\mathbf{I}_{\text{act}}$, used to denote the PF coil currents, for conciseness.

In a similar way, the rate of change of PF coil currents resulting from the request made by the plasma current controller is

$$\left[\frac{d}{dt} \mathbf{I} \right]_{\text{req},\Omega} = \mathcal{V}_\Omega(t) \left[\frac{d}{dt} I_\Omega \right]_{\text{req}}. \quad (13)$$

Instead of being a matrix, $\mathcal{V}_\Omega$ is a vector of dimensionless weights defining the Ohmic circuit, which is (ideally) built such that its contributions to the total poloidal flux do not cause the plasma shape parameters to change (e.g. McArdle *et al* 2020). If so, the request (13) should not cause undesired movements of the plasma parameters beyond those driven by (12).

These requests are then combined with any user-defined FF coil current requests $\mathbf{I}^{\text{ref,FF}}(t)$, such that

$$\left[\frac{d}{dt} \mathbf{I} \right]_{\text{req}} = \left[\frac{d}{dt} \mathbf{I} \right]_{\text{req},P} + \left[\frac{d}{dt} \mathbf{I} \right]_{\text{req},\Omega} + \frac{d}{dt} \mathbf{I}^{\text{ref,FF}}, \quad (14)$$

giving the final output from the VCs controller.

2.3.5. Systems controller

The systems controller enforces PF coil current limits for machine protection and safe tokamak operation to the requests from the VCs Controller. First, limits are applied to the requested rate of change of the PF coil currents:

$$\left[\frac{d}{dt}\mathbf{I}\right]_{\text{req}}^{\text{appr}} = \min\left(|\dot{\mathbf{I}}_{\text{ramp}}|, \max\left(-|\dot{\mathbf{I}}_{\text{ramp}}|, \left[\frac{d}{dt}\mathbf{I}\right]_{\text{req}}\right)\right),$$

where $|\dot{\mathbf{I}}_{\text{ramp}}|(t)$ denotes the user-defined vector of maximum allowed rate of change per coil and $[\frac{d\mathbf{I}}{dt}]_{\text{req}}^{\text{appr}}$ is the resulting approved request. These are then integrated to obtain a set of requested PF coil currents:

$$[\mathbf{I}(t)]_{\text{req}} = [\mathbf{I}(t - dt)]_{\text{req}}^{\text{appr}} + \left[\frac{d}{dt}\mathbf{I}\right]_{\text{req}}^{\text{appr}} dt.$$

These are then optionally clipped based on user-defined upper $\mathbf{I}^{\text{up}}(t)$ and lower $\mathbf{I}^{\text{low}}(t)$ coil current limits, to obtain the final approved current requests

$$[\mathbf{I}(t)]_{\text{req}}^{\text{appr}} = \min\left(\mathbf{I}^{\text{up}}, \max\left(\mathbf{I}^{\text{low}}, [\mathbf{I}(t)]_{\text{req}}\right)\right). \quad (15)$$

2.3.6. PF voltages controller

The PF voltages controller transforms the approved coil current requests from the systems controller into the final voltage requests.

The first contribution to the voltage requests is obtained using a set of simplified CE:

$$[\mathbf{V}(t)]_{\text{req,CE}} = \mathcal{R}(t)\mathbf{I}^{\text{meas}}(t) + \mathcal{M}^{\text{FF}}(t)\left[\frac{d}{dt}\mathbf{I}\right]_{\text{req}}^{\text{appr}}. \quad (16)$$

Here, $\mathcal{R}(t)$ and $\mathcal{M}^{\text{FF}}(t)$ are user-defined resistance and mutual inductance matrices for the PF coils. With respect to the CEs described in section 2.2.1, those in (16) are substantially simplified and do not account for the plasma or the passive structures. To compensate for this, the matrices $\mathcal{R}(t)$ and $\mathcal{M}^{\text{FF}}(t)$ will be different to those defined in section 2.2.1 and can be defined as functions of time if required.

The second contribution is FB driven and is of the form

$$[\mathbf{V}(t)]_{\text{req,FB}} = \mathcal{M}^{\text{FB}}(t)\left[\frac{d}{dt}\mathbf{I}\right]_{\text{FB}}, \quad (17)$$

where a second user-defined mutual inductance matrix, $\mathcal{M}^{\text{FB}}(t)$, potentially different from the one used in (16), can be adopted. The FB derivative on the right hand side is produced by a dedicated PID controller with gains $(\mathbf{K}_I^{\text{ptop}}, \mathbf{K}_I^{\text{int}}, \mathbf{K}_I^{\text{der}})$ and based on the discrepancy

$$\delta_{\mathbf{I}}(t) = [\mathbf{I}(t)]_{\text{req}}^{\text{appr}} - \mathbf{I}^{\text{meas}}(t),$$

between the approved active coil current requests and the measured PF coil currents $\mathbf{I}^{\text{meas}}(t)$. Combining (16) and (17), we get the total (unapproved) voltage requests

$$[\mathbf{V}(t)]_{\text{req}} = [\mathbf{V}(t)]_{\text{req,CE}} + [\mathbf{V}(t)]_{\text{req,FB}}. \quad (18)$$

These requests are also clipped to adhere to any operational limits. First, a clip can be applied to the voltage requests themselves:

$$[\mathbf{V}(t)]_{\text{req}}^{\text{clip}} = \min\left(|\mathbf{V}_{\text{lim}}|, \max\left(-|\mathbf{V}_{\text{lim}}|, [\mathbf{V}(t)]_{\text{req}}\right)\right),$$

where the limits $\mathbf{V}_{\text{lim}}(t)$ are user-defined waveforms. Then, a clip can be applied to the derivatives of the voltage requests using $|\dot{\mathbf{V}}_{\text{ramp}}|(t)$, and built as follows

$$\left[\frac{d}{dt}\mathbf{V}\right]_{\text{req}} = \left([\mathbf{V}(t)]_{\text{req}}^{\text{clip}} - [\mathbf{V}(t - dt)]_{\text{req}}^{\text{appr}}\right)/dt,$$

so that

$$\left[\frac{d}{dt} \mathbf{V} \right]_{\text{req}}^{\text{appr}} = \min \left(|\dot{\mathbf{V}}_{\text{ramp}}|, \max \left(-|\dot{\mathbf{V}}_{\text{ramp}}|, \left[\frac{d}{dt} \mathbf{V} \right]_{\text{req}} \right) \right).$$

With this, we can build the approved voltage requests

$$[\mathbf{V}(t)]_{\text{req}}^{\text{appr}} = [\mathbf{V}(t - dt)]_{\text{req}}^{\text{appr}} + \left[\frac{d}{dt} \mathbf{V} \right]_{\text{req}}^{\text{appr}} dt, \quad (19)$$

which are passed to the simulator along with the voltage to be applied to the vertical control coil (21) and coil resistances (22).

2.3.7. Vertical controller

Elongated plasmas are inherently vertically unstable (e.g. Jardin 2010, section. 4.6.6) and require active FB control. The PF coils used for vertical control on MAST-U are separate from the plasma shaping coils and consist of an up/down symmetric pair of coils wired in anti-series. These generate a radial magnetic field, and hence up/down forces that stabilise the plasma vertically.

The vertical controller in the FPDT is a purely FB driven controller, based on the error

$$\delta_{z_{I_p}}(t) = I_p^{\text{meas}}(t) (z^{\text{ref,FB}}(t) - z^{\text{meas}}(t)),$$

where $z^{\text{ref,FB}}(t)$ is the user-defined reference waveform and $I_p^{\text{meas}}(t)$ the measured plasma current. We measure the average vertical coordinate of the plasma current distribution using

$$z^{\text{meas}}(t) = \frac{1}{I_p^{\text{meas}}(t)} \int_{\Omega_p} Z J_\phi(\psi, R, Z; \boldsymbol{\theta}) dR dZ, \quad (20)$$

noting that J_ϕ is measured at time t , though not explicitly defined here. A dedicated PID controller uses this discrepancy to produce the following voltage request

$$V_{z_{I_p}}(t) = K_{z_{I_p}}^{\text{prop}} \delta_{z_{I_p}} + K_{z_{I_p}}^{\text{int}} \int \delta_{z_{I_p}} + K_{z_{I_p}}^{\text{der}} \frac{d}{dt} \delta_{z_{I_p}}, \quad (21)$$

which is applied to the vertical control coils. The gains used by this controller, $(K_{z_{I_p}}^{\text{prop}}, K_{z_{I_p}}^{\text{int}}, K_{z_{I_p}}^{\text{der}})$, are provided as user-defined waveforms.

2.3.8. Coil activation controller

The coil activation controller handles the timings at which the PF coils are switched on during the shot—an essential feature for modeling the pre-plasma and plasma ramp-up phases when coils can only be activated in certain sequences. Rather than enforcing this by explicitly varying the number of active coils at different times, which would require changes to the machine description itself, we do so instead by altering the resistance matrix $\mathcal{R}_{m,m}$ dynamically, recall (1). A PF coil can effectively be disconnected (switched off) by setting its resistance to an arbitrarily large value, denoted here as Ω_∞ .

In (1), we therefore overwrite the default $\mathcal{R}_{m,m}$ with

$$\mathcal{R}_{m,m}^{\text{ctrl}}(t) = \mathcal{R}_{m,m} + \Omega_\infty \text{diag} \left(\mathbf{1} - \mathbf{h}^{\text{activation}}(t) \right), \quad (22)$$

where $\mathbf{h}^{\text{activation}}(t)$ is a vector of boolean values indicating the on (1) and off (0) state of each coil at time t and $\mathbf{1}$ is a vector of ones.

While the ability to modify resistances during a simulation is *not* something a real PCS would do, this functionality exploits the fact that the virtual PCS and simulator coexist as objects within the same computational framework. This enables us to not only switch coils on or off, but also test deliberate mismatches between the controller's assumed parameters and those of the physical model—for example in the resistance matrices as a means of probing robustness to calibration errors or sensitivity to model uncertainty.

3. Validation using MAST-U discharges

3.1. Overview

3.1.1. Objectives

Before using the FPDT for predictive simulations, it is essential we validate its performance by comparing to experimental data. We do so by re-simulating past MAST-U shots: the close proximity of the MAST-U PF coils to the plasma and each other drive strong cross-coupling effects, making plasma control particularly challenging and therefore an excellent test case for the FPDT.

On one hand, we aim to demonstrate that the FPDT can effectively sustain and control the evolution of plasma current, vertical position, and key shape parameters, as requested by the reference waveforms. In order to do so, we will quantify any discrepancy between the evolution of the FB controlled parameters $X(t)$ (whether from simulation or experiment) with the desired FB reference waveform using the mean absolute difference (MAD):

$$\text{MAD}(X) = \frac{1}{N+1} \sum_{k=0}^N |X(t_k) - X^{\text{ref,FB}}(t_k)|, \quad (23)$$

and the relative MAD (RMAD):

$$\text{RMAD}(X) = \frac{1}{N+1} \sum_{k=0}^N \frac{|X(t_k) - X^{\text{ref,FB}}(t_k)|}{|X^{\text{ref,FB}}(t_k)|}, \quad (24)$$

where N denotes the number of time steps in the simulation/experiment.

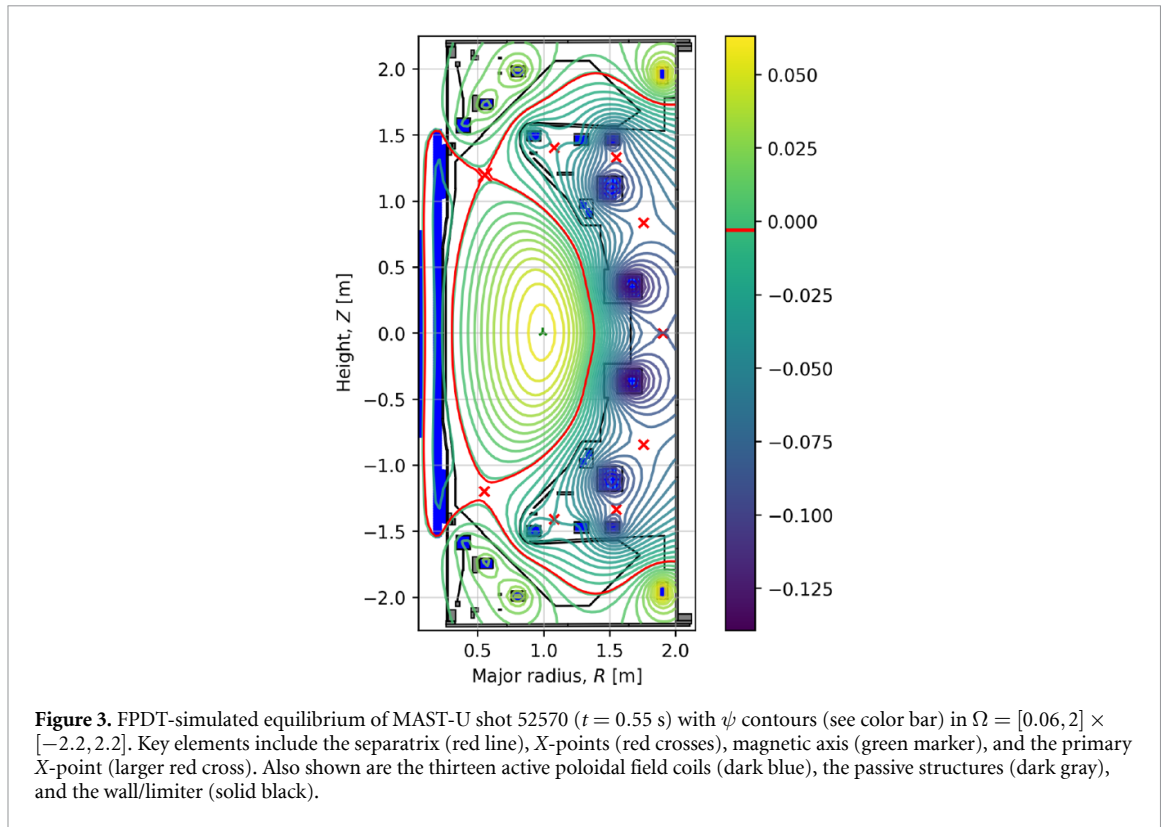
On the other hand, we wish to compare simulation results to the experimental measurements themselves. As we provide the FPDT with the same control settings the actual MAST-U PCS was provided with, we should find agreement between our simulations and the experimental data. However, it is worth highlighting that this latter comparison has clear limitations:

- The experimental data is affected by other forms of control (e.g. fueling, impurity seeding, etc), and by transient events (e.g. internal reconnection events, edge localised modes) that are not captured by the FPDT simulations.
- The FPDT extracts measurements of controlled quantities from the simulated equilibria, whereas experimental values are obtained from magnetic measurements via LEMUR (Kochan *et al* 2023), which can be subject to noise. Any discrepancies and biases introduced by this difference are left for future analysis.

3.1.2. Simulation setup

The MAST-U machine description includes thirteen active PF coils and 150 passive structure elements (see Pentland *et al* (2024)). For the PF coils, we calculate calibrated resistances and mutual inductances using vacuum shot data, whereas for the passive structures, we use purely geometric values. All presented simulations are based on a reduction of the passive structures to 30 eigenmodes—those that couple most strongly to the plasma. We use a 65×65 computational grid and a fixed time step of 0.2 ms, small enough to resolve the vertical dynamics of the plasmas we encounter. We do, however, run the virtual PCS at twice this frequency, i.e. every 0.1 ms, so that we match the frequency that the actual MAST-U PCS runs at. An update of the linearisation used by the PwLD simulation mode is triggered when the relative change in plasma current density exceeds 5%. The update is triggered in the PwL mode when any of the core shape parameters (see next section) change by more than 1 cm; the radial nose position by 5 cm; the strike point¹² by 15 cm; or the vertical position by 2.5 mm. In addition, for the NL mode, the relative convergence tolerances for the dynamics (1) – (2) and GS (3) are set to 1%, i.e. the residuals in the extensive currents and plasma flux are required to be smaller than 1% of their respective change in each time step (we refer the reader to Amorisco *et al* (2024), equations 25 and 26 for further details).

¹² This criteria is larger because the linearisation does not lose significant accuracy when doing a leg sweep—it primarily uses the outer divertor coils, which do not impact the other shape parameters significantly.



The simulations here are focused on the flat-top phase, $t \geq 0.2$ s, with future work aiming to extend this to the ramp-up phase. We start our simulations with static equilibria built using EFIT++ reconstruction data (Team *et al* 2022), including plasma, coil, and passive structure currents, as well as the associated plasma profiles (see below). Through testing, we found that it is often useful to partially nudge these initial equilibria towards the reference waveforms before simulation as the intrinsic uncertainties in the EFIT++ reconstructions introduce exceedingly large initial departures, affecting the subsequent time evolution in a spurious manner.

In the absence of self-consistent plasma profile evolution, we prescribe the parameters $\theta(t)$ as a function of time by using the magnetics-only EFIT++ reconstruction data for the given shot. This prescribed evolution takes into account the heating and current drive generated by the neutral beam injectors on MAST-U, as FreeGSNKE does not currently account for any non-inductive current drive (whose contribution on MAST-U is small anyway, see Turnyanskiy *et al* (2009)). Similarly, the time trace of the plasma resistivity $\rho_p(t)$ is also prescribed. We obtain an estimate of the plasma resistivity in the following MAST-U shots from separate ‘open-loop’ simulations. Instead of using our virtual PCS class, in these we employ the voltages recorded by the real MAST-U PCS (with the exception of the vertical control coil which we continue to control in FB with the virtual PCS). The resistivity of the plasma is then tuned by a custom controller so to that the evolution of the total plasma current is matched to the FB waveform. This offers a rapid approximation of the plasma resistivity for use in the ‘closed-loop’ FPDT simulations that follow.

3.2. Shot 52570 (Super-X divertor)

Here, we re-simulate the flat-top phase of MAST-U shot 52570, a double null 750 kA plasma heated by two sets of neutral beams, designed to demonstrate the formation of a super-X divertor (Valanju *et al* 2009, Morris *et al* 2014, 2018). This is a key configuration for MAST-U divertor studies, as it sweeps the divertor strikepoints to a large major radius, thereby enabling larger connection lengths, flux expansion, and area over which heat flux is deposited on divertor tiles (see figure 3). In the following time series results display real-time LEMUR measurements available to the MAST-U PCS in orange, together with results from the different FPDT simulation modes. The legends display values of the metrics (23) (with units) and (24) (with percentage), comparing the FB reference waveforms to both simulation and experimental measurements. Green background shading represents when FB control is on, yellow when FF control is on, and white when no control is used.

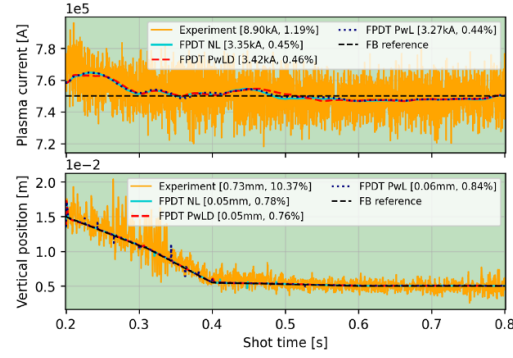


Figure 4. FPDT-simulated evolution of total plasma current (top panel) and vertical position (bottom panel) for MAST-U shot 52570 using the NL (solid light blue), PwLD (dashed red), and PwL (dotted dark blue) simulation modes. Also shown are the FB reference waveforms used in the real and virtual PCS (dashed black) and the measured experimental data from the discharge (orange). Background shading indicates when FB control is on (green), when FF control is on (yellow), and when no control occurs (white)—see later figures for other shading.

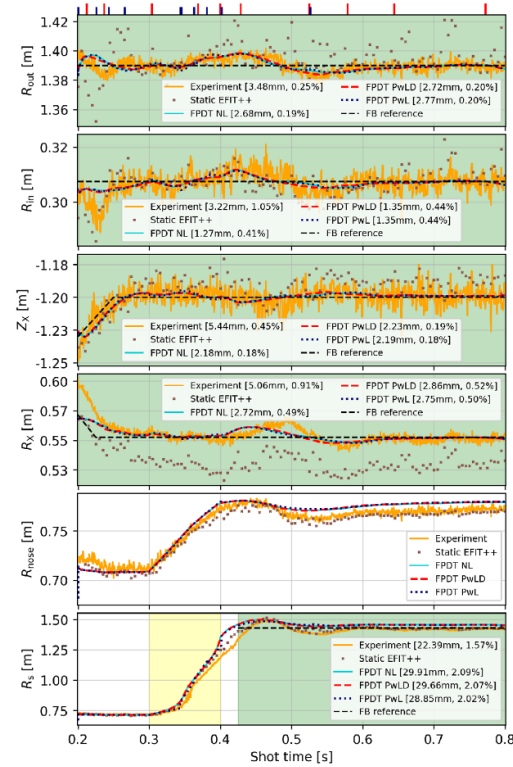
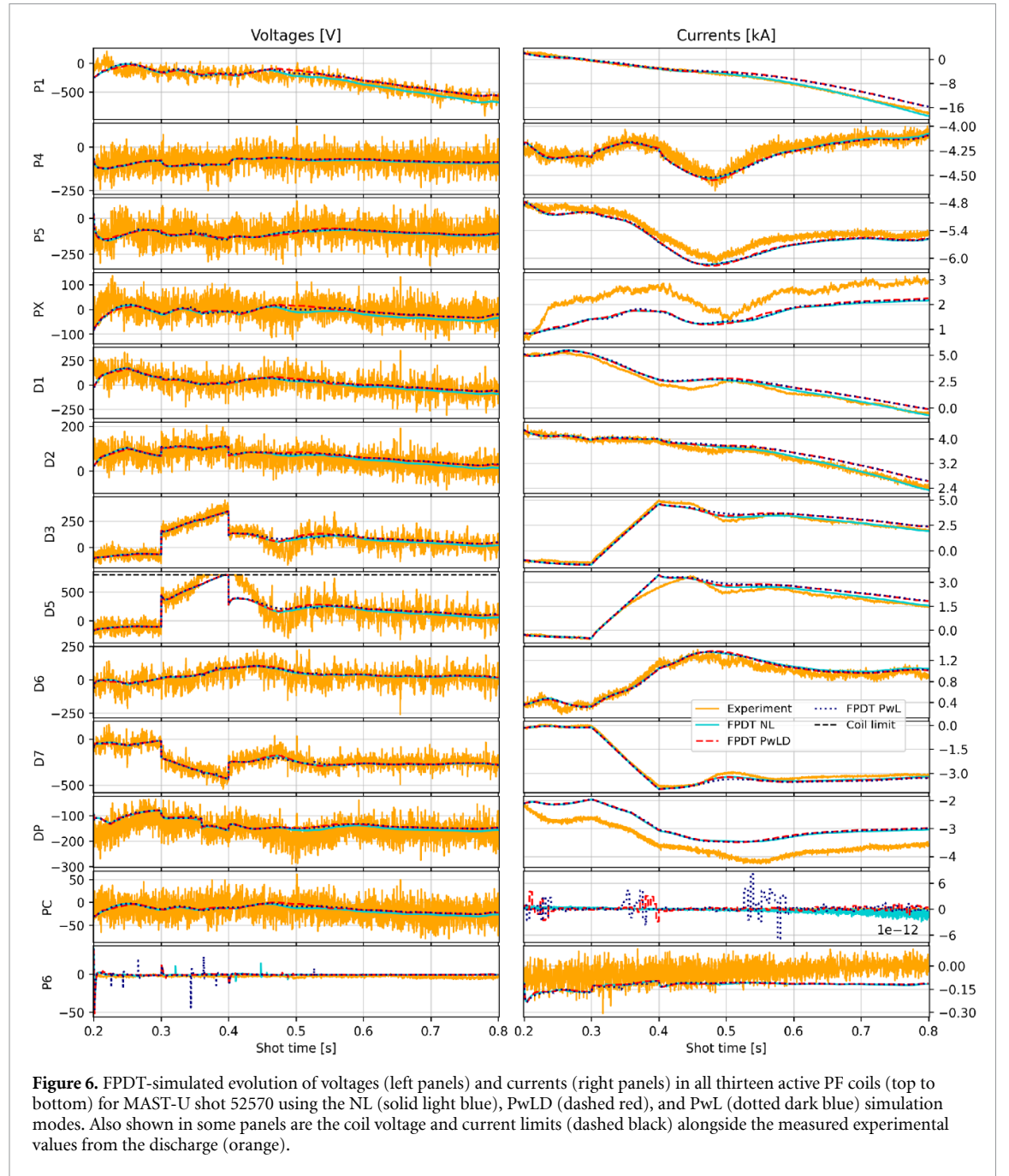


Figure 5. FPDT-simulated evolution of several shape parameters (as described in the main text) for MAST-U shot 52570 using the NL (solid light blue), PwLD (dashed red), and PwL (dotted dark blue) simulation modes. Also shown are the FB (dashed black) reference waveforms used in the real and virtual PCS, alongside the LEMUR measured experimental data from the discharge (orange) and values from static FreeGSNKE-simulated equilibria using EFIT++ reconstruction data (brown crosses). Background shading indicates when FB control is on (green), when FF control is on (yellow), and when no control occurs (white). Tick marks above the top panel indicate times at which relineralisation took place in the PwLD (red) and PwL (dark blue) simulations.

In figure 4, we focus on plasma current (top panel) and vertical position (bottom panel). Excellent agreement can be seen in both, with all simulation modes achieving RMAD values below 0.5% for the plasma current and 1% for the vertical position.

In figure 5, we plot the evolution of six commonly controlled shape parameters in MAST-U (Anand *et al* 2024), namely:

- R_{in} and R_{out} : the inboard and outboard midplane radii.
- R_X and Z_X : the radial and vertical positions of the X-point near the lower divertor chamber.



- R_{nose} : the radial intersection between the lower divertor leg (separatrix) and a virtual line between the center of the DP and D2 coils—see Pentland *et al* (2024), figure 1.
- R_{S} : the radial intersection between the lower divertor leg and the divertor chamber tiles.

In addition to the real-time LEMUR data, each panel of figure 5 also displays values of the same shape parameter obtained from EFIT++ reconstruction data (brown crosses). The size of fast variations in the EFIT++ data and the magnitude of the discrepancy sometimes present between LEMUR and EFIT++ measurements of the same shape parameter provide a qualitative scale for the typical uncertainty in these quantities. In turn, this scale is useful when contextualising the size of any discrepancies between simulations and experimental results. In the first two panels, we can see that both R_{in} and R_{out} are successfully held at their respective constant FB reference values throughout the simulations in all cases, with MAD values no larger than those from the experimental results. Correspondingly, the third panel shows Z_{X} is successfully shifted upwards over 3 cm from its starting value in little under 50 ms and, again, held constant thereafter (with RMAD values less than 0.25%). Similarly, R_{X} is successfully moved inwards in the fourth panel according to the FB waveform (albeit with slightly larger RMAD values

than Z_X , reflected in the uncertainty with the EFIT++ values). The fifth panel displays the uncontrolled parameter R_{nose} evolving similarly to both experimental LEMUR and EFIT++ values. In the final panel, the strike point is swept outwards using a FF drive between 300 ms and 400 ms, moving from a conventional divertor scenario to a super-X one. After a short period of no control, the super-X divertor is thereafter maintained successfully in FB.

Figure 6 displays the evolution of the PF coil voltages produced by the virtual PCS (left panels) and the subsequent evolution of the currents (right panels) in the FPDT, compared to the experimentally recorded values. Although not shown here, the passive structure currents are also evolved and can be monitored. We observe excellent overall qualitative agreement between the calculated voltages in all three simulation modes, noting in particular the correct application of the voltage limits—which are briefly reached on the D5 coil. During the first 10 ms of simulation, however, larger than normal differences can be seen, driven primarily by difference between the actual and simulated initial equilibrium conditions, which we inherit from the EFIT++ initial conditions. We can clearly see the impact that the FF phase on R_s between 300 ms and 400 ms has on the divertor (D) coils with sharp jumps in voltages captured well by the virtual PCS. We should also note that the PC coil is disconnected from the machine, but included in our machine model: a voltage is calculated by the PCS but no current is generated. We should note that for the P6 voltage, in reality, z_{meas} is not actually monitored in real-time on MAST-U; instead, the change in poloidal flux at two locations either side of the midplane is used as a proxy for vertical position. While both measurements serve the same purpose, the P6 voltages produced by the FPDT vertical controller here will be slightly different from those in actual MAST-U discharges.

Despite starting very close to the experimental values (and with the correct voltages applied), a few of the coil currents (PX and DP) diverge after a short period of time, even though, as we saw, the plasma current, shape, and position all evolve correctly. This may suggest that the calibration of our machine model (or perhaps unmodeled 3D effects), including the coil resistances and mutual inductances, can potentially be improved. This said, the remaining currents match well; in particular the solenoid (P1), indicating that the resistivity has been well estimated.

All simulations reported were run on a single workstation (MacBook Pro, Apple M2 Pro chip, 16 GB RAM); no specialised or parallel hardware was used. The total runtime for the NL simulation mode was 81 min 11 s, clearly rendering it prohibitively expensive for inter-shot scenario development and control design. It does, however, provide us with a useful baseline to assess the accuracy of the PwLD and PwL simulation modes, both of which performed comparably to the NL with runtimes an order of magnitude smaller. The PwLD ran in 4 min 32 s, updating its linearisation 10 times, while the PwL ran in 1 min 51 s, also relinearising 10 times. The times at which linearisation updates took place are indicated above the first panel in figure 5. As might be expected, most relinearisation takes place when the plasma shape changes significantly, i.e. early in simulations or when strike point sweeping takes place. Despite improved performance in the PwLD and PwL modes, the dominant computational bottleneck remains the cost of relinearising around the evolving equilibrium, together with uncertainty in how frequent this is required. There are, however, ways to mitigate this by using fewer passive structure modes or possibly re-using existing linearisations when those are applicable.

3.3. Shot 50366 (X-point target divertor)

Next, we re-simulate the flat-top phase of MAST-U shot 50366, a double null, 750 kA, Ohmically heated plasma, which was designed to test the TED framework (Tokamak Exhaust Designer, see Bardsley *et al* 2025) and generate an X-point target divertor configuration. The X-point target configuration generates a secondary X-point in the divertor chamber, close to the divertor leg (see figure 7), in order to improve detachment access and reduce particle and heat fluxes (LaBombard *et al* 2015, Lonigro *et al* 2026).

Similarly to shot 52570, the simulated plasma current remains within 0.5% of the FB reference, despite being initialised over 10 kA from the reference itself—see top panel of figure 8. The lower panel, again, demonstrates good vertical position control in all cases, tracking the steadily increasing FB reference.

As displayed in figure 9, the evolution of the shape parameters appears affected by the overshoot in I_p in figure 8, especially R_{out} and R_{in} . Analogous initial oscillations are seen in both the LEMUR and EFIT++ measurements. In our simulations they are successfully controlled to less than 5 mm on average over the simulation, with peaks of approximately 2 cm for R_{out} . The ramp up in Z_X (third panel) to form a smaller plasma core is tracked well, with the evolution of R_X following the EFIT++ reconstruction despite it being uncontrolled (fourth panel). In this panel, and as previously noted (Anand *et al* 2024, figure 1(b)), we observe a bias between the LEMUR measurements and EFIT++ for R_X . The bottom panel shows good qualitative agreement for R_{nose} , which is also uncontrolled.

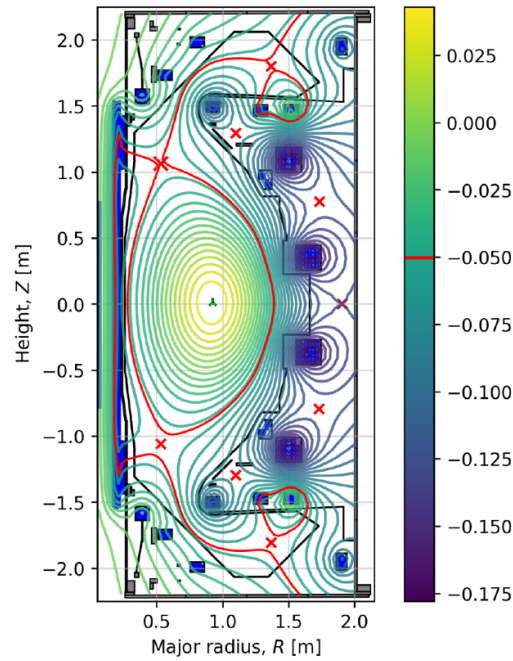


Figure 7. FPDT-simulated equilibrium of MAST-U shot 50366 ($t = 0.50$ s) with ψ contours (see color bar) in $\Omega = [0.06, 2] \times [-2.2, 2.2]$. Key elements include the separatrix (red line), X-points (red crosses), magnetic axis (green marker), and the primary X-point (larger red cross). Also shown are the thirteen active poloidal field coils (dark blue), the passive structures (dark gray), and the wall (solid black).

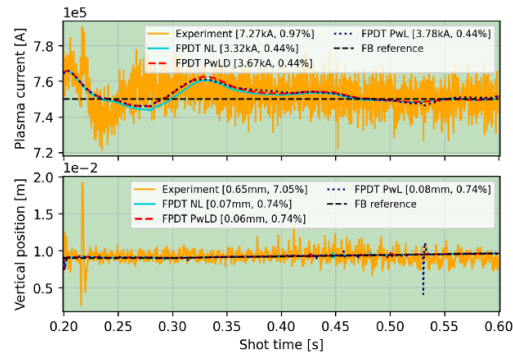
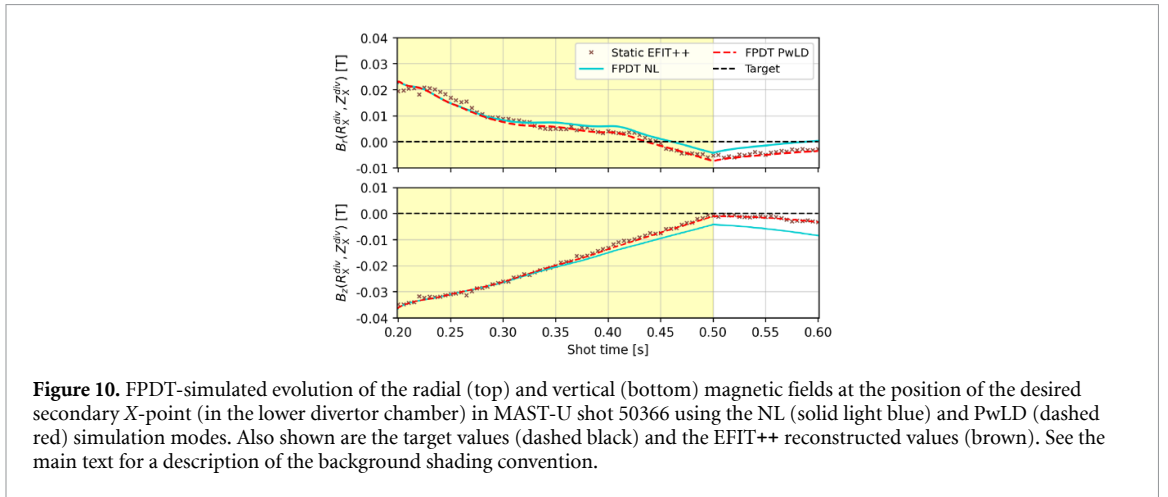
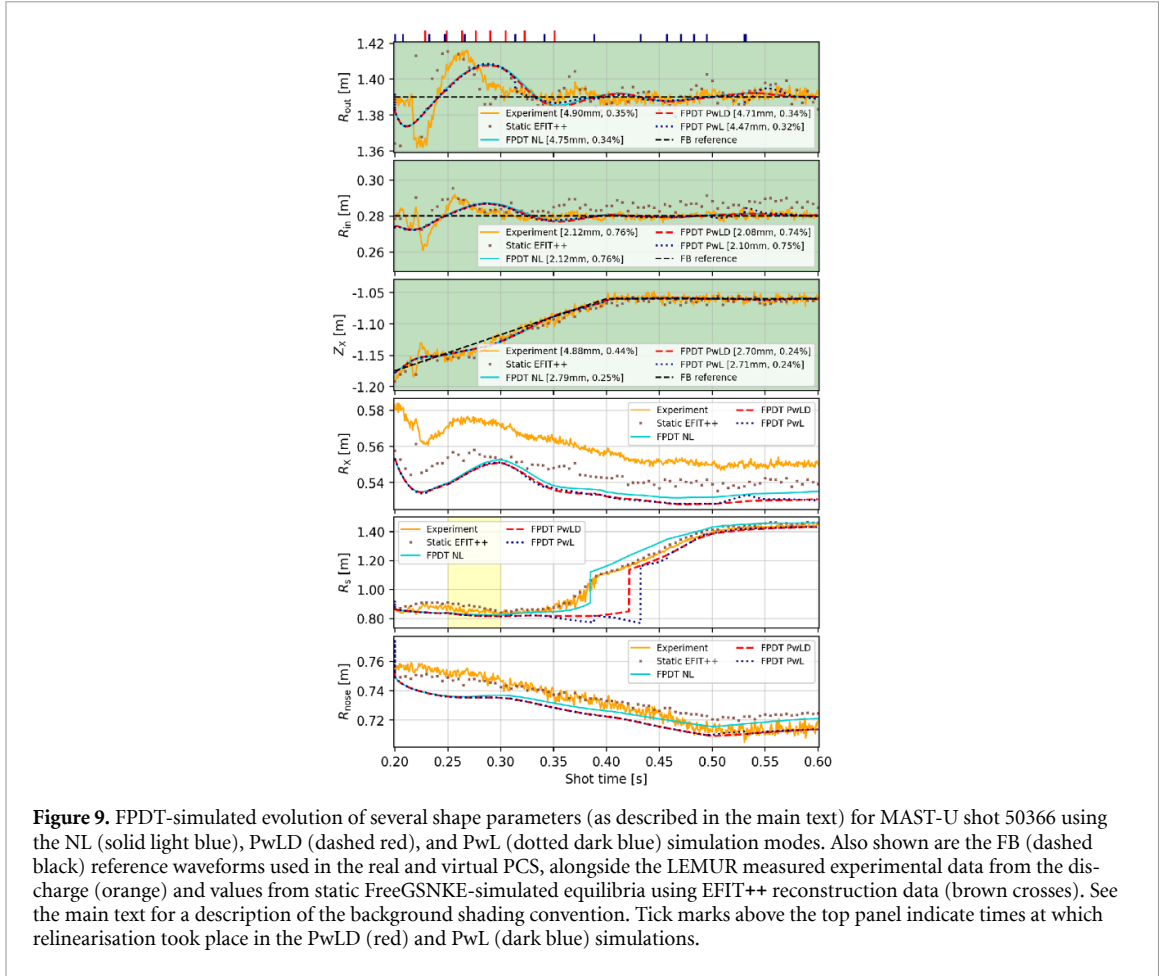


Figure 8. FPDT-simulated evolution of total plasma current (top panel) and vertical position (bottom panel) for MAST-U shot 50366 using the NL (solid light blue), PwLD (dashed red), and PwL (dotted dark blue) simulation modes. Also shown are the FB reference waveforms used in the real and virtual PCS (dashed black) and the measured experimental data from the discharge (orange). See the main text for a description of the background shading convention.

The fifth panel illustrates the evolution of R_s , which is not directly controlled besides a 50 ms period where R_s is driven in FF. Each simulation mode is characterised by what appears to be a discontinuous strike point location, appearing sometime between 350 ms and 450 ms. Figure 11 illustrates the dynamics of the strike point in these simulations, which is in fact continuous: the apparent jumps are a result of the divertor leg ‘brushing against’ the divertor tiles, running almost tangent to them. This phenomenology is not captured by the LEMUR data or by the EFIT++ reconstructions, but is ubiquitous in our simulation modes.

As mentioned, this shot aimed to create a secondary X-point in the divertor chamber. The location could not, however, be directly controlled in FB, as it is not measured in real-time by LEMUR. To circumvent this limit, an additional FF drive is used, as shown in figure 10. This drive is active between 200 ms and 500 ms and uses a specifically designed VC to form the X-point at $(R_{X_{\text{div}}}, Z_{X_{\text{div}}}) = (1.2, -1.8)$. This drive aims to form a magnetic null at this location, i.e. $B_r(R_{X_{\text{div}}}, Z_{X_{\text{div}}}) = B_z(R_{X_{\text{div}}}, Z_{X_{\text{div}}}) = 0$, while also fixing a point along the divertor leg in place, which explains the dynamics of the leg seen in figure 11). In figure 10, we show the evolution of magnetic null at this secondary X-point in both the NL and PwLD FPDT simulations, compared to the static EFIT++-based equilibria.



Both simulations approach the target values toward the end of the FF drive period. We note that no PwL (dotted dark blue) simulation is shown, as this additional X-point was not included in the list of plasma descriptors.

The evolution of the coil voltages and currents is shown in figure 12, with overall good agreement with experiment. Deviations of some coil currents from the measurements again indicate that further improvements to the resistance and inductance matrices are possible.

The simulation execution times are similar to those of the previous experiment but we report them here for completeness. The NL simulation took 74 min 59 s; the PwLD took 6 min 4 s (with 8 relinearisations); and the PwL took 2 min 37 s (with 16 relinearisations). Note that some of the PwL linearisation thresholds were adjusted to account for the larger shape changes: Z_X was increased to 2.5 cm, R_{nose} reduced to 1 cm, and R_s reduced to 5 cm.

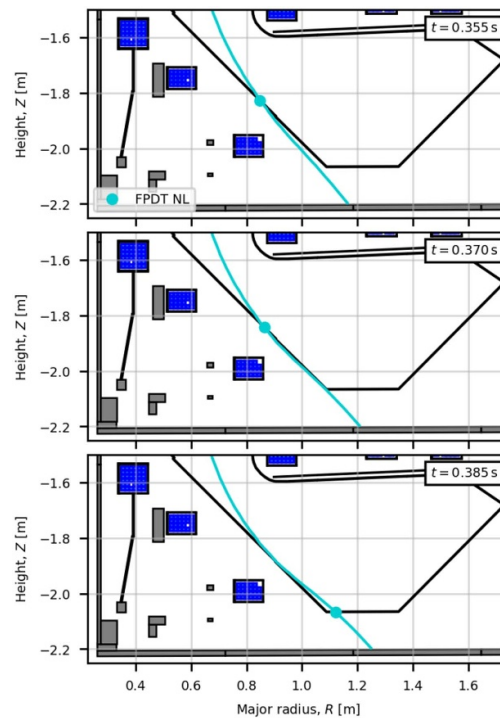


Figure 11. FPDT-simulated evolution of the divertor leg in MAST-U shot 50366 using the NL (solid light blue) simulation mode. The dot indicates the location of the corresponding strikepoint R_s .

4. Summary and outlook

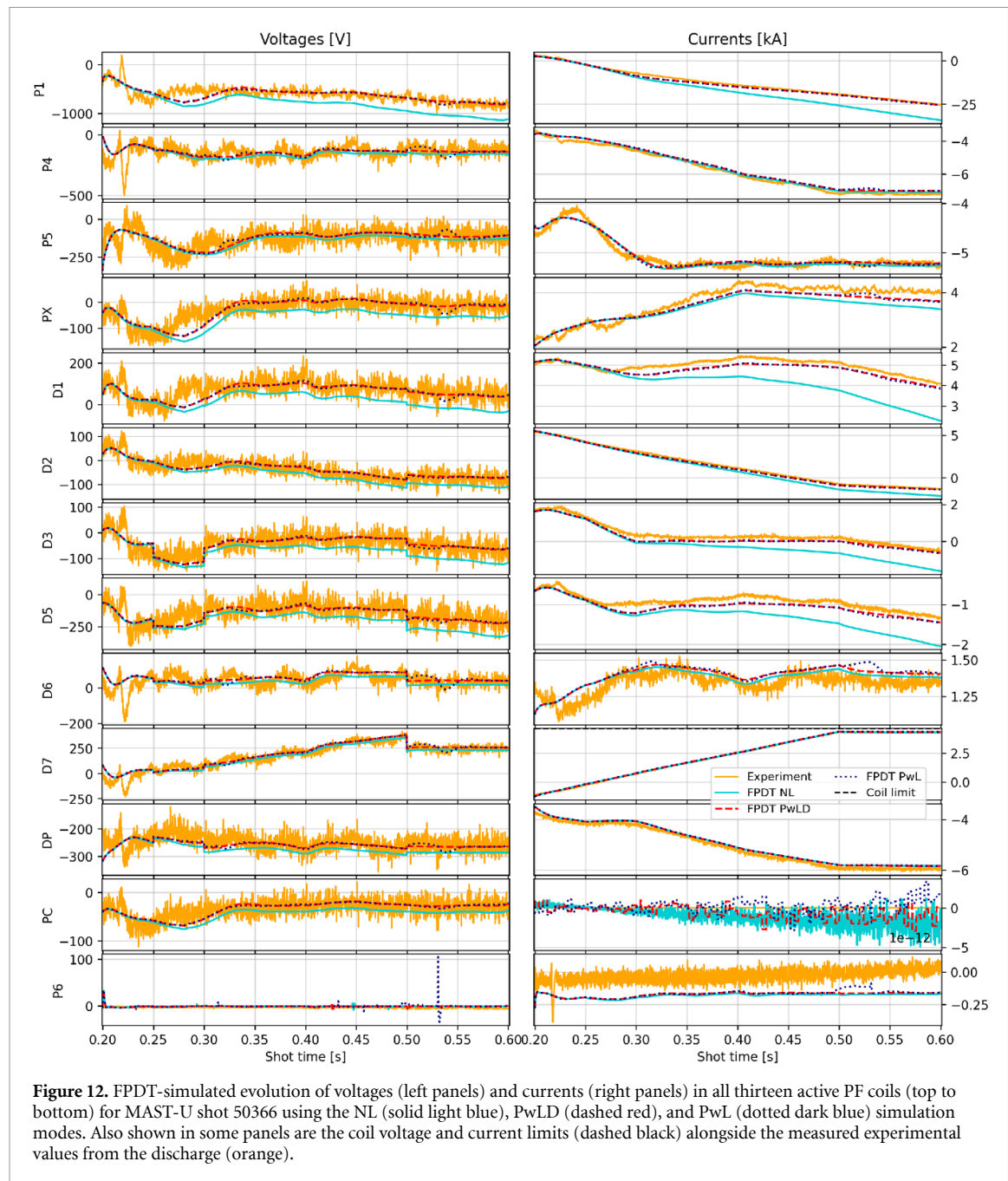
This paper presents the FPDT, a simulation framework for axisymmetric tokamak plasma equilibria that couples FreeGSNKE's evolutive equilibrium solver with a new customisable virtual PCS class. While the virtual PCS presented here reflects the MAST-U control architecture, the modular design of the FPDT means that alternative magnetic control strategies such as isoflux or gap control can be implemented by substituting the relevant controller components without modifying the underlying equilibrium solver or communication infrastructure.

We validate the efficacy of the FPDT by re-simulating existing MAST-U shots with advanced divertor configurations, using the same FB/FF reference waveforms and control settings as in the real discharge. The simulations show excellent quantitative agreement with both reference waveforms and experimental measurements from the discharge itself. The agreement observed in the evolved coil currents is particularly noteworthy, as any discrepancy between measured data and the simulations would be expected to accumulate over time and manifest itself as a progressively increasing deviation. Small divergences in the coil current trajectories are observed nonetheless, and are likely due to small inaccuracies in the machine calibration—something to investigate in future work.

Three different simulation modes were compared: NL, PwLD, and PwL, with the latter two capable of automatically updating their respective linearisations of the dynamics (and GS for PwL) as a simulation progresses. We show, however, that both the PwLD and PwL modes can produce trajectories with accuracy comparable to the NL mode, while reducing computational cost by approximately an order of magnitude. Using either of these piecewise linear modes enables the rapid evolution of key plasma parameters, with turnaround times of only a few minutes, making them well suited to inter-shot plasma scenario development and control design.

The validation demonstrated here ensures the FPDT is ready for use in predictive shape control modeling of the flat-top phase, enabling pre-shot testing of challenging plasma configurations. These predictions do, however, remain dependent on the reconstructed plasma profiles from previous discharges—something that could be avoided with the addition of transport and current diffusion models in the future. Beyond classical control studies, the FPDT also provides a powerful framework for the development and testing of AI-supported control schemes such as those based on emulated VCs (Agnello *et al* 2024, Cavestany *et al* 2025, Sarkar *et al* 2026).

Future work will focus on the use of the FPDT for pre-shot control validation in the fifth MAST-U campaign and extending the simulation window to include the ramp-up phase—where vessel currents



play a larger role in the equilibrium evolution. This would enable full shot scenario and control design, linking the plasma and pre-magnetisation phases together. In addition, a dedicated power supply model that incorporates delays or nonlinearities in the dynamics of power supply switching would be beneficial.

Acknowledgments

We would like to thank Geof Cunningham, Stuart Henderson, James Harrison, Jimmy Measures, and Oliver Bardsley (UKAEA) for some very insightful and helpful discussions around the MAST-U PCS. This work was funded by the Fusion Computing Lab collaboration (between UKAEA and STFC Hartree) and part funded by the EPSRC Energy Programme (EP/W006839/1). For the purpose of open access, the authors have applied a Creative Commons Attribution (CC BY) licence to any author accepted manuscript version arising from this submission. To obtain further information, please contact publicationsmanager@ukaea.uk.

Data availability statement

The FPDT can be found within the FreeGSNKE open-source suite at <https://github.com/FusionComputingLab/freegsnke>.

The data cannot be made publicly available upon publication because they are owned by a third party and the terms of use prevent public distribution.

Conflict interests

The authors have no conflicts of interest to declare.

Appendix. Piecewise linear evolution for the dynamics and GS

In the PwL simulation mode, the GS equation (3) is not solved at each time step. Instead, the plasma state is provided by a reduced-order representation comprising a set of plasma descriptors most relevant to control. In this paper, the user-defined plasma descriptors, denoted $\mathbf{D}(t)$, consist of variables that describe the plasma's local or global shape (e.g. midplane radii, elongation, etc) or position (e.g. vertical position).

Following the solution of (1) – (2) at a given time t in PwL mode, the plasma descriptors are evolved according to

$$\mathbf{D}(t) = \mathbf{D}(t_0) + \delta_{\mathbf{I}_e}(t) \left. \frac{\partial \mathbf{D}}{\partial \mathbf{I}_e} \right|_{t=t_0} + \delta_{\boldsymbol{\theta}}(t) \left. \frac{\partial \mathbf{D}}{\partial \boldsymbol{\theta}} \right|_{t=t_0},$$

where $\delta_x(t) = x(t) - x(t_0)$ and $\mathbf{I}_e = (\mathbf{I}_m, \mathbf{I}_p)$. The Jacobians of the descriptors with respect to $\mathbf{I}_e$ and $\boldsymbol{\theta}$ are calculated at the same time t_0 as the Jacobians used to solve the dynamics (1) – (2). This is done using the equilibrium at time t_0 , which triggered the latest relinearisation process. This enables faster simulations that bypass the large number of GS solves of the PwLD mode, while retaining a physically meaningful description of plasma evolution, thereby facilitating controller development and testing.

ORCID iDs

K Pentland  0000-0001-9530-1890
 N C Amorisco  0000-0002-3205-5335
 A Ross  0000-0002-0787-6756
 T Nunn  0000-0001-7608-5512
 A Agnello  0000-0001-9775-0331
 G K Holt  0000-0001-6814-9117
 G McArdle  0000-0003-0888-0105
 C Vincent  0000-0002-7227-409X
 S J P Pamela  0000-0001-8854-1749

References

- Agnello A, Amorisco N C, Keats A, Holt G K, Buchanan J, Pamela S J P, Vincent C and McArdle G 2024 Emulation techniques for scenario and classical control design of tokamak plasmas *Phys. Plasmas* **31** 043901
- Albanese R *et al* 2005 Design, implementation and test of the XSC extreme shape controller in JET *Fusion Eng. Des.* **74** 627–32
- Albanese R, Ambrosino R and Mattei M 2015 CREATE-NL+: a robust control-oriented free boundary dynamic plasma equilibrium solver *Fusion Eng. Des.* **96–97** 664–7
- Amorisco N C, Agnello A, Holt G, Mars M, Buchanan J and Pamela S J P 2024 FreeGSNKE: a python-based dynamic free-boundary toroidal plasma equilibrium solver *Phys. Plasmas* **31** 042517
- Anand H *et al* 2024 Real-time plasma equilibrium reconstruction and shape control for the MAST upgrade tokamak *Nucl. Fusion* **64** 086051
- Ariola M and Pironti A 2016 *Magnetic Control of Tokamak Plasmas* 2nd edn (*Advances in Industrial Control*) (Springer) (<https://doi.org/10.1007/978-3-319-29890-0>)
- Artaud J *et al* 2018 Metis: a fast integrated tokamak modelling tool for scenario design *Nucl. Fusion* **58** 105001
- Bardsley O P *et al* (the MAST-Upgrade Team) 2025 The tokamak exhaust designer: a new tool for enhanced divertor control on MAST-U *Proc. 51st EPS Conf. on Controlled Fusion and Plasma Physics* p 2.032 (available at: <https://lac913.epfl.ch/epsppd3/2025/html/TitlePage.html>)

- Blum J, Heumann H, Nardon E and Song X 2019 Automating the design of tokamak experiment scenarios *J. Comput. Phys.* **394** 594–614
- Buxton P F 2025 GSFit (available at: <https://github.com/tokamakenergy/gsfitt>)
- Carpanese F 2021 Development of free-boundary equilibrium and transport solvers for simulation and real-time interpretation of tokamak experiments. *PhD Thesis* EPFL
- Cavestany P, Ross A, Agnello A, Garrod A, Amorisco N C, Holt G K, Pentland K and Buchanan J 2025 Real-time applicability of emulated virtual circuits for tokamak plasma shape control (arXiv:2509.01789)
- Clark R, Nurgaliev M, Khayrutdinov E, Subbotin G, Welander A and Orlov D M 2025 Validation of NSFsim as a Grad-Shafranov equilibrium solver at DIII-D *Fusion Eng. Des.* **211** 114765
- Coleman M *et al* 2025 (available at: <https://github.com/Fusion-Power-Plant-Framework/bluemira>)
- De Tommasi G, Albanese R, Ambrosino G, Ariola M, Mattei M, Pironti A, Sartori F and Villone F 2007 XSC Tools: a software suite for tokamak plasma shape control design and validation *IEEE Trans. Plasma Sci.* **35** 709–23
- Degrave J *et al* 2022 Magnetic control of tokamak plasmas through deep reinforcement learning *Nature* **602** 414–9
- Fable E *et al* 2022 The modeling of a tokamak plasma discharge, from first principles to a flight simulator *Plasma Phys. Control. Fusion* **64** 044002
- Farmakalides A, Nikiforakis N, Millmore S, Romanelli M and Buxton P F 2025 CRATOS-GS: a free-boundary, hierarchical adaptive mesh refinement Grad–Shafranov solver *AIP Adv.* **15** 095128
- Galperti C *et al* 2024 Overview of the TCV digital real-time plasma control system and its applications *Fusion Eng. Des.* **208** 114640
- Gerškšič S and De Tommasi G 2016 ITER plasma current and shape control using MPC 2016 *IEEE Conf. on Control Applications (CCA)* pp 599–604
- Grad H and Rubin H 1958 Hydromagnetic equilibria and force-free fields *Proc. 2nd United Nations Int. Conf. on the Peaceful Uses of Atomic Energy: Theoretical and Experimental Aspects of Controlled Nuclear Fusion* vol 31 United Nations, p 400
- Gros G, Faugeras B, Boulbe C, Artaud J F, Nouailletas R and Rapetti F 2025 Numerical simulation of tokamak plasma equilibrium evolution *J. Comput. Phys.* **529** 113849
- Heiß C, Merle A, Carpanese F, Felici F, Donner C, Marchioni S, Mari A and Sauter O 2025 FGE: a fast free-boundary Grad-Shafranov evolutive solver (arXiv:2512.06847)
- Heumann H *et al* 2015 Quasi-static free-boundary equilibrium of toroidal plasma with CEDRES++: computational methods and applications *J. Plasma Phys.* **81** 905810301
- Humphreys D A, Ferron J R, Hyatt A W, Haye R J L, Leuer J A, Penaflor B G, Walker M L, Welander A S and In Y 2008 DIII-D integrated plasma control solutions for ITER and next-generation tokamaks *Fusion Eng. Des.* **83** 193–7
- Ivanov A, Khayrutdinov R, Medvedev S and Poshekhonov Y 2009 The SPIDER code - solution of direct and inverse problems for free boundary tokamak plasma equilibrium *Keldysh Inst. Prepr.* 1–24
- Janky F, Dnestrovskii A, Medvedev S Y, Nemytov V, Vincent B, Buxton P F, Erskine T, Sinha J and Asunta O 2026 SOPHIA: a tokamak simulator *Fusion Eng. Des.* **222** 115447
- Janky F, Fable E, Englberger M and Treutterer W 2021 Validation of the Fenix ASDEX Upgrade flight simulator *Fusion Eng. Des.* **163** 112126
- Jardin S 2010 *Computational Methods in Plasma Physics* (CRC Press)
- Kerboua-Benlarbi S, Nouailletas R, Faugeras B, Nardon E and Moreau P 2024 Magnetic control of WEST plasmas through deep reinforcement learning *IEEE Trans. Plasma Sci.* **52** 3698–703
- Khayrutdinov R R and Lukash V E 1993 Studies of plasma equilibrium and transport in a tokamak fusion device with the inverse-variable technique *J. Comput. Phys.* **109** 193–201
- Kochan M *et al* 2023 Real-time plasma shape reconstruction on MAST Upgrade based on local expansion *Proc. 30th IEEE Symp. on Fusion Engineering (SOFE)* (IEEE)
- Kool B *et al* 2025 Demonstration of Super-X divertor exhaust control for transient heat load management in compact fusion reactors *Nat. Energy* **10** 1116–31
- kuang J, Yang J, Ren Z, Zhang P, Wang Y and Wang W 2026 Implementation of a free-boundary equilibrium solver for the cloverleaf configuration and its implications for x-point radiators *Plasma Sci. Technol.* **28** 045102
- LaBombard B *et al* 2015 ADX: a high field, high power density, advanced divertor and RF tokamak *Nucl. Fusion* **55** 053020
- Lao L L, John H S, Stambaugh R D, Kellman A G and Pfeiffer W 1985 Reconstruction of current profile parameters and plasma shapes in tokamaks *Nucl. Fusion* **25** 1611
- Lonigro N, *et al* (the EUROfusion Tokamak Exploitation Team, the MAST Upgrade Team) 2026 Initial observations in X-point target divertor discharges on MAST-U (arXiv:2601.21840)
- Lvovskiy A *et al* 2025 Framework for assessment of magnetic equilibrium controller performance on the MAST upgrade spherical tokamak *Plasma Phys. Control. Fusion* **67** 075003
- McArdle G J *et al* 2023 Integrated real-time control on the MAST upgrade tokamak 29th IAEA Fusion Energy Conf. (FEC 2023) (<https://scientific-publications.ukaea.uk/papers/integrated-real-time-control-on-the-mast-upgrade-tokamak/>)
- McArdle G, Pangione L and Kochan M 2020 The MAST upgrade plasma control system *Fusion Eng. Des.* **159** 111764
- Mele A, Topalova A A, Galperti C and Coda S (the TCV Team, and the Eurofusion Tokamak Exploitation Team) 2025 the Eurofusion tokamak exploitation team. first experimental demonstration of plasma shape control in a tokamak through model predictive control (arXiv:2506.20096)
- Meneghini O *et al* 2024 FUSE (Fusion Synthesis Engine): a next generation framework for integrated design of fusion pilot plants, (arXiv:2409.05894)
- Morris W *et al* 2014 MAST accomplishments and upgrade for fusion next-steps *IEEE Trans. Plasma Sci.* **42** 402–14
- Morris W, Harrison J R, Kirk A, Lipschultz B, Militello F, Moulton D and Walkden N R 2018 MAST upgrade divertor facility: a test bed for novel divertor solutions *IEEE Trans. Plasma Sci.* **46** 1217–26
- Muldrew S I, Lux H, Cunningham G, Hender T C, Kahn S, Knight P J, Patel B, Voss G M and Wilson H R 2020 PROCESS: systems studies of spherical tokamaks *Fusion Eng. Des.* **154** 111530
- Nouailletas R, Gros G, Faugeras B, Artaud J-F and Moreau P 2025 Development of a flight simulator for the WEST plasma and control system *Fusion Eng. Des.* **219** 115299
- Pajares A, Welander A S, Wehner W P, Thome K E, Eidietis N W and Anand H 2025 Plasma shape and position control development for NSTX-U using the GSEvolve plasma simulator *Fusion Eng. Des.* **220** 115302
- Penaflor B G *et al* 2008 Worldwide collaborative efforts in plasma control software development *Fusion Eng. Des.* **83** 176–80

- Penaflor B G, Ferron J R, Hyatt A W, Walker M L, Johnson R D, Piglowski D A, Kolen E, Welander A S and Lancot M J 2013 Latest advancements in DIII-D plasma control software and hardware *2013 IEEE 25th Symp. on Fusion Engineering (SOFE)* pp 1–4
- Pentland K *et al* 2024 Validation of the static forward Grad–Shafranov equilibrium solvers in FreeGSNKE and fiesta using EFIT++ reconstructions from MAST-U *Phys. Scr.* **100**
- Pentland K, Ross A, Amorisco N A, Cavestany P, Nunn T, Agnello A, Holt G K and Vincent C 2026 Real-time virtual circuits for plasma shape control via neural network surrogates: dynamic validation in closed-loop simulations (arXiv:2604.00781)
- Romanelli M *et al* 2014 JINTRAC: a system of codes for integrated simulation of tokamak scenarios *Plasma Fusion Res.* **9** 3403023
- Sarkar S R, Koneru R B, Gupta R, George R, Kuley A, An sumali S and Kaushal S 2026 Magnetohydrodynamic equilibrium and neutronics study on MAST-U using Jenga framework (arXiv:2605.09913)
- Shafranov V D 1958 On magnetohydrodynamical equilibrium configurations *Sov. J. Exp. Theor. Phys.* **6** 545
- Team M A S T U *et al* 2022 First MAST-U equilibrium reconstructions using the EFIT++ code *48th European Physical Society Conf. on Plasma Physics*
- Tracey B D *et al* 2024 Towards practical reinforcement learning for tokamak magnetic control *Fusion Eng. Des.* **200** 114161
- Turnyanskiy M, Keeling D, Akers R, Cunningham G, Conway N, Meyer H, Michael C and Pinches S 2009 Study of the fast ion confinement and current profile control on MAST *Nucl. Fusion* **49** 065002
- Valanju P M, Kotschenreuther M, Mahajan S M and Canik J 2009 Super-X divertors and high power density fusion devices *Phys. Plasmas* **16** 056110
- Wai J T, Boyer M D, Battaglia D J, Merle A, Carpanese F, Felici F, Kochan M and Kolen E 2025 Feedforward equilibrium trajectory optimization with GSPulse (arXiv:2506.21760)
- Walker M L *et al* 2015 The ITER plasma control system simulation platform *Fusion Eng. Des.* **96–97** 716–9
- Welander A, Olofsson E, Sammuli B, Walker M L and Xiao B 2019 Closed-loop simulation with Grad–Shafranov equilibrium evolution for plasma control system development *Fusion Eng. Des.* **146** 2361–5