

# A general infrastructure for data-driven control design and implementation in tokamaks

Joseph Abbate<sup>1,‡</sup>, Rory Conlin<sup>2</sup>, Ricardo Shousha<sup>2</sup>, Keith Erickson<sup>1</sup> and Egemen Kolemen<sup>1,2,†</sup>

<sup>1</sup>Princeton Plasma Physics Laboratory, Princeton, NJ 08540, USA

<sup>2</sup>Department of Mechanical and Aerospace Engineering, Princeton University, Princeton, NJ 08540, USA

(Received 19 May 2022; revised 4 October 2022; accepted 6 October 2022)

A general infrastructure for tokamak controllers based on data-driven neural net models is presented. The paradigm allows for more flexible choices of both the underlying model and the desired controlled variables and targets. The system is implemented and tested on the DIII-D tokamak, enacting simultaneous pressure and temperature control via a finite-set model-predictive controller. Traditional control methods such as proportional–integral–derivative (PID) have proven effective for decoupled control tasks, but scale poorly when trying to achieve more complicated goals such as full state control. This is exactly where model-based controllers succeed.

**Key words:** fusion plasma, plasma dynamics

---

## 1. Introduction

Modern tokamaks tend to use proportional–integral–derivative (PID) controllers, rather than more advanced controllers such as model-predictive control (MPC), H-infinity, or reinforcement learning. PID controllers need trial-and-error tuning to obtain desired results, which requires many shots even when pre-tuned with simulations offline. However, operators largely are willing to pay this cost because it is infeasible to debug more sophisticated control schemes under a time crunch. Creating a more accurate yet still intuitive and interpretable control would save shots and time, thereby increasing experimental efficiency.

The most robust control mechanisms at the moment are for plasma current, radial position, vertical position and (to a lesser extent) last closed flux surface (LCFS) shape. On top of these base controls, other control algorithms are also sometimes employed. Error fields and resistive wall modes (RWMs) have been controlled via non-axisymmetric magnetic coils; sawteeth and neoclassical tearing modes (NTMs) can be controlled via steerable electron cyclotron current drive (see Walker *et al.* 2020).

† Email address for correspondence: [ekolemen@princeton.edu](mailto:ekolemen@princeton.edu)

‡ Abbate and Conlin contributed equally to this paper, order is alphabetical.

One of the most difficult control problems, however, is full-profile magnetic and kinetic control. Theoretically, many of the modes and parameters that determine fusion performance are derived from these profiles; so by controlling the profiles the operator is directly controlling the quantities that matter. For example, pressure gradients around rational flux surfaces trigger magnetohydrodynamic (MHD) instabilities; kinetic profiles largely determine bootstrap current; and RWM/NTM stability depends on both pressure and rotational velocity. In modern tokamaks, some of the primary knobs for profile control include neutral beams, electron cyclotron waves, ion cyclotron waves, lower hybrid waves, helicon, cold gas fueling, pellet fueling, current ramp rate of the central solenoid and 3D coils (Humphreys *et al.* 2015).

A 2004 JET experiment (Laborde *et al.* 2005) used lower hybrid current drive (LHCD), ion-cyclotron resonance heating (ICRH) and neutral beam injection (NBI) in order to control eight basis functions characterising the safety factor  $q$  and the normalised electron temperature gradient  $\rho_{T_e}^*$ . Both profile and actuator quantities were assumed to be linear perturbations around an experimentally determined steady-state reference. The power deposition profiles from the actuators were assumed fixed (and determined from the reference around which they were linearised). A linear, time-independent response of the profiles to the three actuator values was also assumed. The system response matrix was empirically determined from previous open-loop experiments perturbing the plasma around the reference. This matrix was then used for proportional–integral (PI) feedback in an experiment simultaneously controlling  $q$  and  $\rho_{T_e}^*$ . A variety of other initiatives in the late 1990s/early 2000s also sought to control a limited number of proxy quantities for the profile shape. Some used similar system identification methods for locally linearised model-based control, e.g. Moreau *et al.* (2011) for DIII-D and JT-60 U. Others used simpler feedback (mostly proportional) control, e.g. Ferron *et al.* (2006) for DIII-D, Wijnands *et al.* (1997) for Tore Supra and Fujita & the JT-60 Team (2006) for JT-60 U.

However, more complete control over all profiles (rather than a small number of proxies for individual profile shapes) and global models (rather than carefully tuned local linearisations) are required for some control applications. Through the 2010s a first-principles, model-driven approach was developed for controlling the full  $q$  profile (Barton *et al.* 2013; Wehner *et al.* 2016). The density profile is assumed to maintain a reference shape while varying linearly with the target density actuator  $\bar{n}_e$  (which is achieved via gas puffing by a separate control loop). The temperature profile is also assumed to maintain a reference shape, and scale like  $(I_p/I_p^{ref})^\gamma (P_{tot}/P_{tot}^{ref})^\epsilon (\bar{n}_e/\bar{n}_e^{ref})^\zeta$  for empirically determined coefficients  $\gamma$ ,  $\epsilon$  and  $\zeta$ ; and actuators of plasma current  $I_p$ , total power  $P_{tot}$  and target density  $\bar{n}_e$ . The ion and electron temperatures are assumed equal, and the plasma  $Z_{eff}$  is assumed constant in space and time. Non-inductive current drive from each electron cyclotron gyrotron and each neutral beam is assumed to scale with a reference deposition profile times its power. Using simulations, intuition, and experimental data from the scenario of interest, the reference profiles, reference actuators, empirical coefficients and other parameters are chosen. All of these assumptions combined allow for a linear poloidal flux diffusion (and in turn safety factor  $q$ ) equation with an effective actuator set  $\mathbf{u} = f(I_p, \bar{n}_e, \mathbf{P})$  for  $\mathbf{P}$  the individual power from each neutral beam and each electron cyclotron heating gyrotron, and  $f$  a nonlinear mapping from real to effective actuators.

Via finite difference, the flux diffusion equation in turn allows a linear state-space mapping  $\boldsymbol{\theta}(t+1) = \mathbf{A}\boldsymbol{\theta}(t) + \mathbf{B}\mathbf{u}(t)$  with  $\boldsymbol{\theta} = f(\mathbf{q})$  for  $f$  a nonlinear mapping. This state-space form, where the state evolution is a linear mapping of the present state and actuators, allows an abundance of existing control techniques to be employed. Recent

experiments involved simultaneous control of the interior points of the  $q$  profile via linear MPC to actuate individual powers of neutral beams and the target density, while constraining the model to use a feedforward (i.e. pre-programmed) electron cyclotron heating power. The total power from the beams is constrained to hit a desired total stored energy. The stored energy is assumed to evolve like  $dE/dt = -(E/\tau_E) + P$ , where the energy confinement time  $\tau_E$  is obtained from the  $H_{98}$  scaling (ITER Physics Expert Group on Confinement and Transport 1999); and the injected power  $P$  is assumed to consist only of the neutral beam heating because that is the dominant component. A PI controller is used to determine the best total injected neutral beam power to hit a target stored energy. Meanwhile,  $I_p$  is excluded from the MPC framework and set by a separate linear–quadratic–integral (LQI) control loop (described by Boyer *et al.* 2013) to hit a desired edge  $q$  profile value  $q_{95}$ . In essence then, the group has now demonstrated simultaneous  $q$  profile plus scalar kinetic (via the stored energy) control on DIII-D in a pseudo-first-principles framework, which was tested specifically for the high- $q_{min}$  scenario (Holcomb *et al.* 2015).

An otherwise similar approach to first-principles control instead considers the full nonlinear diffusion equations and *locally* linearises at each timestep. This method was used in a MPC framework to perform offline simulations (Felici & Sauter 2012), scenario optimisation (Teplukhina *et al.* 2017), online state estimation (Felici *et al.* 2011) and realtime  $q$  profile plus  $\beta_n$  control on the TCV tokamak (Maljaars *et al.* 2015, 2017). This local linearisation makes it easier to start from arbitrarily difficult equations, which means more naturally including kinetic profile diffusion ( $T_e$ ,  $T_i$  and  $n_e$  to date) in addition to current profile diffusion (Felici *et al.* 2018).

Unfortunately, after many decades of work developing transport equations the dynamics are still not well-captured, especially for kinetic profiles. The density profile is notoriously hard to predict; and all set-ups require some information at the edge (usually values at  $\rho \sim 0.8$ ) to give decent results. In all the aforementioned methods, there are *ad hoc* assumptions required to obtain the diffusion equations to give semi-accurate results. As more progress continues to be made on the described models, we instead propose a new paradigm for control that uses a fully data-driven model to predict dynamics, and use that model to control all points on all profiles simultaneously. Specifically, we seek to use machine learning and other techniques to generate a model for the plasma state from a large database of experimental data. We then seek to use this model in a model-driven control framework. Machine learning has already been implemented for other areas of tokamak control, especially for disruption prediction (Montes *et al.* 2019; Rea *et al.* 2019; Fu *et al.* 2020). In this paper, we build on the work of (Abbate, Conlin & Kolemen 2021) and (Jalalvand *et al.* 2021) to perform plasma profile control. We describe our workflow to train and deploy deep-learning plasma evolution models in control algorithms on the DIII-D tokamak (Luxon & Davis 1985), within the infrastructure of the machine's standard plasma control system (PCS) (Penaflor *et al.* 2001). The framework is well-modularised so that anyone could train a model with the offline dataset, and then plug their control algorithm into the PCS via these tools. In this paper, we describe this infrastructure, and demonstrate it with an initial, simple test on the device: neural-net-based finite set control of the core pressure and electron temperature via beam power modulation.

## 2. Infrastructure design

Controllers can be viewed as black boxes that map states to actions. These can be anything from simple PID controllers to much more advanced reinforcement learning algorithms.

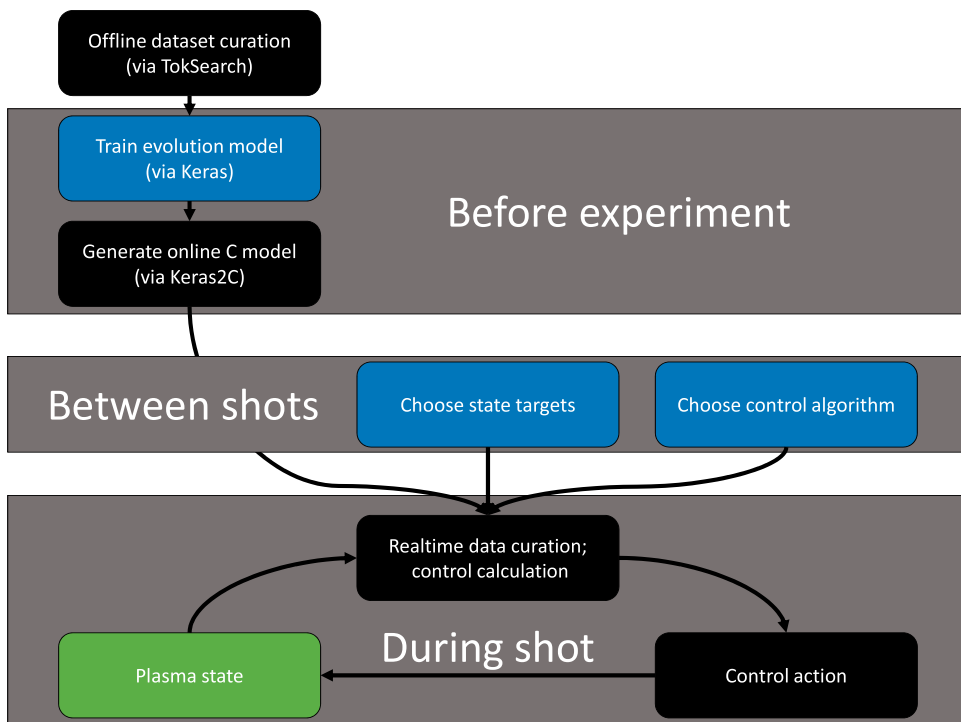


FIGURE 1. Diagram of the new infrastructure for machine learning control on DIII-D. Historical data are used to train a machine learning model to predict plasma evolution, which is then converted into PCS code using Keras2C. The model is used in realtime to predict plasma evolution under different actuator options, and the option that results in predictions closest to user-defined targets is used to evolve the plasma state at each timestep. Blue corresponds to the parameters a user should change before experiments and during shots (as we describe in the Initial experimental control test section); black corresponds to infrastructure described in this section that a user generally would not change; and green corresponds to the state, which is defined by the behaviour of the plasma.

We have built a basic infrastructure for implementing model-based, data-driven control of plasma profiles. As shown in [figure 1](#), the infrastructure includes:

- (i) a workflow for generating data both offline and within the PCS of the same form so that models can be trained offline and deployed on the same type of data online;
- (ii) a user-interface in the control system to specify both the value and importance ('weight') of each controlled state; at present, this includes 33 spatial components from each plasma profile;
- (iii) a code for converting neural networks written with the Python deep learning library Keras (Chollet 2015) into realtime-compatible C code to be used in PCSs, called Keras2C (Conlin *et al.* 2021).

These basic components comprise a code infrastructure within the DIII-D PCS modular enough to easily train a plasma evolution model, and swap in any control algorithm that uses such a model. Each of the components is described in this section.

	Name	Offline source	Online source	Units
Profiles	$n_e$	Zipfit	mtanh	$10^{19} \text{ m}^{-3}$
	$T_e$	Zipfit	mtanh	keV
	Rotation	Zipfit	spline	kHz
	$\iota = 1/q$	EFITRT2	EFITRT2	—
	Pressure	EFITRT2	EFITRT2	Pa
Actuators	Injected power	pinj	VEP	MW
	Injected torque	tinj	VEP	N m
	Target current	iptipp	iptipp	A
	Target density	dstdenp	dstdenp	$10^{19} \text{ m}^{-3}$
	Line average density	dssdenest	dssdenest	$10^{19} \text{ m}^{-3}$

TABLE 1. Signals available both in realtime and offline for training control models and algorithms. ‘Source’ denotes the MDS+ pointname or algorithm used to generate the data. `iptipp` is the plasma current setpoint. `dstdenp` is the interferometer average density target setpoint. VEP is a subcontroller that sets individual beam powers to achieve the algorithm’s requested `pinj` and `tinj`.

### 2.1. Dataset curation

The MDSplus accessing tool TokSearch (Sammuli *et al.* 2018) is leveraged to generate an offline database of timetraces from previous experiments (about 150 000 samples from about 3500 shots run between 2013 and 2018). Before the experiment, one can use this offline database to train a plasma evolution model and Keras2C to convert it into realtime-capable form. A pipeline written in C within the DIII-D PCS interface is used to generate similar samples ‘online’ for use in control. Table 1 summarises the signal sources used for online deployment versus offline training. As described in more detail in Abbate *et al.* (2021), Zipfit is a standard DIII-D workflow that tries a variety of fitting mechanisms and chooses the option with the lowest  $\chi^2$  values; and EFITRT2 is the realtime, kinetically-and-MSE (motional Stark effect) constrained version of the Grad–Shafranov solver EFIT (Lao *et al.* 2005). As is standard for many transport applications on DIII-D, Thomson scattering is used to infer the electron temperature and density; charge exchange recombination (CER) infers the ion temperature and rotation;  $\dot{B}$  probes, MSE, and pressure fits via Thomson scattering are primarily used for the equilibrium reconstruction; and an interferometer is used for the line average density. Note that while the safety factor  $q$  is more common in tokamak physics for describing the ratio of poloidal to toroidal magnetic field, due to the singularity at the separatrix, the rotational transform  $\iota = 1/q$  was used in order to avoid numerical difficulties. The following subsection describes the profile-fitting mechanisms in more detail.

Note that `iptipp` (the plasma current setpoint) and `dstdenp` (the interferometer average density target setpoint) are the exact commands used within a simple PI controller on the tokamak to set a desired current/density by adjusting the solenoid current ramp rate/gas valve flow (voltage). The offline database uses the true values of `pinj` and `tinj` sent into the device; the online controller instead takes in the values requested by the variable energy and perveance (VEP) algorithm. For control, energy and perveance are kept fixed, and VEP merely varies the duty cycle of the beams. VEP coordinates with each of the eight beams to hit the specified total injected power and torque, given other constraints on the beams (Boyer *et al.* 2019). For example, in the control experiment

described later in this paper one of the beams is configured to remain ‘on’ in order to acquire MSE data; and two other beams to stay on to acquire CER data.

For more details on the data excluded, time-averaging, and smoothing used for the offline model, see Abbate *et al.* (2021).

### 2.1.1. Realtime profile fitting

The profiles provided by rtEFIT are already fit as smooth functions of the normalised poloidal flux  $\psi$ . The CER (measuring rotation) and Thomson (measuring electron temperature and density) data are given as point measurements at fixed locations in the lab frame. This requires fitting the data to obtain the values at fixed locations in magnetic coordinates. To accomplish this, a realtime fitting algorithm is used. The mapping between the lab frame and flux coordinates is provided by rtEFIT, which calculates the poloidal flux on a grid in  $(R, Z)$  coordinates. These mapped points are then fit. A modified hyperbolic tangent with a linear core, described in Groebner *et al.* (2001), is used for temperature and density profiles. Smoothing splines (Wand 2000) are used for the rotation profile.

The modified hyperbolic tangent basis was chosen to accurately capture the pedestal region at the expense of slightly less accuracy in the core, where the profile is constrained to be linear. A future upgrade will extend this to higher-order polynomials in the core. Smoothing splines

$$\hat{f}^* = \arg \min_{\hat{f}} \sum_{i=1}^N (y_i - \hat{f}(x_i))^2 + \lambda \int \hat{f}''(x)^2 dx \quad (2.1)$$

were chosen for the rotation profile as they are a flexible basis capable of fitting a wide range of shapes. In contrast to standard interpolating splines, smoothing splines include a term to penalise the curvature of the resulting function, so in general they do not pass directly through the function values, which is desired when working with realtime measurements that may be noisy.

These fitting techniques are placed in a separate algorithm, and are part of an ongoing effort to develop a realtime version of the Consistent and Automatic Kinetic EFIT (CAKE) code (Xing *et al.* 2021).

### 2.1.2. Buffering of fitted points

Each signal is obtained on its own sampling time, generally 1–10 ms between measurements. Values are averaged in time in a boxcar fashion over 50 ms windows, a window long enough to ensure multiple datapoints lie in the window but short enough that the plasma does not change too significantly (the energy confinement time at DIII-D is 100 s of ms).

## 2.2. Plasma control system user interface

Between shots during an experiment, the user can input a (potentially time-varying) set of desired profiles within the PCS: electron temperature, electron density, plasma rotation,  $\iota = 1/q$  and total pressure. The input at present is 33 evenly spaced points between  $\psi_n = 0$  and  $\psi_n = 1$ , for  $\psi_n$  the normalised poloidal flux coordinate. The user also can input time-varying weights corresponding to each of the points within each of the profiles. Control algorithms within the framework are built to minimise the weighted least-squares error between the user-specified targets and the actual profiles:

$$\text{Error}(\mathbf{y}_1, \dots, \mathbf{y}_{\text{Num profiles}}) = \sum_{i=1}^{\text{Num profiles}} |\mathbf{w}_i \cdot (\mathbf{y}_i - \mathbf{y}_i^{\text{target}})|^2 \quad (2.2)$$

for  $\mathbf{w}_i$  the vectors of user-specified weights for each profile,  $\mathbf{y}_i^{\text{target}}$  the values of the individual profiles and  $\mathbf{y}_i^{\text{target}}$  the user-specified target profiles.

### 2.3. Actuator commands

The controllers used in this framework can be thought of as high-level controllers in a cascaded control scheme. After the high-level controller determines the correct actuator values at the current timestep, they are sent to the respective lower-level controllers which interface directly with the hardware. Neutral beam power and torque requests are sent to VEP as described previously, which calculates the required duty cycle then interfaces directly with the master beam algorithm. Requests for plasma current and average density are sent to their respective PI controllers to set the solenoid current ramp rate/gas valve flow (voltage).

## 3. Initial experimental control test

### 3.1. Control algorithm design

The previous sections discuss a new control system infrastructure for training a data-driven model offline and using it for multiple-profile control online. The PCS infrastructure allows users to specify desired target profiles and weights, and the control algorithm chooses actuator values at each cycle time to minimise the error.

In this section, a simple initial test deployed at DIII-D using this infrastructure is described. The system dynamics are identified via a smaller version of the neural network described in Abbate *et al.* (2021). Figure 2 shows the structure of the model inputs and outputs. The model takes as input the present state defined by the plasma profiles. The model also takes as input the recent history of how a set of scalar parameters has been responding to the actuator commands. This empirically increases the model accuracy, and can be seen as a way of adapting the model to brand-new regimes (e.g. if the model sees the plasma  $\beta$  has stayed constant despite an increasing injecting power, the model can guess that a further increase in injected power may not be effective even if the model has not been trained on data from this specific scenario). Finally, the model takes as input proposals for actuators at each of four timesteps into the future (i.e. to the right of the dashed line). These are the values the control algorithm will be allowed to set. Given all of these inputs (the green and blue rectangles), the neural network predicts the future state (defined by plasma profiles) 200 ms into the future (the black rectangle). The specific signals are listed in table 1.

The model is used to choose actuator values via finite set MPC. As described in Cortes *et al.* (2008), finite set control is one of the simplest options for control given a dynamics model of any form. In this control, at every timestep, the model predicts the future state of the plasma conditioned on each of a finite set of control options (i.e. ‘proposals’). In the implementation of this algorithm, a set of (potentially time-varying) proposals is specified in the user interface in the form listed in table 2. At each cycle within the shot, the neural net is evaluated to predict the resulting profiles from each proposal. The algorithm then selects the proposal that led to the smallest error between the profiles predicted and the target profiles specified by the user. This optimal proposal’s first column (i.e. the actuator value for the first timestep in the proposal’s trajectory) is used as the actuator setpoint for the upcoming timestep.

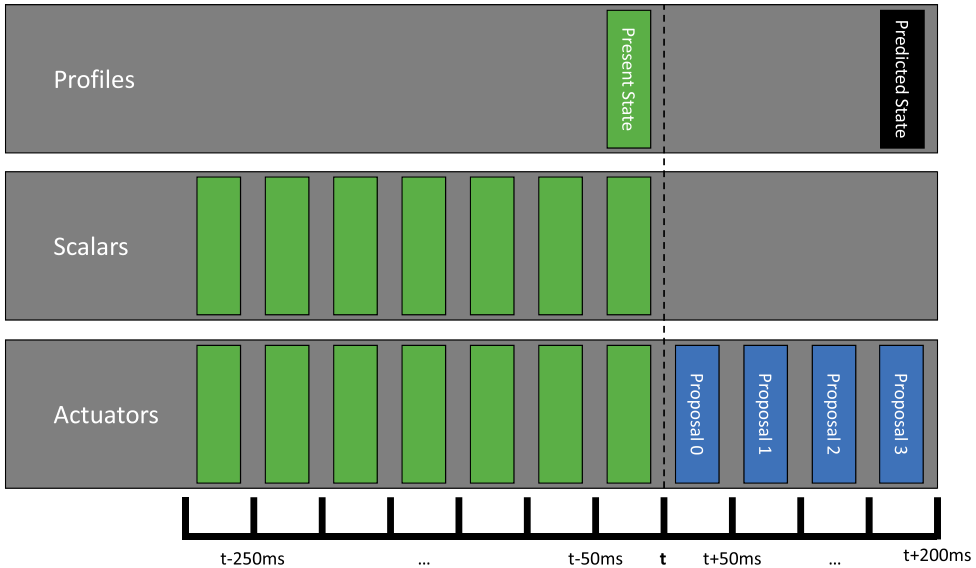


FIGURE 2. Inputs (blue proposals and green information from present and previous timesteps) and output (black profiles 200 ms in the future) of the model. Note that each timestep of data is a boxcar average over all values between the timestep it represents and 50 ms beforehand. Dashed line represents the timestep at which the prediction is made.

Figure 3 demonstrates the mechanism of finite set control for a timeslice in the experiment described in the following section. At this time, the proposals were those in table 2. The targets were the temperature and pressure values at  $\psi_N = 0, 0.03$  and  $0.06$  with weights of 1 (and all other weights 0). In this case, every target point for both pressure and temperature is lower than the corresponding present value by a significant margin. The algorithm predicts that the proposal with increasing  $p_{inj}$  would maintain or increase the pressure and temperature, whereas a decreasing  $p_{inj}$  would bring the values down very close to the targets over the coming 200 ms. Thus, the control algorithm in this case would choose to use the proposal with decreasing  $p_{inj}$ . Note that a full time trace of  $p_{inj}$  is used as input to the model, 300 ms into the past and 200 ms (proposed) into the future. This time-dependent input is shown, along with the input profiles of temperature and pressure. For simplicity, the other inputs (table 1) are not shown in this figure.

### 3.2. Experimental checkout

As part of the initial test, core-profile control is attempted by varying injected power. Throughout the test, three proposals as listed in table 2 are given: a proposal with linearly decreasing injected power ( $p_{inj}$ ), constant  $p_{inj}$  and linearly increasing  $p_{inj}$ . Injected torque ( $t_{inj}$ ) is given a similar ramp to  $p_{inj}$  within each of the three proposals, which should make it easier for the VEP beam controller to properly fulfill the  $p_{inj}$  and  $t_{inj}$  requests (especially because the machine's beams were all pointing in the same direction on the day of the experiment). We control in flattop only, loading in the rampup, shape parameters and other settings from a plasma startup shot in 2020 (182 500). We chose this shot because it had been an experimental test case for the realtime kinetic equilibrium reconstruction. Note that the model was trained on shots from campaign years 2010 through 2018, so that this shot was not seen during training. During the flattop control, all profile targets, when active, have the same shape from a timeslice just after rampup



$\uparrow$ $p_{inj}$				
Actuator	50 ms	100 ms	150 ms	200 ms
$\Delta$ target $p_{inj}$ (MW)	0.31	0.63	0.94	1.26
$\Delta$ target $t_{inj}$ (Nm)	0.27	0.54	0.81	1.08
$\Delta$ target density ( $10^{19} \text{ m}^{-3}$ )	0	0	0	0
$\Delta$ target current (MA)	0	0	0	0
constant				
Actuator	50 ms	100 ms	150 ms	200 ms
$\Delta$ target $p_{inj}$ (MW)	0	0	0	0
$\Delta$ target $t_{inj}$ (Nm)	0	0	0	0
$\Delta$ target density ( $10^{19} \text{ m}^{-3}$ )	0	0	0	0
$\Delta$ target current (MA)	0	0	0	0
$\downarrow$ $p_{inj}$				
Actuator	50 ms	100 ms	150 ms	200 ms
$\Delta$ target $p_{inj}$ (MW)	-0.31	-0.63	-0.94	-1.26
$\Delta$ target $t_{inj}$ (Nm)	-0.27	-0.54	-0.81	-1.08
$\Delta$ target density ( $10^{19} \text{ m}^{-3}$ )	0	0	0	0
$\Delta$ target current (MA)	0	0	0	0

TABLE 2. Set of three ‘proposals’ used to generate the inputs for the model in finite set MPC: desired (signed) changes in  $p_{inj}$ ,  $t_{inj}$ , density and current for each 50 ms window in the 200 ms time horizon over which profiles are predicted. In this case,  $p_{inj}$  and  $t_{inj}$  are proposed to linearly ramp whereas density and current are proposed to stay constant. For each proposal in the set, a model would predict the single-timestep evolution of the plasma state 200 ms into the future. The first column of the proposal with the lowest predicted error 200 ms from now (between the user-specified target and the prediction) is chosen as the control setpoint for the next timestep. These are the actual proposals used for DIII-D shot 187 076.

in the reference shot (182 500 at 1.50 s). All points on the profile are scaled by the same scalar value so that choosing a target is effectively just choosing the core value for each profile.

All experimental results in this section correspond to DIII-D shot 187 076. The following subsections are best read while referencing the annotated experimental trajectory shown in figure 4 which plots timetraces of pressure and temperature at  $\psi_N = 0.03$ .

### 3.2.1. Phase I

The algorithm takes full control of the beams at 1.35 s. In this first phase, the core values (from  $\psi_N = 0$  to 0.06) of pressure and temperature are controlled simultaneously. The same pressure profile from the baseline slice is used, but the temperature profile is scaled from a 3.3 keV core down to a 2.0 keV core (so that effectively the algorithm will attempt to achieve a combination of pressure and temperature outside the range of its training data). Both pressure and temperature (the thick grey timetraces) are initially above the targets (the black dashed lines). However, from the beginning of control at 1.35 s until 1.73 s, an error in the beam set-up meant that MSE data were not available, which in turn meant that the realtime (MSE-constrained) equilibrium fitting code was not returning a pressure and safety factor ( $q$ ) profile. The controller properly floated previous values of pressure and  $q$  in light of the error, but this meant that the controller thought the pressure

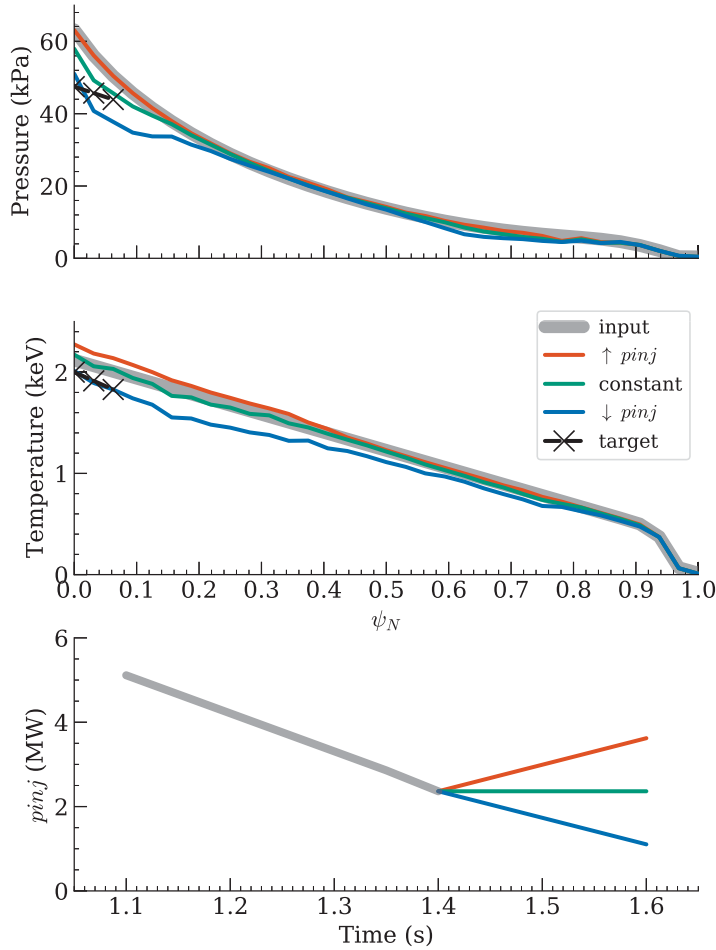


FIGURE 3. Demonstration of the mechanism of finite set control during DIII-D shot 187 076. At this timestep, three points in the core were targeted for both temperature and pressure (weight = 1 for  $\psi_N = 0, 0.03$  and  $0.06$ , and 0 for all other points).  $p_{inj}$  was proposed to linearly decrease, remain constant or linearly increase. The top two panels show the input to the model (greyed out) and the predictions given the three proposals. The black Xs mark the targets, demonstrating that the chosen proposal (that with the lowest mean-squared distance between the target and the prediction) will be the decreasing  $p_{inj}$  proposal. The bottom panel shows the  $p_{inj}$  input to the model: the grey portion is the historical input, and each of the three proposals is included as input to the model for the corresponding predictions on the top plots.

was above the target when really it was below by about 1.6 s. The algorithm therefore continued decreasing beam power for much of this red-shaded region despite the true pressure profile being below the target.

At 1.73 s, rEFIT recovers, and immediately the controller begins to increase  $p_{inj}$ . At 1.97 s, the controller begins to decrease the injected power to settle down on the pressure and temperature targets. At this point, the profiles are still below the targets, so the controller (like a derivative controller) is effectively looking ahead to slow down the upward trajectory as the inputs come in to the final value. At 2 s the phase switches to a new set of targets.

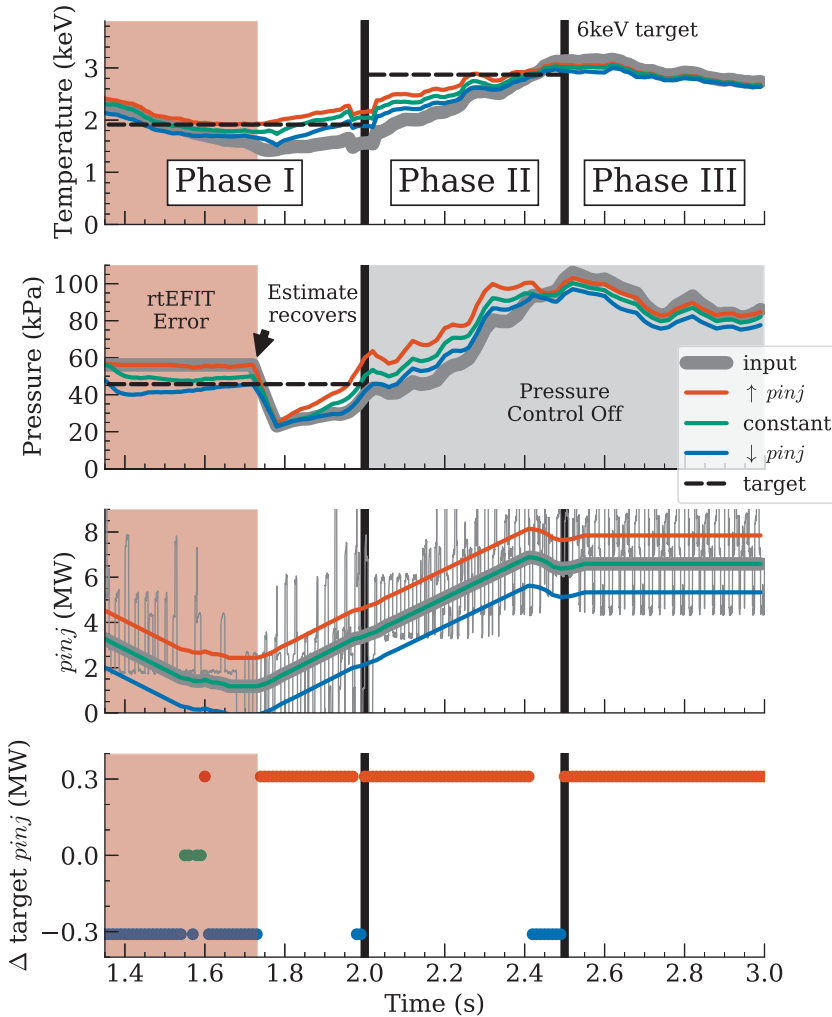


FIGURE 4. DIII-D shot 187076. In the top two panels, true values of pressure and electron temperature at  $\psi_N = 0.03$  are plotted over time in grey. In blue, green and orange the model's prediction 200 ms into the future is tracked proposing a decreasing, constant and increasing injected power, respectively. The bottommost plot shows the change in injected power ( $p_{inj}$ ) requested based on the winning proposal, and the plot above shows the  $p_{inj}$  actually implemented.

### 3.2.2. Phase II

During this next phase, pressure control is turned off and the temperature target is scaled up to have a core value of 3 keV. Once again (likely due to  $p_{inj}$  having been increasing through the previous timesteps), all proposals are predicted to yield an increasing temperature. The controller sensibly chooses to uniformly increase  $p_{inj}$  between 2.0 and 2.42 s, and the temperature increases. A slight overshoot occurs, but (like a proportional controller) the algorithm begins decreasing the injected power right as the temperature passes the target. The value flattens out quite close to the targeted value. For convenience, an enlarged view of the relevant information is shown in figure 5. Note the mechanistic reason for the behaviour: the algorithm correctly predicts that even if beam

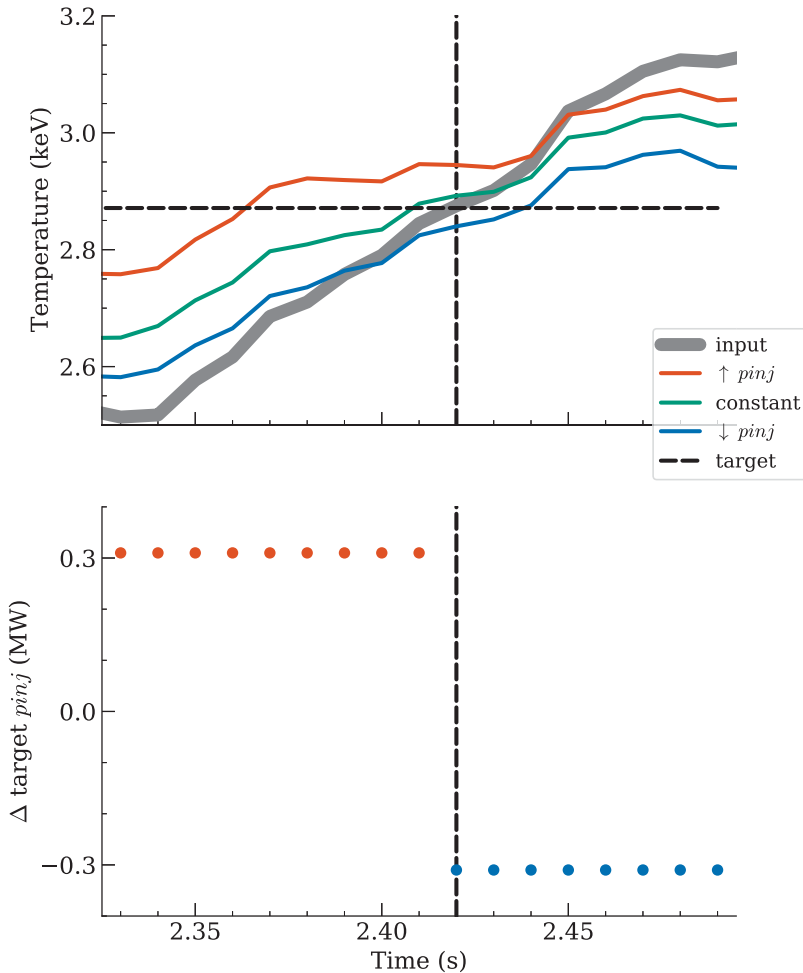


FIGURE 5. Enlarged view of phase II in figure 4, showing the derivative-like action attempting to mitigate overshooting the temperature target by decreasing the injected power right as the target is achieved (at the vertical dashed line).

power is kept constant, the temperature will continue increasing, so that the optimal action to maintain the target is to decrease the beam power right away rather than holding it steady.

### 3.2.3. Phase III

At 2.5 s the final phase of the shot begins: scaling the temperature target profile up to a core value of 6 keV. Towards the end of the previous phase, an  $n = 2$  MHD mode began to grow, and at the beginning of the 6 keV phase, the beams are already firing at maximum power, so the proposals to increase cannot be fulfilled. The temperature therefore hovers around the 3 keV where it had started.

### 3.2.4. Analysis

In summary, the finite set controller is able to perform with some of the beneficial characteristics of a stable proportional–derivative controller without any hand-tuning (though more work would be needed to compare the controllers quantitatively). Although

the phases of the experiment were not long enough to see the full dynamics, we suspect that the controller would have properly settled down close to the target values in the first and second phases. Although the finite set controller is the simplest possible ‘model-predictive’ control algorithm and just a starting point, the benefit of the lookahead to more efficiently achieving a desired state is demonstrated.

Each neural net evaluation takes about a millisecond, and these calculations are performed in serial. The DIII-D control system, like most realtime systems, requires that the algorithm run on a fixed cycle time throughout the shot. To ensure it completes at each cycle with room for error, a cycle time of 10 ms was used.

#### 4. Conclusion and future work

We have built the infrastructure and performed a successful initial test for a new paradigm of fully data-driven, model-based tokamak state control. Moving forward, we plan to use more intelligent control schemes. One idea is to locally linearise the machine learning model that predicts future states so that we have a form  $x(t + 1) = Ax(t) + Bu(t)$  amenable to traditional control techniques. One way to do this is to get a local Jacobian from the machine learning predictor we already have, in which case the state  $x$  would be the true state of the system. Another mechanism is to attempt to encode the true plasma state into a latent state where the dynamics are globally linear. In this case, we would be able to overcome the (generally poor) assumption that the real-state dynamics must be linear in a large enough region around the current state to be useful for model-based control. We have also been developing an offline testbed for simulating controllers via the ASTRA transport modelling suite (Pereverzev & Yushmanov 2002). This will give us the means to evaluate many controllers offline and decide the best few to try during real experiments.

We also at present are using a limited set of actuators (total injected power and torque from beams, target density setpoint and plasma current setpoint). More actuators should be added to the base infrastructure over time (e.g. individual beam power and possibly voltages/perveances, electron cyclotron gyrotron angles and powers, shape coils and 3D coils). More state information may also improve predictive models and, hence, controllability, e.g. high-resolution turbulence information from CO<sub>2</sub> interferometers and beam emission spectroscopy.

#### Acknowledgements

Thanks go to Al Hyatt, Dan Boyer, Jayson Barr, David Eldon and the rest of the DIII-D operations and control team. Thanks also go to Josiah Wai for helpful discussions on MPC. This material is based upon work supported by the US Department of Energy, Office of Science, Office of Fusion Energy Sciences, using the DIII-D National Fusion Facility, a DOE Office of Science user facility, under Awards DE-AC02-09CH11466, DE-SC0021275 and DE-FC02-04ER54698.

*Editor William Dorland thanks the referees for their advice in evaluating this article.*

#### Declaration of interests

The authors report no conflict of interest.

#### Disclaimer

This report is prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal

liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency hereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

## REFERENCES

- ABBATE, J., CONLIN, R. & KOLEMEN, E. 2021 Data-driven profile prediction for DIII-D. *Nucl. Fusion* **61** (4), 046027.
- BARTON, J.E., SHI, W., BESSEGHIR, K., LISTER, J., KRITZ, A., SCHUSTER, E., LUCE, T.C., WALKER, M.L., HUMPHREYS, D.A. & FERRON, J.R. 2013 Physics-based control-oriented modeling of the safety factor profile dynamics in high performance tokamak plasmas. In *52nd IEEE Conference on Decision and Control*, pp. 4182–4187. IEEE.
- BOYER, M.D., BARTON, J., SCHUSTER, E., LUCE, T.C., FERRON, J.R., WALKER, M.L., HUMPHREYS, D.A., PENAFLO, B.G. & JOHNSON, R.D. 2013 First-principles-driven model-based current profile control for the DIII-D tokamak via LQI optimal control. *Plasma Phys. Control. Fusion* **55** (10), 105007.
- BOYER, M.D., ERICKSON, K.G., GRIERSON, B.A., PACE, D.C., SCOVILLE, J.T., RAUCH, J., CROWLEY, B.J., FERRON, J.R., HASKEY, S.R., HUMPHREYS, D.A., *et al.* 2019 Feedback control of stored energy and rotation with variable beam energy and perveance on DIII-D. *Nucl. Fusion* **59** (7), 076004.
- CHOLLET, F. 2015 Keras, GitHub. <https://github.com/fchollet/keras>
- CONLIN, R., ERICKSON, K., ABBATE, J. & KOLEMEN, E. 2021 KERAS2C: a library for converting KERAS neural networks to real-time compatible C. *Eng. Appl. Artif. Intell.* **100**, 104182.
- CORTES, P., KAZMIERKOWSKI, M.P., KENNEL, R.M., QUEVEDO, D.E. & RODRIGUEZ, J. 2008 Predictive control in power electronics and drives. *IEEE Trans. Ind. Electron.* **55** (12), 4312–4324.
- FELICI, F., CITRIN, J., TEPLUKHINA, A.A., REDONDO, J., BOURDELLE, C., IMBEAUX, F., SAUTER, O., JET CONTRIBUTORS & THE EUROFUSION MST1 TEAM 2018 Real-time-capable prediction of temperature and density profiles in a tokamak using RAPTOR and a first-principle-based transport model. *Nucl. Fusion* **58** (9), 096006.
- FELICI, F. & SAUTER, O. 2012 Non-linear model-based optimization of actuator trajectories for tokamak plasma profile control. *Plasma Phys. Control. Fusion* **54** (2), 025002.
- FELICI, F., SAUTER, O., CODA, S., DUVAL, B.P., GOODMAN, T.P., MORET, J.-M., PALEY, J.I. & THE TCV TEAM 2011 Real-time physics-model-based simulation of the current density profile in tokamak plasmas. *Nucl. Fusion* **51** (8), 083052.
- FERRON, J.R., GOHIL, P., GREENFIELD, C.M., LOHR, J., LUCE, T.C., MAKOWSKI, M.A., MAZON, D., MURAKAMI, M., PETTY, C.C., POLITZER, P.A., *et al.* 2006 Feedback control of the safety factor profile evolution during formation of an advanced tokamak discharge. *Nucl. Fusion* **46** (10), L13–L17.
- FUJITA, T. & THE JT-60 TEAM 2006 Steady state operation research in JT60u with extended pulse length. *Nucl. Fusion* **46** (3), S3–S12.
- FU, Y., ELTON, D., ERICKSON, K., KLEIJWEGT, K., LUPIN-JIMENEZ, L., BOYER, M.D., EIDIETIS, N., BARBOUR, N., IZACARD, O. & KOLEMEN, E. 2020 Machine learning control for disruption and tearing mode avoidance. *Phys. Plasmas* **27**, 22501.
- GROEBNER, R.J., BAKER, D.R., BURRELL, K.H., CARLSTROM, T.N., FERRON, J.R., GOHIL, P., LAO, L.L., OSBORNE, T.H., THOMAS, D.M., WEST, W.P., *et al.* 2001 Progress in quantifying the edge physics of the H mode regime in DIII-D. *Nucl. Fusion* **41** (12), 1789–1802.
- HOLCOMB, C.T., HEIDBRINK, W.W., FERRON, J.R., VAN ZEELAND, M.A., GAROFALO, A.M., SOLOMON, W.M., GONG, X., MUELLER, D., GRIERSON, B., BASS, E.M., *et al.* 2015 Fast-ion transport in  $q_{min} < 2$ , high-beta steady-state scenarios on DIII-D. *Phys. Plasmas* **22** (5), 055904.

- HUMPHREYS, D., AMBROSINO, G., DE VRIES, P., FELICI, F., KIM, S.H., JACKSON, G., KALLENBACH, A., KOLEMEN, E., LISTER, J., MOREAU, D., *et al.* 2015 Novel aspects of plasma control in ITER. *Phys. Plasmas* **22** (2), 021806.
- ITER PHYSICS EXPERT GROUP ON CONFINEMENT AND TRANSPORT 1999 Chapter 2: Plasma confinement and transport. *Nucl. Fusion* **39**, 2175.
- JALALVAND, A., ABBATE, J., CONLIN, R., VERDOOLAEGE, G. & KOLEMEN, E. 2021 Real-time and adaptive reservoir computing with application to profile prediction in fusion plasma. *IEEE Trans. Neural Netw. Learn. Syst.* **33** (6), 2630–2641.
- LABORDE, L., MAZON, D., MOREAU, D., MURARI, A., FELTON, R., ZABEO, L., ALBANESE, R., ARIOLA, M., BUCALOSSO, J., CRISANTI, F., *et al.* 2005 A model-based technique for integrated real-time profile control in the JET tokamak. *Plasma Phys. Control. Fusion* **47** (1), 155–183.
- LAO, L.L., ST JOHN, H.E., PENG, Q., FERRON, J.R., STRAIT, E.J., TAYLOR, T.S., MEYER, W.H., ZHANG, C. & YOU, K.I. 2005 MHD equilibrium reconstruction in the DIII-D tokamak. *Fusion Sci. Technol.* **48**, 968–977.
- LUXON, J.L. & DAVIS, L.G. 1985 Big dee - a flexible facility operating near breakeven conditions. *Fusion Technol.* **8** (1), 441–449.
- MALJAARS, E., FELICI, F., BLANKEN, T.C., GALPERTI, C., SAUTER, O., DE BAAR, M.R., CARPANESE, F., GOODMAN, T.P., KIM, D., KIM, S.H., *et al.* 2017 Profile control simulations and experiments on TCV: a controller test environment and results using a model-based predictive controller. *Nucl. Fusion* **57** (12), 126063.
- MALJAARS, E., FELICI, F., DE BAAR, M.R., VAN DONGEN, J., HOGEWELJ, G.M.D., GEELEN, P.J.M. & STEINBUCH, M. 2015 Control of the tokamak safety factor profile with time-varying constraints using MPC. *Nucl. Fusion* **55** (2), 23001.
- MONTES, K.J., REA, C., GRANETZ, R.S., TINGUELY, R.A., EIDIETIS, N., MENEGHINI, O.M., CHEN, D.L., SHEN, B., XIAO, B.J., ERICKSON, K., *et al.* 2019 Machine learning for disruption warnings on Alcator C-Mod, DIII-D, and EAST. *Nuclear Fusion* **59** (9), 096015.
- MOREAU, D., MAZON, D., WALKER, M.L., FERRON, J.R., BURRELL, K.H., FLANAGAN, S.M., GOHIL, P., GROEBNER, R.J., HYATT, A.W., LA HAYE, R.J., *et al.* 2011 Plasma models for real-time control of advanced tokamak scenarios. *Nucl. Fusion* **51** (6), 063009.
- PENAFLO, B.G., FERRON, J.R., WALKER, M.L., PIGLOWSKI, D.A. & JOHNSON, R.D. 2001 Real-time control of DIII-D plasma discharges using a linux alpha computing cluster. *Fusion Eng. Des.* **56–57**, 739–742.
- PEREVERZEV, G.V. & YUSHMANOV, P.N. 2002 *ASTRA Automated System for TRansport Analysis*. IPP.
- REA, C., MONTES, K.J., ERICKSON, K.G., GRANETZ, R.S. & TINGUELY, R.A. 2019 A real-time machine learning-based disruption predictor in DIII-D. *Nucl. Fusion* **59** (9), 096016.
- SAMMULI, B.S., BARR, J.L., EIDIETIS, N.W., OLOFSSON, K.E.J., FLANAGAN, S.M., KOSTUK, M. & HUMPHREYS, D.A. 2018 TokSearch: a search engine for fusion experimental data. *Fusion Eng. Des.* **129**, 12–15.
- TEPLUKHINA, A.A., SAUTER, O., FELICI, F., MERLE, A., KIM, D., THE TCV TEAM, THE ASDEX UPGRADE TEAM & THE EUROFUSION MST1 TEAM 2017 Simulation of profile evolution from ramp-up to ramp-down and optimization of tokamak plasma termination with the RAPTOR code. *Plasma Phys. Control. Fusion* **59** (12), 124004.
- WALKER, M.L., DE VRIES, P., FELICI, F. & SCHUSTER, E. 2020 Introduction to tokamak plasma control. In *2020 American Control Conference*, pp. 2901–2918. IEEE.
- WAND, M.P. 2000 A comparison of regression spline smoothing procedures. *Comput. Stat.* **15** (4), 443–462.
- WEHNER, W., LAURET, M., SCHUSTER, E., FERRON, J.R., HOLCOMB, C., LUCE, T.C., HUMPHREYS, D.A., WALKER, M.L., PENAFLO, B.G. & JOHNSON, R.D. 2016 Predictive control of the tokamak q profile to facilitate reproducibility of high-qmin steady-state scenarios at DIII-D. In *2016 IEEE Conference on Control Applications*, pp. 629–634. IEEE.
- WIJNANDS, T., VAN HOUTTE, D., MARTIN, G., LITAUDON, X. & FROISSARD, P. 1997 Feedback control of the current profile on Tore Supra. *Nucl. Fusion* **37** (6), 777–791.
- XING, Z.A., ELTON, D., NELSON, A.O., ROELOFS, M.A., EGGERT, W.J., IZACARD, O., GLASSER, A.S., LOGAN, N.C., MENEGHINI, O., SMITH, S.P., *et al.* 2021 CAKE: consistent automatic kinetic equilibrium reconstruction. *Fusion Eng. Des.* **163**, 112163.