

Magnetic Control of Tokamak Plasmas Through Deep Reinforcement Learning

Jonas Degraeve^{*,1}, Federico Felici^{*,2,m}, Jonas Buchli^{*,1,m}, Michael Neunert^{*,1}, Brendan Tracey^{*,1,m}, Francesco Carpanese^{*,1,2}, Timo Ewalds^{*,1}, Roland Hafner^{*,1}, Abbas Abdolmaleki¹, Diego de las Casas¹, Craig Donner¹, Leslie Fritz¹, Cristian Galperti², Andrea Huber¹, James Keeling¹, Maria Tsimpoukelli¹, Jackie Kay¹, Antoine Merle², Jean-Marc Moret², Seb Noury¹, Federico Pesamosca², David Pfau¹, Olivier Sauter², Cristian Sommariva², Stefano Coda², Basil Duval², Ambrogio Fasoli², Pushmeet Kohli¹, Koray Kavukcuoglu¹, Demis Hassabis¹ and Martin Riedmiller^{*,1}

^{*}Equal contributions, ¹DeepMind, ²SPC-EPFL, Lausanne, Switzerland, ^mCorresponding authors: btracey@deepmind.com, buchli@deepmind.com, federico.felici@epfl.ch

Nuclear fusion using magnetic confinement, in particular in the tokamak configuration, is a promising path towards sustainable energy. A core challenge is to shape and maintain a high temperature plasma within the tokamak vessel. This requires high dimensional, high frequency, closed-loop control using magnetic coils as actuators, made even more demanding by the diverse requirements across a wide range of plasma configurations. In this work, we introduce a novel architecture for tokamak magnetic controller design that autonomously learns to command the full set of control coils. This architecture can meet control objectives specified at a high level, while satisfying physical and operational constraints. This approach has unprecedented flexibility and generality in problem specification and yields a significant reduction in design effort to produce new plasma configurations. We use this to successfully produce and control a diverse set of plasma configurations on the Tokamak à Configuration Variable (TCV) [1, 2], including elongated, conventional shapes as foreseen for ITER, as well as advanced configurations, such as negative triangularity and “snowflake” configurations. Our approach achieves accurate tracking of the location, current, and shape for these configurations. We additionally demonstrate the first ever sustained “droplets” on TCV where two separate plasmas are maintained simultaneously within the vessel. This represents the first use of reinforcement learning for feedback control on a tokamak, showing potential to accelerate research in the fusion domain, and is one of the most challenging real-world systems to which reinforcement learning has been applied.

¹ *Keywords: Tokamak Plasma Physics, Deep Reinforcement Learning, Control Engineering*

² Tokamaks are torus-shaped devices for nuclear fusion research, and are a leading candidate for the
³ generation of sustainable electric power. A major direction of research is to study the effects of shaping
⁴ the distribution of the plasma into different configurations [3–5] to optimize the stability, confinement
⁵ and energy exhaust, and in particular to inform the first burning plasma experiment, ITER. Confining
⁶ each configuration within the tokamak requires designing a feedback controller that can manipulate
⁷ the magnetic field [6] through precise control of multiple coils which are magnetically coupled to the
⁸ plasma to achieve the desired plasma current, position and shape, a problem known as the *tokamak*
⁹ *magnetic control problem*.

¹⁰ The conventional approach to this time-varying, nonlinear, multivariate control problem is to first
¹¹ solve an inverse problem to precompute a set of feedforward coil currents and voltages [7, 8]. Then,
¹² a set of independent single-input single-output PID controllers are designed to stabilize the plasma
¹³ vertical position, control the radial position, and plasma current, all of which must be designed to not
¹⁴ mutually interfere [6]. Most control architectures are further augmented by an outer control loop for
¹⁵ the plasma shape, which involves implementing a real-time estimate of the plasma equilibrium [9,

10] to modulate the feedforward coil-currents [8]. The controllers are designed based on linearized model dynamics, and gain scheduling is required to track time-varying control targets. While these controllers are usually effective, they require substantial engineering effort, design effort, and expertise whenever the target plasma configuration is changed, together with complex real-time calculations for equilibrium estimation.

A radically new approach to controller design is made possible by employing reinforcement learning (RL) to generate nonlinear feedback controllers. The RL approach, already used successfully in a number of challenging applications in other domains [11–13], enables intuitive setting of performance objectives, shifting the focus towards what should be achieved, rather than how. Furthermore, RL greatly simplifies the control system. A single computationally inexpensive controller replaces the nested control architecture, and an internalized state reconstruction removes the requirement for independent equilibrium construction. These combined benefits reduce the controller development cycle and accelerate the study of alternative plasma configurations. Indeed, artificial intelligence (AI) has recently been identified as a “Priority Research Opportunity” for fusion control [14], building on demonstrated successes in reconstructing plasma shape parameters [15, 16], accelerating simulations using surrogate models [17, 18], and detecting impending plasma disruptions [19]. RL has not, however, been used for magnetic controller design, which is challenging due to high dimensional measurements and actuation, long time horizons, rapid instability growth rates, and the need to infer the plasma shape through indirect measurements.

In this work, we present the first experimental application of an RL-designed magnetic controller in a tokamak. The control policies are learned solely through interaction with a tokamak simulator, and are shown to be directly capable of tokamak magnetic control on hardware, successfully bridging the “sim-to-real” gap. This enables a fundamental shift from engineering-driven control of a pre-designed state to AI-driven optimization of objectives specified by an operator. We demonstrate the effectiveness of our controllers in experiments carried out on the Tokamak à Configuration Variable (TCV) [1, 2], where we demonstrate control of a variety of plasma shapes, including elongated ones, such as those foreseen in ITER, as well as advanced configurations such as negative triangularity and “snowflake” plasmas. Additionally, we demonstrate the first ever sustained configuration where two separate plasma “droplets” are simultaneously maintained within the vessel. Tokamak magnetic control is one of the most complex real-world systems to which reinforcement learning has been applied. This is a promising new direction for plasma controller design, with the potential to accelerate fusion science, explore novel configurations, and aid in future tokamak development.

Deep Learning Control and Training Architecture

Our architecture, depicted in Fig. 1, is a flexible approach for designing tokamak magnetic confinement controllers. The approach has three main phases. First, a designer specifies objectives for the experiment, potentially accompanied by time-varying control targets. Second, a deep RL algorithm interacts with a tokamak simulator to find an approximately optimal control policy to meet the specified goals. Third, the control policy, represented as a neural network, is run directly (“zero-shot”) on tokamak hardware in real time.

In the first phase, a set of objectives is specified to meet the experimental goals. A wide variety of desired properties can be specified in the objective set (Extended Data Table 3), ranging from basic stabilization of position and plasma current to sophisticated combinations of multiple time-varying targets including a precise shape outline with specified elongation, triangularity, and X-point location. These objectives are then combined into a “reward function” that assigns a scalar quality measure to the state at each time step. This function can also penalize the control policy for reaching undesired

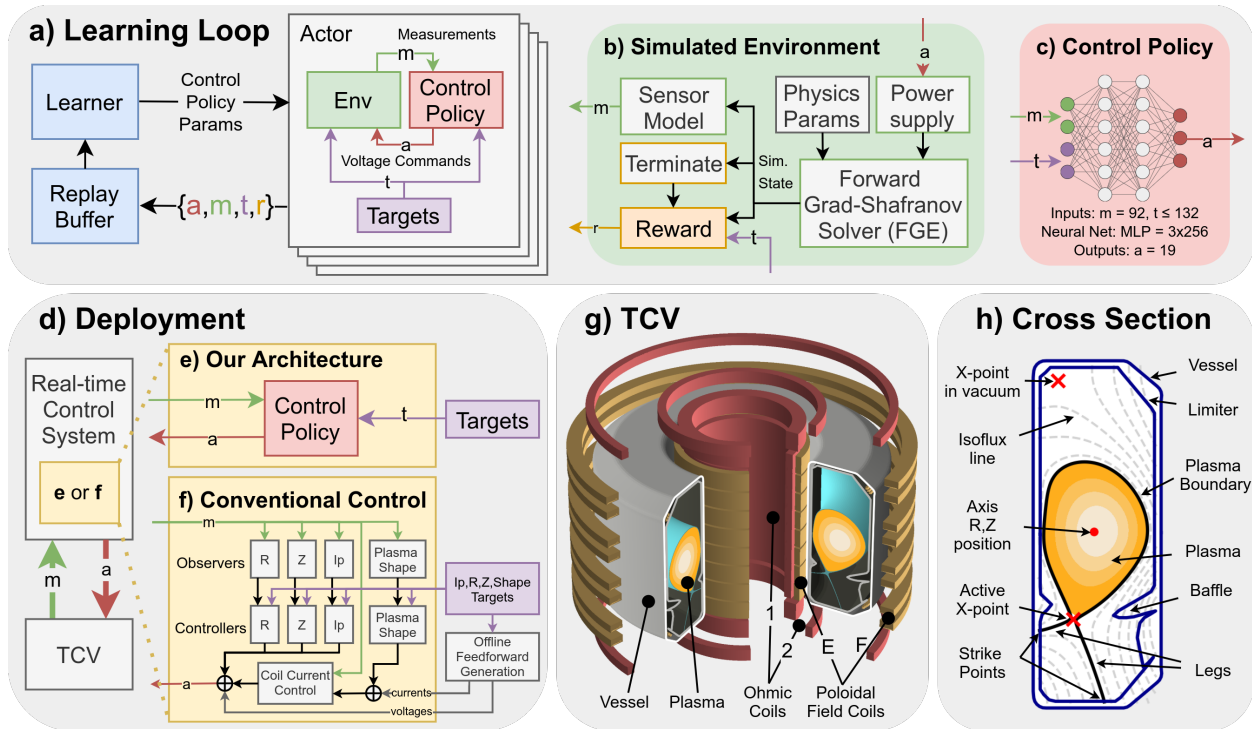


Fig. 1 | Representation of the components of our controller design architecture.

a) Depiction of the learning loop. The controller sends voltage commands based on the current plasma state and control targets. This data is sent to the replay buffer, which feeds data to the learner to update the policy.

b) Our environment interaction loop, consisting of a power supply model, sensing model, environment physical parameter variation, and reward computation.

c) Our control policy is an MLP with three hidden layers that takes measurements and control targets and outputs voltage commands.

d) The interaction of TCV and the real-time deployed control system implemented using either (f) a conventional controller composed of many sub-components or (e) our architecture using a single deep neural network to control all 19 coils directly.

g) A depiction of TCV and the 19 actuated coils. The vessel is 1.5 m high, minor radius 0.88 m, vessel half-width 0.26 m.

h) A cross section of the vessel and plasma, with the important aspects labelled.

terminal states, as discussed below. Crucially, a well-designed reward function will be minimally specified, giving the learning algorithm maximum flexibility to find the best way to attain the desired outcome.

In the second phase, a high performance RL algorithm collects data through interaction with an environment, as depicted in Fig. 1a and Fig. 1b. We employ a simulator that has enough physical fidelity to describe the evolution of plasma shape and current, while remaining sufficiently computationally cheap for learning. Specifically, we model the dynamics governing the evolution of the plasma state under the influence of the poloidal field coil voltages using a free-boundary plasma evolution model [20]. In this model, the currents in the coils and passive conductors evolve under influence of externally applied voltages from the power supplies as well as induced voltages from time-varying currents in other conductors and in the plasma itself. The plasma is in turn modelled by the Grad-Shafranov equation [21], which results from the balance between the Lorentz force and the pressure gradient inside the plasma on the time scales of interest. The evolution of total plasma current I_p is modeled using a lumped circuit equation. This set of equations is solved numerically by the FGE software package [22].

An RL algorithm uses simulator data to find a near-optimal policy with respect to the specified reward function. The data rate of our simulator is significantly slower than that of a typical RL environment due to the computational requirements of evolving the plasma state. We overcome the paucity of data by optimizing the policy using Maximum a Posteriori policy Optimization (MPO) [23], a recently developed actor-critic algorithm. MPO supports data collection across distributed parallel streams, and is known to efficiently learn from collected data. We additionally exploit the asymmetry inherent to MPO’s actor-critic design to overcome the constraints of magnetic control. In actor-critic algorithms, the “critic” learns the discounted expected future reward for various actions using the available data, and the “actor” uses the critic’s predictions to set the control policy. The representation of the actor’s control policy is restricted as it must run on TCV with real-time guarantees, while, crucially, the critic is unrestricted as it is only used during training (in the learner). We thus use a fast (4-layer) feedforward neural network in the actor, Fig. 1c, and a much larger recurrent neural network in the critic. This asymmetry enables the critic to infer the underlying state from measurements, deal with complex state transition dynamics over different time-scales, and assess the influence of system measurement and action delays. The information from the coupled dynamics is then distilled into a real-time capable controller.

In the third phase, the control policy is bundled with the associated experiment control targets into an executable using a compiler tailored toward real-time control at 10 kHz that minimizes dependencies and eliminates unnecessary computations. This executable is loaded as a block within the TCV control framework [24] (Fig. 1d). Each experiment begins with standard plasma formation procedures, where a traditional controller maintains the plasma’s location and total current. At a prespecified time, termed the “handover”, control is switched directly to our control policy which then actuates the 19 TCV control coils to transform the plasma shape and current to the desired targets. Experiments are executed without further tuning of the control policy network weights after training, in other words, there is “zero-shot” transfer from simulation to hardware.

The deployed control policies reliably transfer onto TCV through several key attributes of the learning procedure, depicted in Fig. 1b. We identified an actuator and sensor model that incorporates properties impacting control stability such as delays, measurement noise, and control voltage offsets. We applied targeted parameter variation during training across an appropriate range for the plasma pressure, current density profile and plasma resistivity through analysis of experiment data, to account for varying, uncontrolled, experimental conditions. This provides robustness while ensuring good performance. While the simulator is generally accurate, there are known regions where the dynamics

are known to be poorly represented. We built “learned region avoidance” into the training loop to avoid these regimes through the use of rewards and termination conditions (Extended Data Table 6), which halt the simulation when specified conditions are encountered. Termination conditions are also used to enforce operational limits. The control policies learn to stay within the specified limits, for example on maximum coil current or the edge safety factor [25].

The controllers designed by our architecture are structurally significantly simplified compared with conventional designs, as depicted in Fig. 1e and Fig. 1f. Instead of a series of controllers, RL driven design creates a single nonlinear multiple-input multiple-output network controller.

Results

We demonstrate the capability of our architecture on a wide variety of control targets in real-world experiments on TCV. We first show accurate control of the fundamental qualities of plasma equilibria. We then control a wide range of equilibria with complex, time-varying objectives and physically relevant plasma configurations. Finally, we demonstrate first-of-its-kind control of a configuration with multiple plasma “droplets” in the vessel simultaneously.

Fundamental Capability Demonstration

We first test the fundamental tasks of plasma control through a series of changes representative of those required for a full plasma discharge. First, from the handover at 0.0872 s, take over and stabilize I_p at -110 kA. Next, ramp the plasma current to -150 kA, and then elongate the plasma from 1.24 to 1.44, thereby increasing the vertical instability growth rate to 150 Hz. Next, demonstrate position control through shifting the vertical plasma position by 10 cm, and then divert the plasma with control of the active X-point location (see Fig. 1h). Finally, return the plasma to the handover condition, and ramp down I_p to -70 kA to shut down safely. While, in general, accuracy requirements will depend on the exact experiment, a reasonable aim is to control I_p to within 5 kA (3 % of the final 150 kA target) and the shape to within 2 cm (8 % of the vessel radial half-width of 26 cm).

The performance of the control policy is depicted in Fig. 2. All tasks are performed successfully, with a tracking accuracy well below the desired thresholds. In the initial limited phase (0.1 s to 0.45 s), the I_p RMSE is 0.71 kA (0.59 % of the target) and the shape RMSE is 0.78 cm (3 % of the vessel half-width). In the diverted phase (0.55 s to 0.8 s), I_p and shape RMSE are 0.28 kA and 0.53 cm respectively (0.2 % and 2.1 %), yielding RMSE across the full window (0.1 s to 1.0 s) of 0.62 kA and 0.75 cm (0.47 % and 2.9 %). This demonstrates that our RL architecture is capable of accurate plasma control across all relevant phases of a discharge experiment.

Control Demonstrations

We next demonstrate the capability of our architecture to produce complex configurations for scientific study. Each demonstration has its own specific time-varying targets but, otherwise, uses the same architectural setup to generate a control policy, including the training and environment configuration, with only minor adjustments to the reward function (shown in Extended Data Table 2). Recall that, in each experiment, the plasma has low elongation before the handover, and the control policy actively modulates the plasma to the configuration of interest. Selected time slices from these experiments are shown in Fig. 3 with additional detail in Extended Data Fig. 6.

Elongated plasmas are desirable for their improved thermal confinement properties, but are difficult to control due to an increased vertical instability growth rate. We targeted a high elongation of 1.9 with

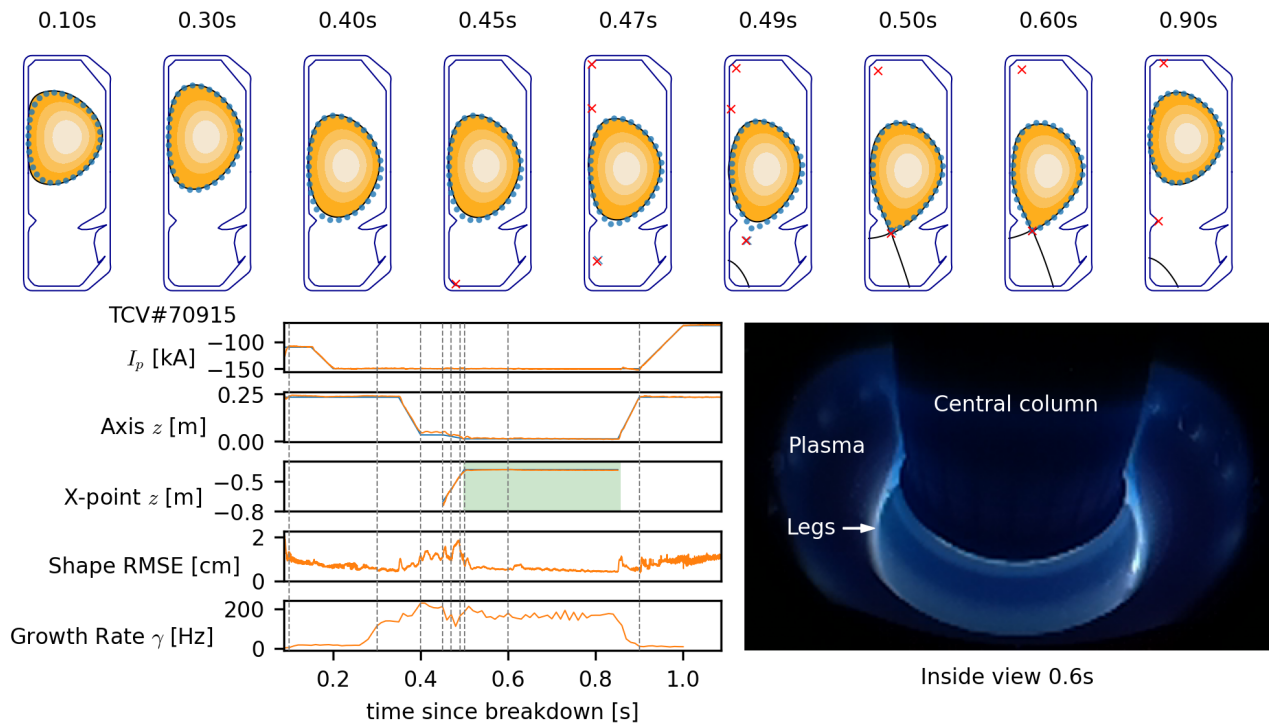


Fig. 2 | Demonstration of plasma current, vertical stability, position and shape control. *Top*: Target shape points (green circles 2 cm radius) compared to (post-experiment) equilibrium reconstruction (black continuous line in contour plot). *Bottom Left*: Target time traces (blue traces) compared to reconstructed observation (orange traces). *Right*: Picture inside the vessel at 0.6 s showing the diverted plasma with its legs.

a considerable growth rate. The controller was able to produce and stabilize this elongation, as shown in Fig. 3a. We obtain a good match between the targeted and desired elongation, with an RMSE of 0.018 between 0.55 s and 1.0 s. We also control shape and plasma current to their target values, with I_p RMSE of 1.2 kA and shape RMSE of 1.6 cm. This demonstrates the capability to stabilize a high vertical instability growth rate of over 1.4 kHz, despite acting at only 10 kHz (including an in-vessel coil).

We next test applying auxiliary heating through neutral beam injection to enter “H-mode”, which is desirable for energy production (having higher energy confinement time) but causes significant changes to the plasma properties. We were provided a time-varying trajectory based on the proposed ITER configuration that uses such auxiliary heating. As the normalized pressure β_p increases to 1.12, seen in Fig. 3b, the plasma position and current are maintained accurately, with I_p RMSE of 2.6 kA and shape RMSE of 1.34 cm between 0.1 s and 1.0 s. This shows our controller can robustly adapt to a changing plasma state, and can successfully work with high performance heated H-mode plasma under externally-specified configurations.

Negative triangularity plasmas are attractive as they have favourable confinement properties without the strong edge pressure gradient typical of H-modes. We targeted a diverted configuration with triangularity of -0.8 , and with X-points at both corners. We successfully achieve this configuration, shown in Fig. 3c. The triangularity is accurately matched, with an RMSE of 0.067 between 0.5 s and 0.9 s, as are the plasma current and shape with RMSE of 3.4 kA and 1.3 cm respectively. This demonstrates the ability to rapidly and directly create a configuration under active study [26].

Snowflake configurations are promising as they distribute the particle exhaust across multiple strike points, and are also actively researched [27, 28]. A crucial parameter is the distance between the two X-points that form the divertor legs. We demonstrate our ability to control this distance, shown in Fig. 3d. The control policy first establishes a snowflake configuration with X-points separated by 34 cm. It then manipulates the far X-point to approach the limiting X-point, ending with a separation of 6.6 cm. The time-varying X-point target is tracked accurately with an RMSE of 4.1 cm between 0.6 s and 1.1 s. The plasma current and shape are maintained to high accuracy during this transition, with RMSE of 0.52 kA and 0.62 cm respectively. This demonstrates accurate control of a complex time-varying target with multiple coupled objectives.

In aggregate, these experiments demonstrate the ease with which new configurations can be explored, prove our architecture’s ability to operate in high-performance discharges, and confirm the breadth of its capability.

Novel Multi-Domain Plasma Demonstration

Lastly, we demonstrate the power of our architecture to explore novel plasma configurations. We test control of “droplets”, a configuration where two separate plasmas exist within the vessel simultaneously. While it is likely possible that existing approaches could stabilize such droplets, significant investment would be required to develop feedforward coil current programming, implement real-time estimators, tune controller gains, and successfully take control after plasma creation. In contrast, with our approach we simply adjust the simulated handover state (to account for the different handover condition from single-axis plasmas), and define a reward function to keep the position of each droplet component steady while ramping up the domain plasma currents. This loose specification gives the architecture the necessary freedom to choose how to best adapt the droplet shapes as I_p increases to maintain stability. The architecture was able to successfully stabilize droplets over the entire 200 ms control window, and individually ramp the current within each domain as shown in Fig. 4. This is the first time droplets have been sustained for such a control window in TCV, highlighting the significant advantage of a general, learning based control architecture with flexibly-defined objectives.

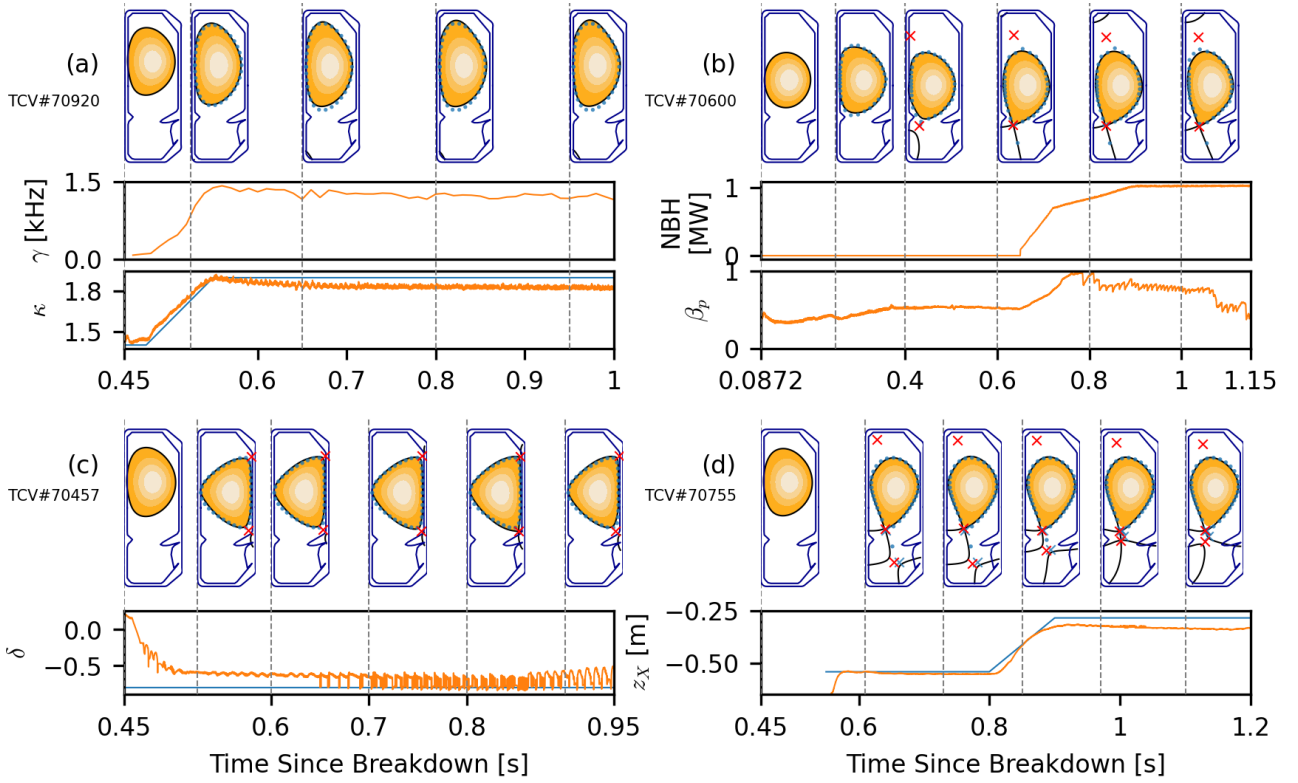


Fig. 3 | Control demonstrations obtained during TCV experiments. Target shape (green circles with 2 cm radius) compared to the equilibrium reconstruction plasma boundary (black continuous line). In all figures the first time slice shows the handover condition. (a) Elongation of 1.9 with vertical instability growth rate of 1.4 kHz. (b) Approximate ITER proposed shape with neutral beam heating entering H-mode. (c) Diverted negative triangularity of -0.8 . (d) Snowflake configuration with a time-varying control of the bottom X-point. Extended traces for these shots can be found in Fig. 6.

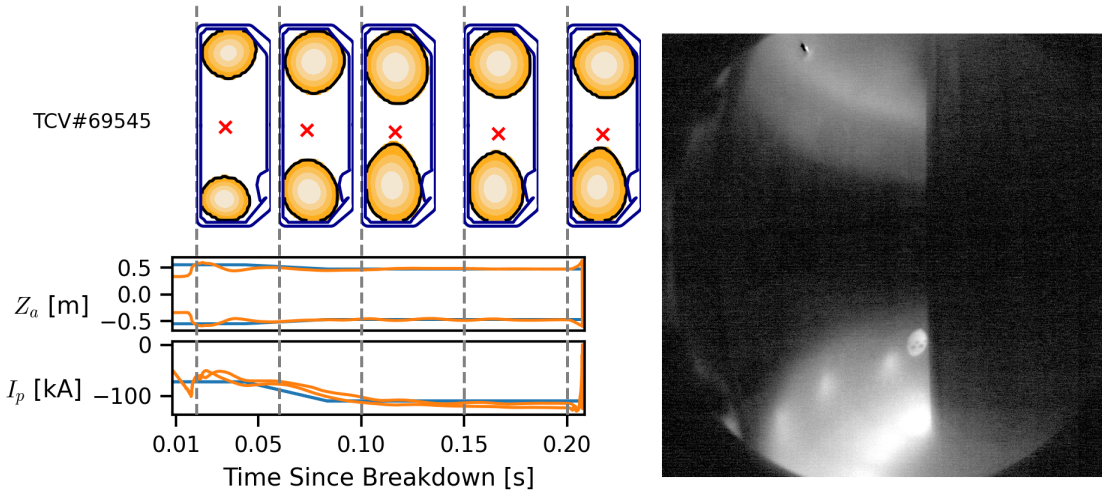


Fig. 4 | First ever demonstration of sustained control of two independent droplets on TCV for full control window of 200 ms. *Left:* Control of I_p for each independent lobe up to the same target value. *Right:* a picture where the two droplets are visible, taken from a camera looking into the vessel at $t=0.55$ s.

Additional Findings

Some controllers exhibited a number of interesting behaviours, briefly mentioned here and depicted in Extended Data Fig. 7. These control behaviours hint at further potential capabilities of learned control approaches. When given the goal to maintain only the plasma position and current, our architecture autonomously constructed a low elongation plasma that eliminates the vertical instability mode (Extended Fig. 7a), without being explicitly told to do so. Our control architecture can naturally choose to employ a varying combination of poloidal field and ohmic coils to drive the inductive voltage required for sustaining the plasma current (Extended Fig. 7b), in contrast to existing control architectures that typically assume a strict separation. Our architecture can learn to include non-linear physical and control requests by adding objectives to the goal specification. It can, for example, avoid limitations in the power supplies which occasionally cause “stuck” control coil currents when reversing polarity (Extended Fig. 7c), and avoid X-points in the vessel but outside the plasma (Extended Fig. 7d) when requested with high-level rewards.

Discussion

We present a new paradigm for plasma magnetic confinement on tokamaks. Our control design fulfils many of the community’s hopes for a machine learning based control approach [14], including high performance, robustness to uncertain operating conditions, intuitive target specification, and unprecedented versatility. This achievement required overcoming known gaps in capability and infrastructure through a combination of scientific and engineering advances: an accurate, numerically robust simulator; an informed trade-off between simulation accuracy and computational complexity; a sensor and actuator model tuned to specific hardware control; a realistic variation of operating conditions during training; a highly data efficient RL algorithm that scales to high dimensional problems; an asymmetric learning setup with an expressive critic but fast-to-evaluate policy; a process for compiling neural networks into real-time capable code; and deployment on a tokamak digital control system. This resulted in a broad range of successful hardware experiments that demonstrate fundamental capability alongside advanced shape control without requiring fine-tuning on the plant. It additionally shows that a free boundary equilibrium evolution model has sufficient fidelity to develop transferable controllers, offering a justification for using this approach to test control of future devices.

Efforts could further develop our architecture to quantify its robustness through analysis of the nonlinear dynamics [29–31], and reduce training time through increased re-use of data and multi-fidelity learning [32]. Additionally, the set of control targets can be expanded, for example to reduce target heat loads through flux expansion [5], aided by the use of privileged information in the critic to avoid requiring real-time observers. Furthermore, while the neural network infers the state of the system, it could be trained to be a full observer providing state estimates as additional outputs. The architecture can be coupled to a more capable simulator, for example incorporating plasma pressure and current density evolution physics, to optimize the global plasma performance.

Our learning framework has the potential to shape future fusion research and tokamak development. Underspecified objectives can be harnessed to find configurations that maximize a desired performance objective, or even the obtainable power production. Our architecture can be rapidly deployed on a novel tokamak without the need to design and commission the complex system of controllers deployed today, and evaluate proposed designs before they are constructed. More broadly, our approach may enable the discovery of novel reactor designs by jointly optimizing the plasma shape, sensing, actuation, wall design, heat load, and magnetic controller to maximize overall performance.

References (main text)

1. Hofmann, F. *et al.* Creation and control of variably shaped plasmas in TCV. *Plasma Physics and Controlled Fusion* **36**, B277 (1994).
2. Coda, S. *et al.* Physics research on the TCV tokamak facility: from conventional to alternative scenarios and beyond. *Nuclear Fusion* **59**, 112023. <https://doi.org/10.1088/1741-4326/ab25cb> (Aug. 2019).
3. Anand, H., Coda, S., Felici, F., Galperti, C. & Moret, J.-M. A novel plasma position and shape controller for advanced configuration development on the TCV tokamak. *Nuclear Fusion* **57**, 126026 (2017).
4. Mele, A. *et al.* MIMO shape control at the EAST tokamak: Simulations and experiments. *Fusion Engineering and Design* **146**, 1282–1285 (2019).
5. Anand, H. *et al.* Plasma flux expansion control on the DIII-D tokamak. *Plasma Physics and Controlled Fusion* **63**, 015006 (2020).
6. De Tommasi, G. Plasma magnetic control in tokamak devices. *Journal of Fusion Energy* **38**, 406–436 (2019).
7. Walker, M. L. & Humphreys, D. A. Valid Coordinate Systems for Linearized Plasma Shape Response Models in Tokamaks. *Fusion Science and Technology* **50**, 473–489. ISSN: 1536-1055. <https://doi.org/10.13182/FST06-A1271> (Nov. 2006).
8. Blum, J., Heumann, H., Nardon, E. & Song, X. Automating the design of tokamak experiment scenarios. *Journal of Computational Physics* **394**, 594–614 (2019).
9. Ferron, J. R. *et al.* Real time equilibrium reconstruction for tokamak discharge control. *Nuclear Fusion* **38**, 1055 (1998).
10. Moret, J.-M. *et al.* Tokamak equilibrium reconstruction code LIUQE and its real time implementation. *Fusion Engineering and Design* **91**, 1–15 (2015).
11. Xie, Z., Berseth, G., Clary, P., Hurst, J. & van de Panne, M. *Feedback control for Cassie with deep reinforcement learning in 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)* (2018), 1241–1246.
12. Akkaya, I. *et al.* Solving rubik’s cube with a robot hand. *arXiv preprint arXiv:1910.07113* (2019).
13. Bellemare, M. G. *et al.* Autonomous navigation of stratospheric balloons using reinforcement learning. *Nature* **588**, 77–82 (2020).
14. Humphreys, D. *et al.* Advancing Fusion with Machine Learning Research Needs Workshop Report. *Journal of Fusion Energy* **39**, 123–155 (2020).
15. Bishop, C. M., Haynes, P. S., Smith, M. E., Todd, T. N. & Trotman, D. L. Real-time control of a tokamak plasma using neural networks. *Neural Computation* **7**, 206–217 (1995).
16. Joung, S. *et al.* Deep neural network Grad-Shafranov solver constrained with measured magnetic signals. *Nuclear Fusion* **60**, 16034. <https://doi.org/10.1088/1741-4326/ab555f> (Dec. 2019).
17. Van de Plassche, K. L. *et al.* Fast modeling of turbulent transport in fusion plasmas using neural networks. *Physics of Plasmas* **27**, 022310. ISSN: 1070-664X. <https://doi.org/10.1063/1.5134126> (Feb. 2020).
18. Abbate, J., Conlin, R. & Kolemen, E. Data-driven profile prediction for DIII-D. *Nuclear Fusion* **61**, 046027 (2021).
19. Kates-Harbeck, J., Svyatkovskiy, A. & Tang, W. Predicting disruptive instabilities in controlled fusion plasmas through deep learning. *Nature* **568**, 526–531 (2019).

- 280 20. Jardin, S. *Computational methods in plasma physics* (CRC Press, 2010).
- 281 21. Grad, H. & Rubin, H. Hydromagnetic equilibria and force-free fields. *Journal of Nuclear Energy*
282 (1954) 7, 284–285 (1958).
- 283 22. Carpanese, F. Development of free-boundary equilibrium and transport solvers for simulation
284 and real-time interpretation of tokamak experiments. *EPFL, PHD thesis* (2021).
- 285 23. Abdolmaleki, A. *et al.* Relative entropy regularized policy iteration. *arXiv preprint arXiv:1812.02256*
286 (2018).
- 287 24. Paley, J. I., Coda, S., Duval, B., Felici, F. & Moret, J.-M. *Architecture and commissioning of the*
288 *TCV distributed feedback control system in 2010 17th IEEE-NPSS Real Time Conference* (IEEE, May
289 2010), 1–6. ISBN: 978-1-4244-7108-9.
- 290 25. Freidberg, J. P. *Plasma physics and fusion energy* (Cambridge university press, 2008).
- 291 26. Austin, M. E. *et al.* Achievement of reactor-relevant performance in negative triangularity shape
292 in the DIII-D tokamak. *Physical review letters* **122**, 115001 (2019).
- 293 27. Kolemen, E. *et al.* Initial development of the DIII-D snowflake divertor control. *Nuclear Fusion*
294 **58**, 066007 (2018).
- 295 28. Anand, H. *et al.* Real time magnetic control of the snowflake plasma configuration in the TCV
296 tokamak. *Nuclear Fusion* **59**, 126032 (2019).
- 297 29. Wigbers, M. & Riedmiller, M. *A new method for the analysis of neural reference model control in*
298 *Proceedings of International Conference on Neural Networks (ICNN'97)* **2** (1997), 739–743.
- 299 30. Berkenkamp, F., Turchetta, M., Schoellig, A. P. & Krause, A. Safe model-based reinforcement
300 learning with stability guarantees. *arXiv preprint arXiv:1705.08551* (2017).
- 301 31. Wabersich, K. P., Hewing, L., Carron, A. & Zeilinger, M. N. Probabilistic model predictive safety
302 certification for learning-based control. *IEEE Transactions on Automatic Control* (2021).
- 303 32. Abdolmaleki, A. *et al.* On Multi-objective Policy Optimization as a Tool for Reinforcement Learning.
304 *arXiv preprint arXiv:2106.08199* (2021).

305 Acknowledgements

306 We gratefully acknowledge the work and support of the TCV team (see the author list of Coda *et al.*
307 [2]) in enabling these experimental results. We thank Curdin Wüthrich and Yanis Andrebe for support
308 with the diagnostics. We would like to thank Chris Jones and Eluned Smith for strategic help and
309 inspiration at the start of the project. We thank Razia Ahamed, Paul Komarek, Veda Panneershelvam,
310 and Francis Song for their support in preparation and during this research. This work was supported
311 in part by the Swiss National Science Foundation.

Methods

Tokamak à Configuration Variable

The TCV tokamak [1, 33], shown in Fig. 5, is a research tokamak at the Swiss Plasma Center, with a major radius of 0.88 m, and vessel height and width of 1.5 m and 0.512 m respectively. TCV has a flexible set of magnetic coils that enable the creation of a wide range of plasma configurations. Electron cyclotron resonance heating and neutral beam injection [34] systems provide external heating and current drive, as used in the experiment on Fig. 3b. TCV is equipped with a number of real-time sensors, and our control policies use a subset of these sensors. In particular, we use 34 of the wire loops that measure magnetic flux, 38 probes that measure the local magnetic field and 19 measurements of the current in active control coils (augmented with an explicit measure of the difference in current between the Ohmic coils). In addition to the magnetic sensors, TCV is equipped with other sensors which are not available in real-time, such as the cameras shown in Fig. 2 and Fig. 4. Our control policy consumes TCV's magnetic and current sensors at a 10 kHz control rate. The control policy produces a reference voltage command at each timestep for the active control coils.

Tokamak simulator

The coupled dynamics of the plasma and external active and passive conductors are modelled with a free-boundary simulator, FGE [22]. The conductors are described by a circuit model where the resistivity is considered known and constant, and the mutual inductance is computed analytically.

The plasma is assumed to be in a state of toroidally symmetric equilibrium force balance (Grad-Shafranov equation [21]) where the Lorentz force $J \times B$ generated from the interaction of the plasma current density, J , and the magnetic field, B , balances the plasma pressure gradient ∇p . The transport of radial pressure and current density caused by heat and current drive sources is not modeled. Instead, the plasma radial profiles are modelled as polynomials whose coefficients are constrained by the plasma current I_p plus two free parameters: the normalized plasma pressure β_p , which is the ratio of kinetic pressure to the magnetic pressure, and the safety factor at the plasma axis q_A which controls the current density peakedness.

The evolution of the total plasma current I_p , is described as a lumped parameter equation based on the generalized Ohm's law for the MHD model. For this model, the total plasma resistance, R_p , and the total plasma self-inductance, L_p , are free parameters. Finally, FGE produces the synthetic magnetic measurements that simulate the TCV sensors, which are used to learn the control policies as discussed below.

Specific settings for the droplets

In the experiment with the droplets (Fig. 4), the plasma is considered pressureless, which simplifies the numerical solution of the force balance equation. Moreover, the G-coil was disabled in simulation since it was placed in open circuit during experiments (the fast radial fields it generates were deemed unnecessary for these plasmas). This experiment used an earlier model for the I_p evolution designed for stationary state plasma operation. This model has one free parameter, the radial profile of the neoclassical parallel plasma conductivity σ_{\parallel} [22]. This model was replaced with the one described above for the single domain plasma experiment, as it better describes the evolution of I_p , especially when it is changing rapidly.

Plasma parameter variation for robustness

We vary the plasma evolution parameters introduced above during training, in order to provide robust performance across the true, but unknown condition of the plasma. The level of variation is set within ranges identified from experimental data as shown in Table 1. In the single plasma experiments, we vary the plasma resistivity R_p , as well as the profile parameters β_p and q_A . L_p is not varied as it can be computed from a simple relation [35]. These are all independently sampled from a parameter-specific log-uniform distribution. In the experiment with droplets, we vary the initial ohmic coil current values according to a uniform distribution. We set two different values for the droplet $\sigma_{||}$ components. We sample the log of the difference between them from a scaled beta distribution, and the overall shift in the combined geometric mean from a log uniform distribution, and then solve for the individual $\sigma_{||}$. Parameter values are sampled at the beginning of each episode and kept constant for the duration of the simulation. The sampled value is deliberately not exposed to the learning architecture because it is not directly measurable. Therefore, the agent is forced to learn a controller which can robustly handle all combinations of these parameters. This informed and targeted domain randomization technique proved to be effective to find policies that track time targets for shape and I_p while being robust to the injection of external heating and the ELM perturbations during high confinement mode.

Sensing and Actuation

The raw sensor data on TCV goes through a low-pass filtering and signal conditioning stage [36]. We model this stage in simulation by a time delay and a Gaussian noise model, identified from data during a stationary plasma operation phase (Table 1). This sensor model (shown in Fig. 1b) captures the relevant dynamics affecting control stability. The power supply dynamics (also shown in Fig. 1b) are modelled with a fixed bias and a fixed time delay identified from data as well as an additional offset varied randomly at the beginning of each episode. The values for these modifications can be found in Table 1. This is a conservative approximation of the true thyristor based power supplies [36], but captures the essential dynamics for control purposes.

The control policy can learn to be robust against very non linear hardware specific phenomena. For example, when the current in the active coils changes polarity and the controller requests a too low voltage, the power supplies can get 'stuck', erroneously providing zero output current over extended period of time (Fig. 7b). This phenomenon might affect both the controller stability and precision. To demonstrate the capability of our controller to deal with this issue, we applied "learned region avoidance" in the advanced control demonstration to indicate that currents near zero are undesirable. As a result, the control policy effectively learns to increase the voltages when changing the current polarity to avoid stuck coils on the plant (Fig. 7c).

Learning Loop

Our approach uses an episodic training approach where data is collected by running the simulator with a control policy in the loop, as shown in Fig. 1a. The data from these interactions are collected in a finite-capacity first-in-first-out buffer [37]. The interaction trajectories are sampled at random from the buffer by a "learner" which executes the MPO algorithm to update the control policy parameters. During training, the executed control policy is stochastic to explore successful control options. This stochastic policy is represented by a diagonal Gaussian distribution over coil actions.

Each episode corresponds to a single simulation run which terminates either when a termination condition is hit, which we will discuss below, or when a fixed simulation time has passed in the episode. This fixed time was 0.2 s for the droplets, 0.5 s in the case of Fig. 6a and Fig. 6c, and 1 s otherwise. Each episode is initialized from an equilibrium state at the pre-programmed handover time which was

reconstructed from a previous experiment on TCV.

Our training loop emulates the control frequency of 10 kHz. At each step, the policy is evaluated using the observation from the previous step. The resulting action is then applied to the simulator which is then stepped. Observations and rewards are also collected at the 10 kHz control frequency resulting at training data collected at 0.1 ms intervals. For our simulation, we chose a time-step of 50 kHz. Hence, for each evaluation of the policy, five simulation time steps are computed. The action, i.e. the desired coil voltage, is kept constant during these substeps. Data from intermediate steps is only used for checking termination data and is discarded afterwards. This allows for choosing the control rate and simulator time step independently and hence setting the latter based on numerical considerations.

We use a distributed architecture [38] with a single learner instance on a Tensor Processing Unit and multiple actors each running an independent instance of the simulator. We used 5000 actors in parallel for our experiments, generally resulting in training times of 1 to 3 days, though sometimes longer for complex target specifications. We ran a sweep on the number of actors required to stabilize a basic plasma, and the results can be seen in Extended Data Fig. 8b. We see that a similar level of performance can be achieved with a drastic reduction in the number of actors for a moderate cost in training time. In Extended Data Fig. 8a, we also show the importance of using an asymmetric setup. In the symmetric version, the critic was sized the same as the policy, whose size is already limited by the control rate on the plant.

Since reinforcement learning only interacts sample-wise with the environment, the policy could be fine-tuned further with data from interacting with the plant. Alternatively, one might imagine leveraging the database of past experiments performed on TCV in order to improve the policy. However, it is unclear if the data is sufficiently diverse, given TCV’s versatility and the fact that the same plasma configuration can be achieved by various coil voltage configurations. Especially for novel plasma shapes, no data or only very limited data is available, rendering this approach ineffective. Conversely, the simulator can directly model the dynamics for the configurations of interest.

Rewards and Terminations

All of our experiments have multiple objectives that must be satisfied simultaneously. These objectives are specified as individual reward components that track an aspect of the simulation, typically a physical quantity, and these individual components are combined together into a single scalar reward value. Descriptions of the targets used are listed in Extended Data Table 3. The target values of the objectives are often time-varying (e.g. the plasma current and boundary target points), and are sent to the policy as part of the observations. This time-varying trace of targets is defined by a sequence of values at points in time, and are linearly interpolated for all time steps in between.

Shape targets for each experiment were generated using the shape generator [39] or specified manually. These points are then canonicalized to 32 equally spaced points along a spline, which are the targets that are fed to the policy. The spline is periodic for closed shapes, but non-periodic for diverted shapes, ending at the x-points.

The process for combining these multiple objectives into a single scalar is as follows. First, for each objective, the difference between the actual and target value is computed, and then transformed with a nonlinear function (see Extended Data Table 4) to a quality measure between 0 and 1. In the case of a vector-valued objective (e.g. distance to each target shape point), the individual differences are first merged into a single scalar through a “combiner”, a weighted nonlinear function (see Extended Data Table 5). Finally, a weighted combination of the individual objective-specific quality measures is computed into a single scalar reward value between 0 and 1 using a combiner as above. This (stepwise)

reward is then normalized so the maximum cumulative reward is 100 for 1 s of control. In cases where the control policy has triggered a termination (see Extended Data Table 6) a large negative reward is given.

We typically compute the quality measure from the error using a softplus or sigmoid, which provides a non-zero learning signal early in training when the errors are large, while simultaneously encouraging precision as the policy improves. Similarly, we combine the rewards using a (weighted) smooth max or geometric mean, which gives a larger gradient to improving the worst reward, while still encouraging improving all objectives. The precise reward definitions used in each of our experiments are listed in Extended Data Table 2, and the implementations are available in the supplementary material.

Deployment

As the stochastic nature of the training policy is only useful for exploration, the final control policy is taken to be the mean of the Gaussian policy at the conclusion of training. This gives a deterministic policy to execute on the plant. During training, we monitor the quality of this deterministic policy prior to deployment.

TCV’s control loop runs at 10 kHz though only half of the cycle time, i.e. 50 μ s, is available for the control algorithm due to other signal processing and logging. Hence, we created a deployment system that compiles our neural network into real-time capable code that is guaranteed to run within this time window. To achieve this, we remove superfluous weights and computations (such as the exploration variance) and then use tfcompile [40] to compile it into binary code, carefully avoiding unnecessary dependencies. We tailored the neural network structure to optimize the usage of the processor’s cache and enable vectorized instructions for optimal performance. The table of time varying control targets is also compiled into the binary for ease of deployment. In future work, targets could easily be supplied at run-time to dynamically adjust the control policy’s behaviour. We then test all compiled policies in an automated, extensive benchmark prior to deployment to ensure timings are met consistently.

Post-experiment Analysis

The plasma shape and position is not directly observed and needs to be inferred from the available magnetic measurements. This is done with magnetic equilibrium reconstruction, which solves an inverse problem to find the plasma current distribution that respects the force balance (Grad-Shafranov equation) and best matches the given experimental magnetic measurements at a specific time in a least-squares sense.

In a conventional magnetic control design, a real time capable magnetic equilibrium reconstruction is needed as a plasma shape observer to close the shape control feedback loop (shown as the “Plasma Shape” observer in Fig. 1f). In our approach instead we only make use of equilibrium reconstruction with LIUQE code [10] during post-discharge analysis to validate the plasma shape controller performances and compute physical initial condition for the simulation during training.

After running the experiment, we use this equilibrium reconstruction code to obtain an estimate of the plasma state and magnetic flux field. The plasma boundary is defined by the last closed flux surface (LCFS) in the domain. We extract the LCFS as 32 equiangular points around the plasma axis and then canonicalize with splines to 128 equidistant points. The error distance is computed using the shortest distance between each of the points that defined the target shape and the polygon defined by the 128 points on the LCFS. The shape RMSE is computed across these 32 error distances over all time steps in the time range of interest.

Errors on scalar quantities, such as I_p or elongation, are computed from the error between the

reference and the respective estimation from the equilibrium reconstruction over the time period of interest. The estimate of the growth rate of the vertical displacement instability [6] is computed from a spectral decomposition of the linearized system of equations of the simulator around the reconstructed equilibrium.

Comparison to Prior Work

In recent years, advanced control techniques have been applied to magnetic confinement control. De Tommasi *et al.* [41] describe a model-based control approach for plasma position control using a linear model and a cascaded feedback control structure. Gerškšič & De Tommasi [42] propose a Model Predictive Control approach, demonstrating linear MPC for plasma position and shape control in simulation including a feasibility estimate for hardware deployment. Boncagni *et al.* [43] have proposed a switching controller, improving on plasma current tracking on hardware but without demonstrating additional capabilities.

More generally, machine learning based approaches are being developed for magnetic confinement control and fusion in general not limited to control. A survey of this area is provided in Humphreys *et al.* [14], categorizing approaches into seven priority research opportunities including accelerating science, diagnostics, model extraction, control, large data, prediction and platform development. The first use of neural networks in a control loop for plasma control is presented in Bishop *et al.* [15], where a small-scale neural network is estimating the plasma position and low-dimensional shape parameters which are subsequently used as error signals for feedback control.

To the best of our knowledge our work is the first where (deep) reinforcement learning is used for feedback control for magnetic confinement control on a tokamak. In addition, our architecture constitutes a significant step forward regarding generality, where a single framework is used to solve a broad variety of fusion control challenges, satisfying several of the key promises of machine learning and artificial intelligence for fusion set out in [14].

Data availability

TCV experimental data is available on reasonable request from the authors (Federico Felici, federico.felici@epfl.ch).

Code availability

The learning algorithm used in the actor-critic RL method is MPO [23], a reference implementation of which is available under an open-source license [38]. Additionally, the software libraries launchpad [44], dm_env [45], sonnet [46], tensorflow [47] and reverb [37] were used, which are available as open-source as well. The code to compute the control targets, rewards and terminations is available in the supplementary materials. FGE and LIUQE are available on reasonable request from the Swiss Plasma Center at EPFL (Antoine Merle antoine.merle@epfl.ch, Federico Felici, federico.felici@epfl.ch), subject to agreement.

Author contributions

BT, FC, FF, JB, JD, MN, RH and TE contributed equally. DP, FF, JB, JD, MR, and RH conceived the project. AH, BT, FF, JB, JD, LF, MN, and MR led the project. AM, BT, CD, CS, FC, FF, FP, JB, JMM, MN and OS developed the physics simulations. BT, CD, DC, FF, JD, JKay, MN, MT, and TE integrated

the physics simulations with the learning framework. AA, BT, JD, JKeeling, RH, and TE developed the learning framework and performed learning experiments. CG, DC, FF, JB, JD, MN, SN, and TE developed the real-time neural network interface. CG, FC, FF, JD, and SC integrated the real-time neural network with the control system and ran tokamak experiments. CD, DC FC, FF, JB, JKeeling, MN, and TE developed data curation tools. BT, CG, FC, FF, JB, JKeeling, MN, RH, and TE developed and ran data analysis. AF, BD, DH, SC, KK, and PK consulted for the project. BT, FC, FF, JB, JD, MN, MR, RH, and TE wrote the manuscript.

Competing interests

BT, FC, FF, JB, JD, MN, RH and TE have filed a provisional patent application about the contents of this manuscript. The remaining authors declare no competing interests.

References (Methods)

33. Coda, S. *et al.* Overview of the TCV tokamak program: scientific progress and facility upgrades. *Nuclear Fusion* **57**, 102011 (2017).
34. Karpushov, A. N. *et al.* Neutral beam heating on the TCV tokamak. *Fusion Engineering and Design* **123**, 468–472 (2017).
35. Lister, J. B. *et al.* Plasma equilibrium response modelling and validation on JT-60U. *Nuclear Fusion* **42**, 708 (2002).
36. Lister, J. *et al.* The Control of Tokamak Configuration Variable Plasmas. *Fusion Technology* **32** (Nov. 1997).
37. Cassirer, A. *et al.* *Reverb: A Framework For Experience Replay* 2021.
38. Hoffman, M. *et al.* Acme: A research framework for distributed reinforcement learning. *arXiv preprint arXiv:2006.00979* (2020).
39. Hofmann, F. FBT-a free-boundary tokamak equilibrium code for highly elongated and shaped plasmas. *Computer Physics Communications* **48**, 207–221 (1988).
40. Abadi, M. *et al.* *Tensorflow: A system for large-scale machine learning in 12th USENIX symposium on operating systems design and implementation (OSDI 16)* (2016), 265–283.
41. De Tommasi, G. *et al.* Model-based plasma vertical stabilization and position control at EAST. *Fusion Engineering and Design* **129**, 152–157 (2018).
42. Gerškšič, S. & De Tommasi, G. *ITER plasma current and shape control using MPC in 2016 IEEE conference on control applications (CCA)* (2016), 599–604.
43. Boncagni, L. *et al.* Performance-based controller switching: An application to plasma current control at FTU in 2015 54th IEEE Conference on Decision and Control (CDC) (2015), 2319–2324.
44. Yang, F. *et al.* Launchpad: A Programming Model for Distributed Machine Learning Research. *arXiv preprint arXiv:2106.04516* (2021).
45. Muldal, A. *et al.* *dm_env: A python interface for reinforcement learning environments* 2019. http://github.com/deepmind/dm_env.
46. Reynolds, M. *et al.* *Sonnet: TensorFlow-based Neural Network Library* 2017. <http://github.com/deepmind/sonnet>.
47. Martín Abadi *et al.* *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems* Software available from tensorflow.org. 2015. <https://www.tensorflow.org/>.

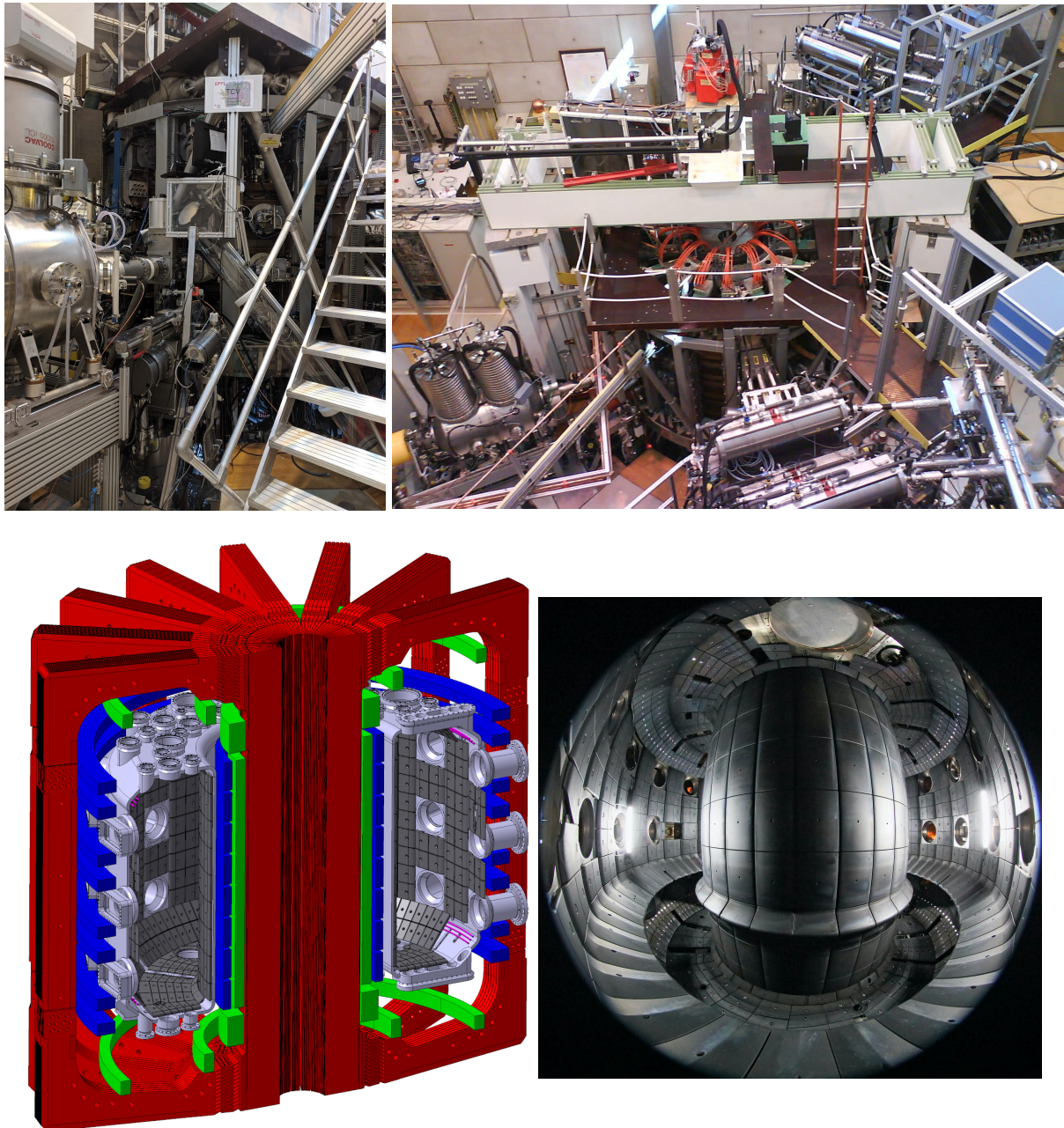


Fig. 5 | Pictures and illustration of the TCV. *Top*: Photographs showing the part of the TCV inside the bioshield. *Bottom Left*: CAD drawing of the vessel and coils of the TCV. *Bottom Right*: (©Alain Herzog / EPFL) view inside the TCV, showing the limiter tiling, baffles and central column

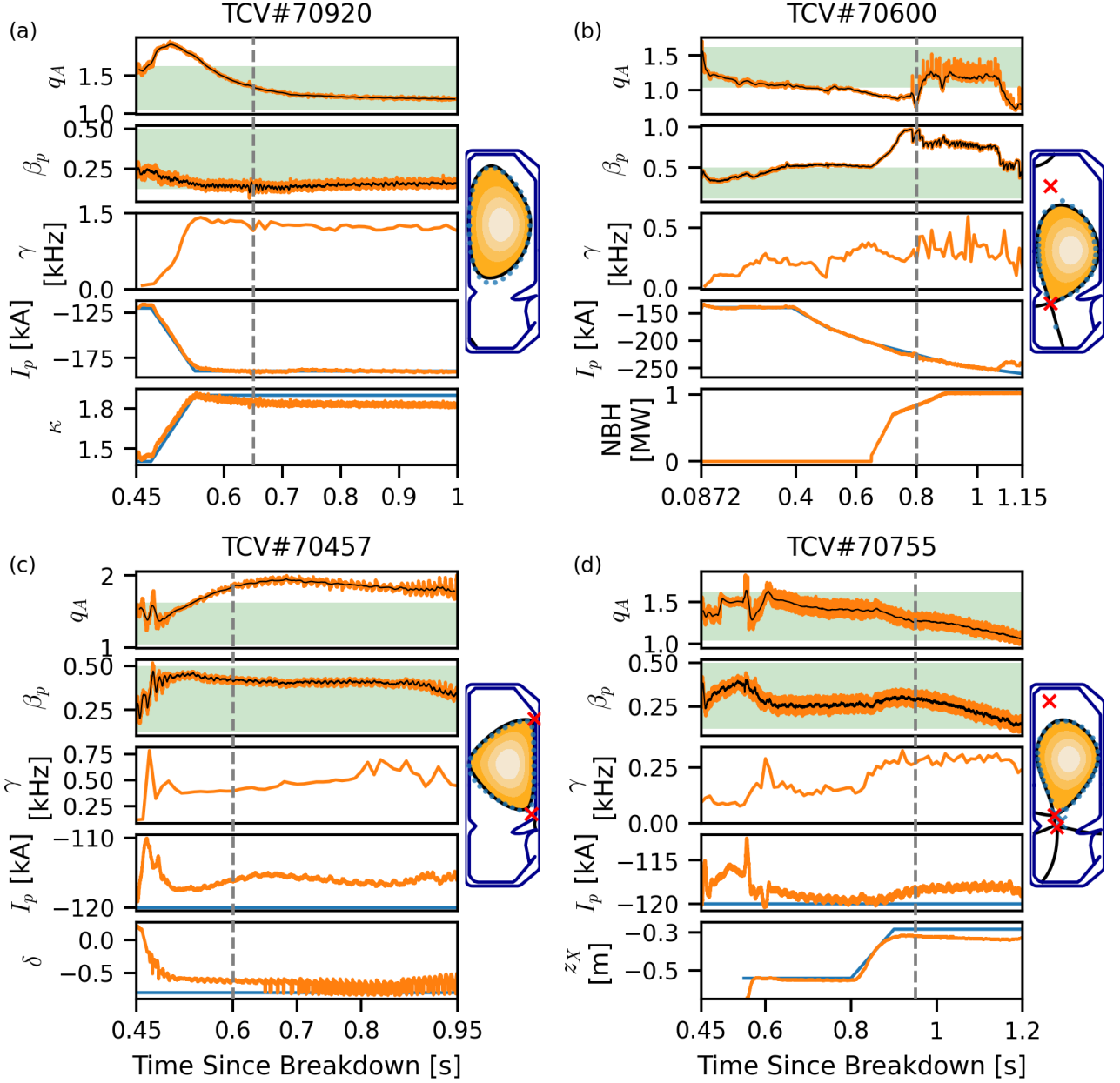
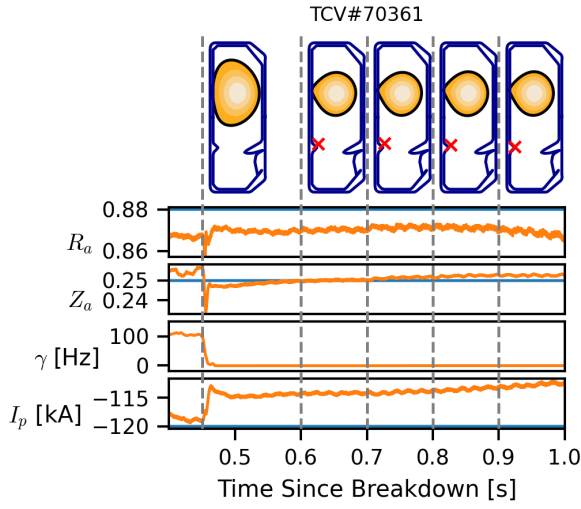
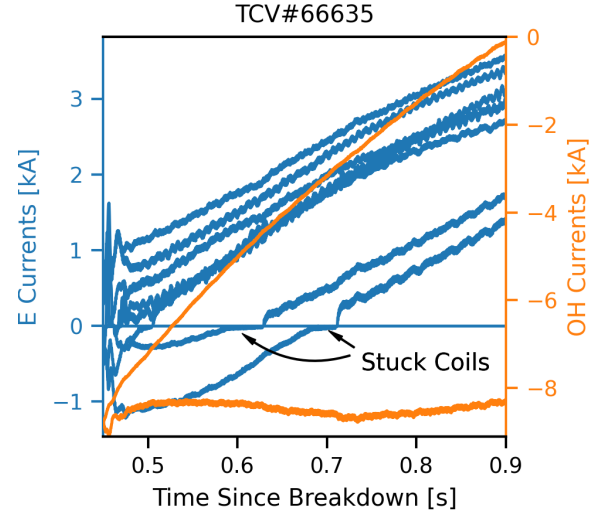


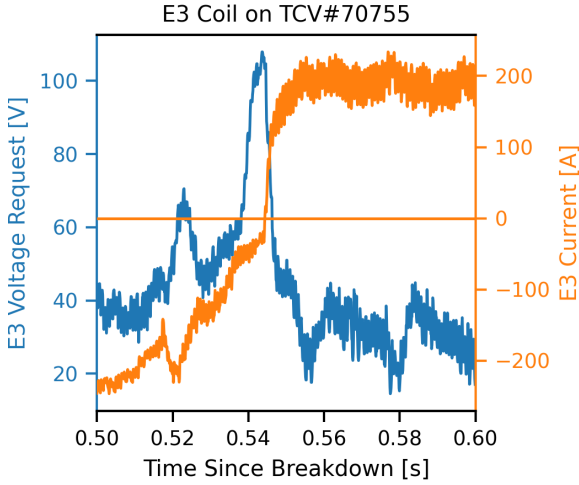
Fig. 6 | A larger overview of the shots in Figure 3. We plotted the reconstructed values for the normalized pressure β_p and safety factor q_A , along with in green the range of domain randomization these variables saw during training, which can be found in Table 1. We also plot the growth rate, γ , and the plasma current, I_p , along with the associated target value. Where relevant, we plot the elongation κ , the neutral beam heating, the triangularity δ and the vertical position of the bottom X-point z_X and its target.



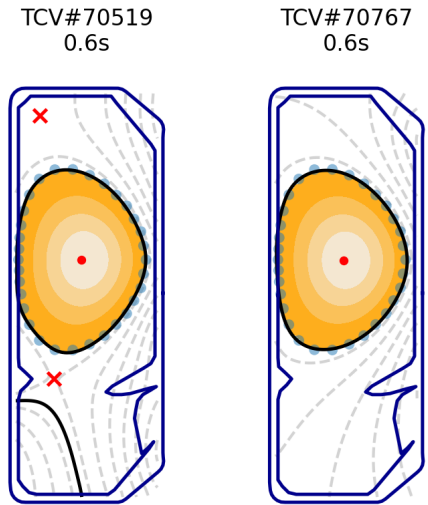
(a) When asked to stabilize the plasma without further specifications, the agent creates a round shape. The agent is in control from $t=0.45$ s and changes the shape while trying to attain R_a and Z_a targets. This discovered behaviour is indeed a good solution, since this round plasma is intrinsically stable with a growth rate $\gamma < 0$.



(b) When not given a reward to have similar current on both Ohmic coils, the algorithm tended to use the E-coils to obtain the same effect as the OH001-coil. This is indeed possible as can be seen by the coil positions in Fig. 1g, but causes electromagnetic forces on the machine structures. Therefore, in later shots a reward was added to keep the current in both Ohmic coils close together.



(c) Showing voltage requests by the policy to avoid the E3-coil from sticking when crossing 0 A. As can be seen in e.g. Figure 7b, the currents can get stuck on 0 A for low voltage requests, a consequence of how these requests are handled by the power system. Since this behaviour was hard to model, we introduced a reward to keep the coil currents away from 0 A. The control policy produces a high voltage request to move through this region quickly.



(d) An illustration of the difference in cross-sections between two different shots, where the only difference is the policy for the right shot was trained with an additional reward to avoid X-points in vacuum. This demonstrates how high level rewards can be used to find desired behaviour.

Fig. 7 | An overview of additional observations on the behaviour of the agent and the use of rewards.

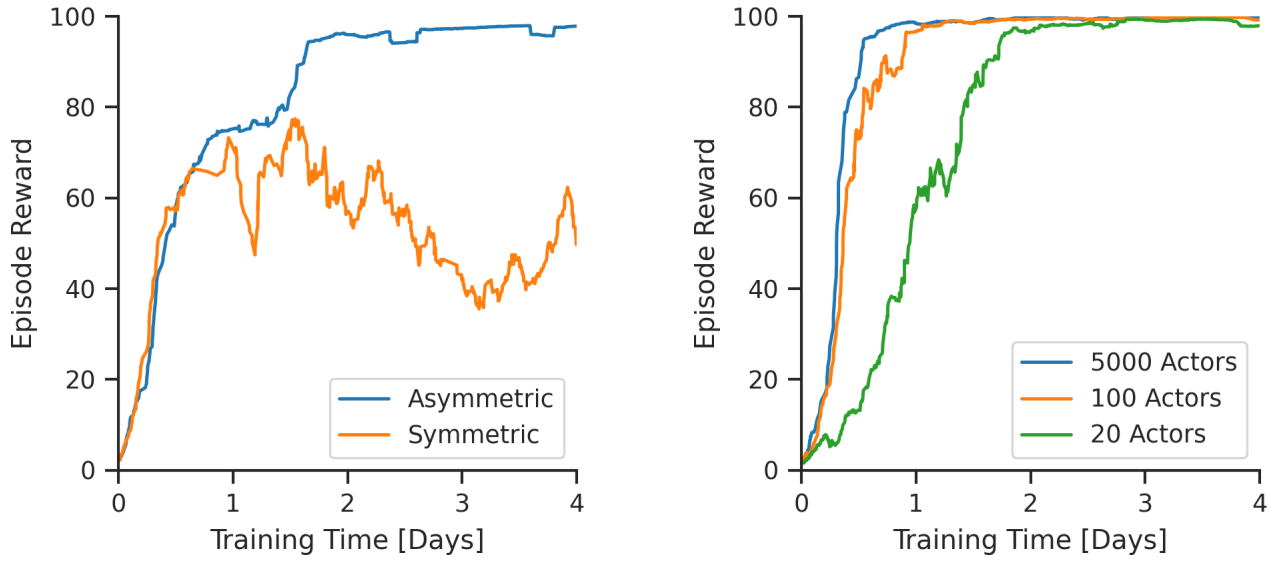


Fig. 8 | Episodic reward for the deterministic policy smoothed across 20 episodes with parameter variations enabled, where 100 means all objectives are perfectly met. *Left*: Comparison of the learning curve for the capability benchmark (as shown in Figure 2) using our asymmetric actor-critic vs. a symmetric actor-critic where the critic is using the same real-time capable feedforward network as the actor. These results indicate that a more expressive critic, as used in the asymmetric setup, is crucial for achieving high performance. *Right*: Comparison between using various amounts of actors for stabilizing a mildly elongated plasma. While the policies in this paper were trained with 5000 actors, this comparison shows that at least for simpler cases the same level of performance can be achieved with significantly lower computational resources.

	parameter	value	lower bound	upper bound
action delay	E	0.5 ms		
	F	0.5 ms		
	OH	0.5 ms		
	G	0.1 ms		
action bias (fixed)	E001	7 V		
	E002	−10 V		
	E003	−1 V		
	E004	0 V		
	E005	11 V		
	E006	−1 V		
	E007	−4 V		
	E008	44 V		
	F001	38 V		
	F002	−3 V		
	F003	6 V		
	F004	1 V		
	F005	−37 V		
	F006	−9 V		
	F007	5 V		
	F008	10 V		
	OH001	−54 V		
	OH002	−15 V		
action offset (random)	all coils		−20 V	20 V
measurement noise (std dev)	integrated flux loops	0.1 mWb		
	magnetic probes	0.1 mT		
	E coil currents	20 A		
	F coil currents	5 A		
	OH coil currents	20 A		
	G coil currents	2.5 A		
measurement delay	all measurements	0.02 ms		
plasma parameters (single domain)	R_p		2.5 $\mu\Omega$	10 $\mu\Omega$
	β_p		0.125	0.5
	q_A		1.04	1.625
plasma parameters (multiple domain)	$\sigma_{ }$ scaling		0.1	10
	$\sigma_{ }$ difference		0.33	3
	I_{OH}		−10 kA	−6 kA

Table 1 | **Simulation parameters for actuator, sensor and current diffusion models.** All parameters are identified from data. The action bias was fit on the power supply output voltage. Measurement noise is Gaussian additive noise and randomly sampled at each simulation time step. We use a fixed action bias with an additive random offset to account for non-ideal behaviour of power supply hardware. Current diffusion parameter variations account for the uncontrolled operating conditions. Parameter variations are sampled at the beginning of each episode but kept constant during the episode. The samples are drawn from uniform (action bias) and loguniform (current diffusion) distributions using the bounds in this table. For single plasma training, R_p , β_p and q_A are varied, while in a multiple plasmas training, we vary $\sigma_{||}$ and I_{OH} . In the latter case, we sample an overall geometric mean offset of the two $\sigma_{||}$ from a log-uniform distribution. We sample the log of the multiplicative difference between them from $B_s(4, 4)$, where B_s is a scaled beta distribution. We sample a single I_{OH} value for both coils. Parameters are sampled as absolute values unless explicitly indicated as scaling factors.

	Fundamental Capability	Elongated shape	ITER-like shape	Negative Triangularity	Snowflake	Droplets
Figure	Figure 2	Figures 3a, 6a	Figures 3b, 6b	Figures 3c, 6c	Figures 3d, 6d	Figure 4
Shot	TCV#70915	TCV#70920	TCV#70600	TCV#70457	TCV#70755	TCV#69545
Reward Components	Transforms, Combiners (if necessary), and weight (default=1)					
Diverted			Equal()	Equal()		
E/F Currents		SoftPlus(good=100, bad=50) GeometricMean()	SoftPlus(good=100, bad=50) GeometricMean()	SoftPlus(good=100, bad=50) GeometricMean()	SoftPlus(good=100, bad=50) GeometricMean()	
Elongation		SoftPlus(good=0.005, bad=0.2)		SoftPlus(good=0, bad=0.5)		
LCFS Distance	SoftPlus(good=0.005, bad=0.05) SmoothMax(-1)	SoftPlus(good=0.003, bad=0.03) SmoothMax(-1) weight=3	SoftPlus(good=0.005, bad=0.05) SmoothMax(-1) weight=3	SoftPlus(good=0.005, bad=0.05) SmoothMax(-1) weight=3	SoftPlus(good=0.005, bad=0.05) SmoothMax(-1) weight=3	
Legs Normalized Flux			Sigmoid(good=0.1, bad=0.3) SmoothMax(-5) weight=2			
Limit Point	Sigmoid(good=0.1, bad=0.2)	Sigmoid(good=0.2, bad=0.3)			Sigmoid(good=0.1, bad=0.2)	
OH Current Diff	SoftPlus(good=50, bad=1050)	ClippedLinear(good=50, bad=1050)	ClippedLinear(good=50, bad=1050)	ClippedLinear(good=50, bad=1050)	ClippedLinear(good=50, bad=1050)	ClippedLinear(good=50, bad=1050)
Plasma Current	SoftPlus(good=500, bad=20000)	SoftPlus(good=500, bad=30000)	SoftPlus(good=500, bad=20000) weight=2	SoftPlus(good=500, bad=20000) weight=2	SoftPlus(good=500, bad=20000) weight=2	Sigmoid(good=2000, bad=20000) weight=[1, 1]
R						Sigmoid(good=0.02, bad=0.5) weight=[1, 1]
Radius		SoftPlus(good=0.002, bad=0.02)		SoftPlus(good=0, bad=0.04)		
Triangularity		SoftPlus(good=0.005, bad=0.2)		SoftPlus(good=0, bad=0.5)		
Voltage Out of Bounds		Mean() SoftPlus(good=0, bad=1)	Mean() SoftPlus(good=0, bad=1)	Mean() SoftPlus(good=0, bad=1)	Mean() SoftPlus(good=0, bad=1)	
X-point Count		Equal()				
X-point Distance	Sigmoid(good=0.01, bad=0.15)		Sigmoid(good=0.01, bad=0.15) weight=0.5	Sigmoid(good=0.02, bad=0.15) weight=[0.5, 0.5]	Sigmoid(good=0.01, bad=0.15) weight=[0.5, 0.5]	
X-point Far	Sigmoid(good=0.3, bad=0.1) SmoothMax(-5)					
X-point Flux Gradient	SoftPlus(good=0, bad=3) weight=0.5		SoftPlus(good=0, bad=3) weight=0.5	SoftPlus(good=0, bad=3) weight=[0.5, 0.5]	SoftPlus(good=0, bad=3) weight=[0.5, 0.5]	
X-point Normalized Flux	SoftPlus(good=0, bad=0.08)		SoftPlus(good=0, bad=0.08)	SoftPlus(good=0, bad=0.08) weight=[1, 1]	SoftPlus(good=0, bad=0.08) weight=[1, 1]	
Z						Sigmoid(good=0.02, bad=0.2) weight=[1, 1]
Final Combiner	SmoothMax(-0.5)	SmoothMax(-5)	SmoothMax(-5)	SmoothMax(-0.5)	SmoothMax(-5)	GeometricMean()

Table 2 | **Rewards used in the experiments** Empty cells are not used in that reward. Any cell that does not specify a weight has an implicit weight of 1. Vector-valued weights (e.g. Droplets: R) return multiple values to the final combiner. See Table 3 for the descriptions of the different reward components, Table 4 for the transforms, and Table 5 for the combiners. All the terminations in Table 6 were used for these experiments. Code for these rewards is available in the supplementary material.

Reward Component	Description
Diverted	Whether the plasma is limited or diverted.
E/F Currents	The currents in the E and F coils, in amperes.
Elongation	The elongation of the plasma, this is its height divided by its width.
LCFS Distance	The distance in meters from the target points to the nearest point on the last closed flux surface (LCFS).
Legs Normalized Flux	The difference in normalized flux from the flux at the LCFS at target leg points.
Limit Point	The distance in meters from the actual limit point (wall or X-point) and target limit point.
OH Current Diff	The difference in amperes between the two OH coils.
Plasma Current	The plasma current in amperes.
R	The radial position of the plasma axis/centre.
Radius	Half of the width of the plasma.
Triangularity	The upper triangularity is defined as the radial position of the highest point relative to the median radial position. The overall triangularity is the mean of the upper and lower triangularity.
Voltage Out of Bounds	Punish the agent for going outside of the voltage limits.
X-point Count	Return the number of actual and requested X-points within the vessel.
X-point Distance	Returns the distance in meters from actual X-points to target X-points. Only X-points within 20cm are considered.
X-point Far	For any X-point that isn't requested, return the distance in meters from the X-point to the LCFS. This helps avoid extra X-points that may attract the plasma and lead to instabilities.
X-point Flux Gradient	The gradient of the flux at the target location with a target of 0 gradient. This encourages an X-point to form at the target location, but isn't very precise on the exact location.
X-point Normalized Flux	The difference in normalized flux from the flux at the LCFS at target X-points. This encourages the X-point to be on the last closed flux surface, and therefore for the plasma to be diverted.
Z	The vertical position of the plasma axis/centre.

Table 3 | **Reward Components** All of these return an actual and a target value, and many allow time-varying target values. See Table 2 for where and how they are used.

Transform	Description
ClippedLinear	Linearly maps the input values such that the good goes to 1 and bad to 0, then clips between 0 and 1.
Equal	Returns 1 if there is no error, returns 0 otherwise. Useful for boolean or integer outputs.
Sigmoid	Maps the input values such that good is 0.95 and bad is 0.05 in the output of the logistic function. This is similar to ClippedLinear, except there's still small impetus to improve beyond the good value and a little bit of reward signal for improvements below the bad value.
SoftPlus	Maps the input values such that good is 1 and bad is 0.1 in the output of the lower half of the logistic function, then clips to 0 and 1. This leads to a sharp drop-off as the value moves away from the good value, and a slow drop-off past bad. This is similar to a smooth relu.

Table 4 | **Reward transformations** Transforms that scale the different reward components. Transforms take a good and bad value that usually have some semantic meaning defined by the reward component, and then map it to the range 0 to 1. The good value should lead to a reward close to or equal to 1, while a bad value should lead to a reward close to or equal to 0.

Combiner	Formula	Description
Geometric Mean	$\left(\prod_{i=1}^n x_i w_i\right)^{\frac{1}{\sum_{i=1}^n w_i}}$	Takes the weighted geometric mean of the values.
Mean	$\frac{\sum_{i=1}^n x_i w_i}{\sum_{i=1}^n w_i}$	Takes the weighted mean of the values.
Smooth Max	$\frac{\sum_{i=1}^n x_i w_i e^{\alpha x_i}}{\sum_{i=1}^n w_i e^{\alpha x_i}}$	Takes the smooth maximum, parameterized with an α such that $\alpha = 0$ is equivalent to taking the mean, $\alpha = -\infty$ is equivalent to taking the minimum, and $\alpha = \infty$ is equivalent to taking the maximum.

Table 5 | **Reward Combiners** Combiners take a list of values and corresponding weights and returns a single value. Any values with a weight of 0 are excluded.

Termination	Termination Criteria
Coil current limits	Any coil current exceeds the physical limit of the plant.
Edge safety factor	Terminate when the edge safety factor q_{95} [48] goes below 2.2, which provides some margin over the threshold for a stable plasma ($q_{95} > 2$).
OH too different	The OH coil currents differ by more than 4 kA, which would cause high structural forces.
Plasma current limit	Plasma current is below the plant's disruption detector threshold, which is -60 kA for a single plasma, and -25 kA per plasma for droplets.
Solver not converged	Multiple subsequent simulation steps did not converge.

Table 6 | **Episode Termination Criteria** Description of the different terminations used. If any termination triggers, the episode ends with a large negative reward.