Magnetic control of tokamak plasmas through deep reinforcement learning with privileged information

Dmitri Sorokin[a], **Aleksandr Granovskiy**[a], **Ivan Kharitonov**[a], **Maksim Stokolesov**[a], **Igor Prokofyev**[a], **Evgeny Adishchev**[a], **Georgy Subbotin**[a], **Maxim Nurgaliev**[a]

[a] *Next Step Fusion, 56A Avenue François Clément, Mondorf-les-Bains, L-5612, Luxembourg* info@nextfusion.org

* Presenting author

1. Introduction

Reinforcement learning (RL) is capable of training high-performance control policies for a variety of domains from computer games [1] to physical robots [2, 3] and scientific equipment [4]. Recently, RL was applied to train real-time controllers for a tokamak plasma [5, 6, 7]. Tokamak plasma is controlled at a time scale of hundreds of microseconds and requires precise stabilization algorithms, which should be robust to sensor and actuator noises.

In the present work, we propose a novel approach to real-time plasma control using reinforcement learning with privileged information. Our method provides fast training times and learns an efficient control policy for real-time magnetic control without explicit plasma state reconstruction. We validate our approach on various plasma scenarios and successfully test it on a physical DIII-D tokamak.

2. Magnetic control of tokamak plasma

2.1 Related work

Traditional methods often decompose the control task to separate PID controllers for semi-independent control variables and use such controllers with state reconstruction codes [8], which compute control variables from the row observation data. RL provides a promising approach over traditional plasma control methods since it trains end-to-end controllers, which does not require additional state reconstruction codes and task decomposition. In [5], authors opted for MPO [9] algorithm with recurrent critic architecture, which enables the agent to assess the plasma state more accurately but slows the training. In [6], authors applied the PPO [10] algorithm together with the surrogate model of state reconstruction code [11], which requires diverse and high-quality experimental data for training.

2.2 Our method

We consider the task of magnetic control of plasma inside DIII-D tokamak. The agent is evaluated at 250 kHz frequency and observes real-time sensors (o^{rt}) which include magnetic probes (measures magnetic field), flux loops (measures magnetic flux), and coil current measurements. The agent trains in a simulated environment to learn a policy $\pi(a|o^{\text{rt}})$ which dynamically adjusts coil currents using actuator commands (a) to stabilize the plasma at target shape and position. We view the interaction between an agent and environment as a partially observable Markov decision process (POMDP) since the noisy real-time measurements do not provide enough information to fully reconstruct the internal state of the plasma. Besides the noisy real-time observations, during training agent has access to privileged information (o^{pri}), which includes non-noisy real-time sensors, current plasma shape and position, location of x-points (points in space at which the poloidal field has zero magnitude), and time derivatives.

We use NSFsim [12] simulator to reconstruct the plasma state from the experimental data and predict the evolution of the plasma given the actuator command. The simulator allows us to vary plasma parameters, including electron and ion temperatures, plasma resistance, and pressure gradients, which cannot be measured directly to train robust controllers. Application of reinforcement learning to physical simulators presents several challenges, such as (1) simulation speed, (2) reward function design, (3) NN-architecture choice to run in real-time, and (4) sim-to-real transfer.

To handle the **(1) simulation speed challenge**, we use a sample-efficient off-policy Soft Actor-Critic algorithm to train a continuous control policy. To speed-up the data acquisition, we run 50 copies of the training environment in parallel and perform one update of the neural networks per each environment step, which balances the speed of the environment step and neural networks update. In contrast to [5], we apply simple feedforward neural networks for both Actor and Critic, which allows us to train a plasma shape controller from scratch in 12 hours.

To address the **(2) reward function design challenge**, we formulate the reward as a function of several metrics that describe different aspects of the shape of the plasma:

$$\Delta_{LCFS} = \sum_{i=1}^{N} \left\| x_i^{\text{LCFS}} - y_i^{\text{LCFS}} \right\|_2 ,$$

$$\Delta_{mag} = \left\| x^{\text{mag}} - y^{\text{mag}} \right\|_2 ,$$

$$\Delta_{X-point} = \left\| x_{x-point} - y_{x-point} \right\|_2 ,$$

where x_i^{LCFS} and y_i^{LCFS} denote points of the current and target plasma shapes, x^{mag} and y^{mag} are current and target positions of the magnetic center and $x^{\text{x-point}}$, $y^{x-point}$ are positions of current and target x-points respectively. All metrics are transformed

to $[0, 1]$ range similarly to [5] and aggregated using smooth maximum function.

The episode terminates either if its length exceeds 1000 ms or if the distance between the current and target shape parameters exceeds 16 cm. This last condition helps streamline the training process by excluding states that moved too far from the target.

To tackle the **(3) NN-architecture choice challenge**, we assume that the run-time observation o^{rt} provides enough information to learn a control policy and privileged observation o^{pri} provides enough information on the state of the environment s. Hence we can use small $132 \times 256 \times 18$ MLP network which fits run-time constraints to model policy $\pi(o^{\text{rt}})$ and larger $552 \times 256 \times 1$ MLP network to model Q-value $Q(o^{\text{pri}}, a)$.

To address the **(4) sim-to-real transfer challenge**, we randomize key plasma parameters at the beginning of each episode. Specifically, we sample electron and ion temperatures in both the plasma center and boundary regions, as well as the effective ionic charge state (Z_{eff}), from uniform distributions. The specific ranges for these parameters are provided in the Appendix.

2.3 Evaluation

First, to demonstrate the robustness of our approach we perform sim-to-sim transfer and evaluate agent trained on NSFsim using well-tested DIII-D simulator GSEvolve [13]. We performed an evaluation using four different plasma shots shown in Fig. 1. Shots 182392, 182450, and 186093 have different shapes, while shots 186093 and 196088 have the same shape but opposite signs of plasma current. Evaluation time was limited by 1 second. The average quality of maintaining shape parameters is shown in Tab. 1. It is seen from the Tab. 1 that in sim-to-sim transfer control quality slightly decreases due to simplified modeling of plasma kinetics used in NSFSim during training. However, all agents managed to control the plasma successfully.

Table 1: Control performance.

	NSFSim	GSEvolve	DIII-D
Δ_{LCFS}, cm	1.5	2.3	2.2
Δ_{mag}, cm	0.63	1.5	1.6
$\Delta_{\text{x-point}}$, cm	0.57	6.3	6.3
t, ms	1000	1000	3000

2.4 Evaluation on physical DIII-D tokamak

After the sim-to-sim transfer we tested the agent on a real tokamak device. The shot started with the classical PID controller, after that, the control was transferred to our RL agent. The agent was able to successfully control plasma until the end of discharge (more then 3 seconds). The resulted precision of control is shown in Tab. 1.

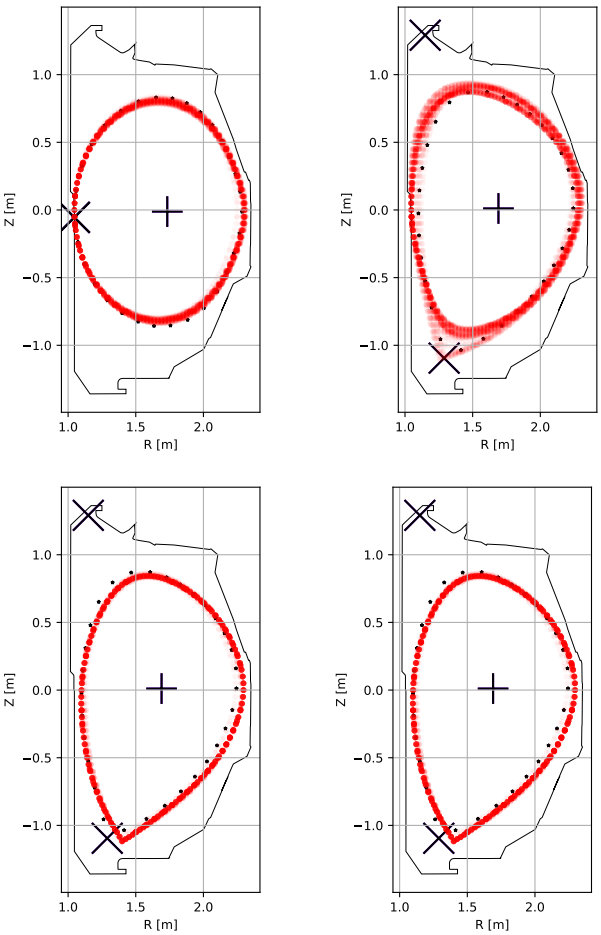

Fig. 1: Discharges used to validate RL agent: 182392, 182450, 186093, 196088. Discharges 186093 and 196088 have similar plasma shapes but different signs of the plasma current. Black dots indicate the target plasma shape, and red dots indicate the plasma shape maintained by the RL agent during all times of control.

3. Conclusion

High-quality plasma controllers are an essential part of future power plants. Compared to traditional approaches, we presented an RL-based approach which is fast to train and can be used to control different plasma shapes without tuning the training parameters. Our results on sim-to-sim and sim-to-real transfers, demonstrate the robustness of our approach to varying dynamics of the plasma and noises in sensor data. Our results demonstrate that privileged information enables efficient training of RL agents while keeping the simple architecture of the Critic network. The Actor learns the control policy directly from sensor data, which can speed up the controllers by eliminating the intermediate step of plasma state reconstruction.

Acknowledgments

This report was 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 thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

Work supported by US DOE under DE-FC02-04ER54698. We thank R. Clark, D.M. Orlov from the Center for Energy Research, University of California, San Diego, and H. Shen, W. Choi, and J. Barr from General Atomics for their valuable contributions and support of this work.

References

[1] OpenAI, :, Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemysław Dębiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Chris Hesse, Rafal Józefowicz, Scott Gray, Catherine Olsson, Jakub Pachocki, Michael Petrov, Henrique P. d. O. Pinto, Jonathan Raiman, Tim Salimans, Jeremy Schlatter, Jonas Schneider, Szymon Sidor, Ilya Sutskever, Jie Tang, Filip Wolski, and Susan Zhang. Dota 2 with large scale deep reinforcement learning, 2019.

[2] Tuomas Haarnoja, Sehoon Ha, Aurick Zhou, Jie Tan, George Tucker, and Sergey Levine. Learning to walk via deep reinforcement learning, 2018.

[3] Lerrel Pinto, Marcin Andrychowicz, Peter Welinder, Wojciech Zaremba, and Pieter Abbeel. Asymmetric actor critic for image-based robot learning. *arXiv preprint arXiv:1710.06542*, 2017.

[4] Dmitry Sorokin, Alexander Ulanov, Ekaterina Sazhina, and Alexander Lvovsky. Interferobot: aligning an optical interferometer by a reinforcement learning agent. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 13238–13248. Curran Associates, Inc., 2020.

[5] Jonas Degrave, Federico Felici, Jonas Buchli, Michael Neunert, Brendan Tracey, Francesco Carpanese, Timo Ewalds, Roland Hafner, Abbas Abdolmaleki, Diego de Las Casas, et al. Magnetic control of tokamak plasmas through deep reinforcement learning. *Nature*, 602(7897):414–419, 2022.

[6] Niannian Wu, Zongyu Yang, Rongpeng Li, Ning Wei, Yihang Chen, Qianyun Dong, Jiyuan Li, Guohui Zheng, Xinwen Gong, Feng Gao, et al. High-fidelity data-driven dynamics model for reinforcement learning-based magnetic control in hl-3 tokamak. *arXiv preprint arXiv:2409.09238*, 2024.

[7] Jaemin Seo, SangKyeun Kim, Azarakhsh Jalalvand, Rory Conlin, Andrew Rothstein, Joseph Abbate, Keith Erickson, Josiah Wai, Ricardo Shousha, and Egemen Kolemen. Avoiding fusion plasma tearing instability with deep reinforcement learning. *Nature*, 626(8000):746–751, February 2024.

[8] J.R Ferron, M.L Walker, L.L Lao, H.E. St John, D.A Humphreys, and J.A Leuer. Real time equilibrium reconstruction for tokamak discharge control. *Nuclear Fusion*, 38(7):1055–1066, July 1998.

[9] Abbas Abdolmaleki, Jost Tobias Springenberg, Yuval Tassa, Remi Munos, Nicolas Heess, and Martin Riedmiller. Maximum a posteriori policy optimisation, 2018.

[10] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms, 2017.

[11] Guohui Zheng, Songfen Liu, Zongyu Yang, Rui Ma, Xinwen Gong, Ao Wang, Shuo Wang, and Wulyu Zhong. Real-time equilibrium reconstruction by neural network based on hl-3 tokamak, 2024.

[12] Randall Clark, Maxim Nurgaliev, Eduard Khairutdinov, Georgy Subbotin, Anders Welander, and Dmitri M Orlov. Validation of nsfsim as a grad-shafranov equilibrium solver at diii-d. *Fusion Engineering and Design*, 211:114765, 2025.

[13] Anders Welander, Erik Olofsson, Brian Sammuli, Michael L. Walker, and Bingjia Xiao. Closed-loop simulation with grad-shafranov equilibrium evolution for plasma control system development. *Fusion Engineering and Design*, 146:2361–2365, September 2019.

Appendix A. Hyperparameters

Table A1: Hyperparameters.

Parameter	Value
num envs	50
tau	0.005
policy	Gaussian
discount factor	0.9
actor, critic learning rates	3e-5, 3e-5
alpha	0.2
entropy tuning	False
batch size	1024
num steps	10M
actor, critic hidden size	256, 256
start steps	10K
target network update frequency	1 step
replay buffer size	1M transitions
training steps	10M
action frequency	100 Hz
$T_{electron}$ center, boundary	1000-5000, 10-300 eV
T_{ion} center, boundary	1000-5000, 10-300 eV
$Z_{effective}$ center, boundary	1.0-4.0, 1.0-4.0