Offline Reinforcement Learning for Rotation Profile Control in Tokamaks

Rohit Sonker[1] **Hiro Josep Farre Kaga**[2] **Jiayu Chen**[1,4] **Andrew Rothstein**[2] **Ian Char**[5]
Ricardo Shousha[3] **Egemen Kolemen**[2,3] **Jeff Schneider**[1]
[1]*Robotics Institute, Carnegie Mellon University*
[2]*Princeton University*
[3]*Princeton Plasma Physics Lab*
[4]*The University of Hong Kong*
[5]*Lila Sciences* *

Editors: G. Sukhatme, L. Lindemann, S. Tu, A. Wierman, N. Atanasov

Abstract

Tokamaks remain leading candidates for achieving practical fusion energy, yet many important control problems inside these devices are still difficult or unsolved. One such challenge is controlling the plasma rotation profile, which strongly influences stability, confinement, and transport. While the average rotation can be controlled, controlling the full profile is challenging due to high dimensionality, response to multiple actuators and dependence on plasma condition. Learning-based control methods, such as reinforcement learning (RL), provide a potential solution to this challenging problem with an ability to model complex interactions leading to effective multi-input multi-output control. However, learning such policies is challenging due to the lack of accurate simulators that can model the rotation profile dynamics. In this work, we investigate the use of offline RL and offline model-based RL algorithms for rotation profile control, training them solely on historical data from the DIII-D tokamak. Our final method uses probabilistic models of plasma dynamics to generate rollouts for RL training. We deploy this policy on the DIII-D Tokamak and observe promising real-world results. We conclude by highlighting key challenges and insights from training and deploying an RL policy on a complex physical device while using only limited past data.

Keywords: Reinforcement learning, Tokamak Fusion, Offline RL, Probabilistic modelling

1. Introduction

Nuclear fusion has long been regarded as one of the most promising paths toward a stable and abundant energy future. Tokamaks, large toroidal devices where magnetic fields confine a rotating plasma, have emerged as the leading candidate to achieving sustained fusion conditions. Although significant progress has been made in plasma shaping, heating, and current-profile control (Ambrosino and Albanese, 2005), many control problems remain challenging due to nonlinear plasma dynamics and changing operating conditions.

In recent years, developments in Reinforcement Learning (RL) have led to significant progress in many applied domains, where a long horizon optimization problem needs to be solved in complex dynamics. RL has recently been explored for tokamak control, but obtaining on-policy data is difficult because experiments are expensive. Two main approaches have emerged to handle this limitation. (Degrave et al., 2022) was the first application of RL to tokamaks for learning shape

* work done while at Carnegie Mellon University

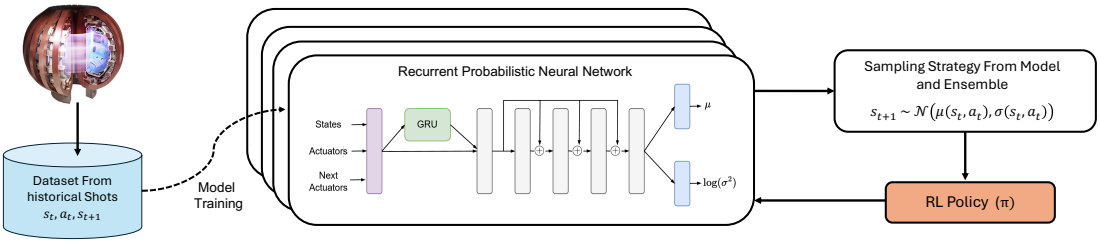

Figure 1: RL Policy Training using trajectories generated autoregressively from the RPNN dynamics model. The model is trained from historical experiment runs at the DIII-D Tokamak.

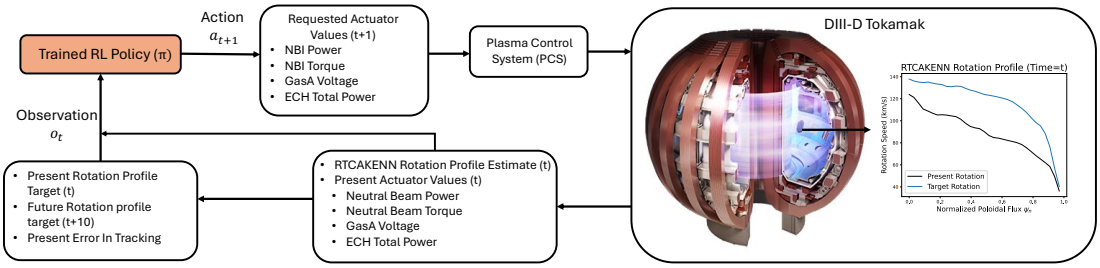

Figure 2: RL Policy Deployment on the DIII-D Tokamak. The policy is uploaded onto the Plasma Control System (PCS) where it receives signals and real profile estimates from RTCAKENN. The rotation profile variation can be seen from high value at the core to lower values at the edge. Tokamak diagram courtesy of DIII-D.

control. This line of work uses a physics-based simulator during training and then deploys the learned policy directly on the tokamak. A different line of work, for example (Char et al., 2023), learns a state transition model from historical data and then uses it for RL policy training. These works are promising, yet, they only target a limited set of tasks such as shape control or controlling scalars in the plasma (normalized plasma pressure, or current) for which traditional controllers already exist and are in use on tokamaks (Galperti et al., 2024; Walker et al., 2003; Boyer et al., 2019). To realize the full potential of learning-based control, we must tackle tasks where traditional methods are insufficient.

In this work, we address the challenging task of controlling the rotation profile in tokamaks. The profile here refers to the spatial variation in rotation value from core (innermost point) to the edge (outermost point) in the plasma. Controlling the rotation profile is important as it contributes to plasma stability and confinement (Wehner et al., 2015). Traditionally, rotation is controlled as a spatially averaged quantity by adjusting the neutral beam injection power (NBI or NB power) and the neutral beam injection torque (NBI or NB torque). Full profile control has been explored in prior works, (Wehner et al., 2015, 2017) show rotation profile control in simulation with MPC. The challenge stems from controlling a high dimensional profile which is affected by interactions from various nonlinear processes and also shows radial coupling - changes at one location impact profile at another location. Traditionally, neutral beam injection power and torque are the main actuators because they directly supply heating and momentum to the plasma. In addition, earlier DIII-D studies showed that Electron Cyclotron Heating (ECH) can modify toroidal rotation (DeGrassie et al.,

1999), and more recent work demonstrated sustainment of differential rotation using feedforward ECH and gas fueling (Bardoczi et al., 2025). These prior studies motivate the actuator choices used in this work and show that full rotation profile control in tokamaks is a multi-input, multi-output problem, a setting where simple PID approaches become difficult to apply effectively.

Our goal in this work is to extend past reinforcement learning methods to the challenging problem of rotation profile control in tokamaks. We restrict our approach to methods that rely only on past data and do not utilize physics-based simulators. This choice is motivated by the analysis of Abbate et al. (2024) and Wang et al. (2025) that show that purely physics-simulated models may not lead to accurate learning-based control. We therefore study this problem in the offline reinforcement learning setting, where policies are learned from a static dataset of past experiments from the DIII-D tokamak, operated by General Atomics in San Diego, California.

Thus, our contribution is to demonstrate the use of offline RL approaches on the task of full profile rotation control, a task which has not been addressed before, using only historical data. Our actuator space utilizes ECH and Gas Puffing in addition to commonly used neutral beams. We use a bootstrapped probabilistic dynamics model ensemble for learning tokamak dynamics (state transition probability), which supports robust policy training and provides an accurate test environment. Under this setup, we benchmark popular offline RL and offline model-based RL algorithms, demonstrating how RL can be applied to such a complex system with limited data and lack of accurate simulations. Finally, we show promising results in deploying an RL policy at DIII-D tokamak facility, while highlighting important lessons and challenges.

2. Related Works

Reinforcement Learning in Real-World Systems: Reinforcement learning has been applied across many real-world domains—from legged locomotion Lee et al. (2020) and robotic manipulation Levine et al. (2016) to control of industrial power systems Su et al. (2025). Advances in stability, expressiveness, and data efficiency have helped RL move closer to practical deployment, yet many challenges remain. Model-free RL Haarnoja et al. (2018); Schulman et al. (2015, 2017); Mnih et al. (2015) often requires large amounts of on-policy interaction, which is rarely available for safety-critical systems. Offline RL Levine et al. (2020); Kumar et al. (2020); Fujimoto et al. (2019) partly addresses this issue by learning entirely from past data and combining successful trajectories to form a policy.

Model-Based RL and Uncertainty: Model-based RL offers another path by learning a dynamics model from data and then training a policy using the learnt model as a simulator. These methods can be more data-efficient than model-free approaches (Chua et al., 2018), but inaccurate or overconfident models may lead to model exploitation, where a policy performs well in simulation but fails on the real system (Janner et al., 2019). Managing epistemic uncertainty is therefore essential, and probabilistic ensembles have become a standard way to capture uncertainty and stabilize training (Chua et al., 2018). Model-based offline approaches such as MOPO Yu et al. (2020) and MOBILE Sun et al. (2023a) extend this idea by limiting rollouts to well-modeled regions or penalizing uncertain predictions.

Planning-Based Control and its Limits: Model-based RL is closely connected to model predictive control methods such as MPPI Williams et al. (2017), the Cross-Entropy Method Botev et al. (2013), and other sampling-based controllers that evaluate many candidate trajectories through a model. These methods often require hundreds or thousands of rollouts at each control step, which

is not feasible for real-time tokamak control, since accurate dynamics models are very complex, and hence have higher inference time. Model-based RL avoids these issues by using the model only during training; once a policy is learned, it runs with a single forward pass and no online optimization.

Rotation-Profile Control in Tokamaks: Controlling the plasma rotation profile provides several benefits, differential rotation affects suppression of tearing instabilities Richner et al. (2024) and edge rotation affects neutral-gas penetration, leading to asymmetric fueling Emdee et al. (2024); Wilkie et al. (2024). Because of these effects, rotation-profile control is important for improving plasma performance. However, controlling the full profile is difficult due to nonlinear interactions and radial coupling. Early work by Wehner et al. (2015) proposed a linear–quadratic controller based on physics-based models, and Wehner et al. (2017) extended this using MPC, though both were limited to simulation studies. In practice, sim-to-real transfer remains a major challenge, which we discuss later in this work.

Learning-based control in Tokamak Fusion: Learning-based methods have been applied to several feedforward and feedback control tasks in tokamaks. For feedforward control, Mehta et al. (2024) used Bayesian optimization to design safe rampdowns, and Sonker et al. (2025) used Bayesian optimization with dynamics-informed priors to improve stability. Closed-loop RL has been used for plasma shape control at the TCV Tokamak, Degrave et al. (2022) Tracey et al. (2024) and at the WEST tokamak Kerboua-Benlarbi et al. (2024). These works rely on boundary-condition simulators because plasma shape dynamics are relatively well understood and easier to model Walker et al. (1997, 2020). Recent progress in data-driven plasma modeling Abbate et al. (2021) and Char et al. (2024) has enabled new learning-based control approaches. Char et al. (2023) used such models to train RL controllers for normalized pressure and differential rotation. Seo et al. (2024) developed policies that maintain high pressure while avoiding tearing modes. Wu et al. (2025) applied model-based RL to control current and shape parameters. Most recently, Wang et al. (2025) combined physics priors with historical data to model plasma dynamics and design RL rampdown trajectories for the TCV tokamak. However, in this work, we rely solely on historical data and learn RL policies for rotation-profile control directly from past experiments.

3. Methodology

3.1. Problem Setup

We frame tokamak profile control as a discrete-time Markov decision process (MDP) $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{T}, \gamma, R, d_0)$. Here, $\mathcal{S}$ is the plasma state, $\mathcal{A}$ is the action space, and $\mathcal{T} : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathcal{S})$ is the transition distribution over next states. Each timestep corresponds to 20 ms. The reward function is $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, γ is the discount factor, and d_0 is the initial-state distribution. Because the full plasma state cannot be observed in real time due to diagnostic limits, we instead learn a policy for a partially observable MDP. Let $\mathcal{O} \subset \mathcal{S}$ denote the observation space available to the policy. The objective is to learn a policy $\pi : \mathcal{O} \to \mathcal{P}(\mathcal{A})$ that maximizes the expected discounted return $\mathbb{E}\left[\sum_t \gamma^t R(s_t, a_t)\right], \quad s_0 \sim d_0,\ a_t \sim \pi(o_t),\ s_{t+1} \sim \mathcal{T}(s_t, a_t)$. The specific state, action, and observation variables used in this work are listed in Table 2

3.2. Dynamics Model Learning

We first aim to learn the plasma dynamics, such that given a state–action pair (s_t, a_t), we obtain a distribution over the next state $s_{t+1} \sim T(s_t, a_t)$. For this, we use the recurrent probabilistic neural network (RPNN) of Char et al. (2024) as our dynamics model. The RPNN predicts the parameters of a Gaussian distribution (mean and log variance), such that $s_{t+1} \sim \mathcal{N}(\mu_{t+1}, \sigma_{t+1})$ where for every timestep, $\mu_{t+1}, \sigma_{t+1} = T_\theta(s_t, a_t)$ where T_θ is an RPNN model parameterized by θ. For this, the RPNN has a mean head and a log variance head. A diagram of the network architecture can be seen in Fig.1.

Our model is trained on roughly 18,000 past DIII-D experiments ("plasma shots"). Each shot contains about four seconds of data sampled at 20 ms. We restrict training to the flat-top phase of the discharge, where the plasma is held in steady, high-performance conditions. The full list of states and actions is given in Table 2. For profiles i.e., electron temperature, ion temperature, pressure, rotation, and the q profile, we use Zipfit reconstructions (Logan et al., 2018), which provide smooth, physically constrained estimates. These reconstructions are not available in real time, so during deployment the policy instead uses RTCAKENN predictions (Shousha et al., 2023). This creates a noticeable sim2real gap, which we account for later in the policy-learning and discussion sections. We tried training a model with RTCAKENN data, however the model did not learn well, due to large noise and variations in the signal. Following Char et al. (2024), we reduce all profile quantities using PCA: 4 components each for electron temperature, ion temperature, and rotation, and 2 components each for pressure and the q profile, which captures more than 99% of the variance.

We introduce two modifications to the RPNN training procedure that improve performance. First, we train the model in two stages: (1) an initial phase using mean squared error loss, and (2) a second phase where only the log variance head is trained using negative log-likelihood. This leads to higher explained-variance scores. Second, we train a bootstrapped ensemble of RPNNs. The predicted log variance captures the aleatoric uncertainty, while disagreement across ensemble members provides an estimate of epistemic uncertainty (Chua et al., 2018). Further training details are given in the Appendix. This ensemble of probabilistic models serves as the evaluation environment for all policy-learning methods, since direct testing on the tokamak is prohibitively expensive.

3.3. RL Policy Learning

We now use the learned dynamics model to train an RL policy. To simulate a tokamak experiment, or a "shot," we generate autoregressive rollouts using the ensemble of dynamics models T_{θ_i}, where $i = 1, .., 25$. Each model in the ensemble represents a plausible version of the device dynamics, and we assume that the true tokamak dynamics lies within the distribution spanned by these models.

During RL training, we sample one model T_{θ_i} for each trajectory and generate autoregressive trajectories by sampling from that model's predictive distribution (mean and log-variance). A new model is selected when a new trajectory is sampled. This trains the policy to perform well across all plausible dynamics, and because regions of high model variance produce more spread in the sampled next states, the policy naturally receives fewer synthetic transitions in uncertain areas. As a result, we do not include any explicit uncertainty penalty in the reward.

Policy Observation Space - The policy observes the current actuator values (neutral beam power, torque, gas-puffing voltage, and total ECH power), which together describe the machine's present actuation state. We also include the RTCAKENN estimate of the present rotation profile, along with the present target profile and a future target profile shifted 10 timesteps ahead. All profile

quantities, i.e., present values and targets, are converted into their PCA components before being passed to the policy. Finally, we also compute the present tracking error and include it as part of the observation space.

Policy Action Space - The policy controls NBI power, NBI torque, gas-puffing voltage, and total ECH power. NBI torque directly affects the rotation profile at the beam deposition location, which is typically near the plasma core. Because torque and power originate from the same source, and the neutral beams are central to overall plasma sustainment, we include both as independent actuators. Gas puffing and ECH influence rotation in more indirect ways: gas particles can increase density, or adding slow moving particles slows the rotation through momentum conservation, starting at the outermost spatial locations and propagating inwards. Whereas ECH tends to reduce density and thereby increases rotation, typically starting at intermediate spatial locations. Together, these actuators provide enough flexibility to meaningfully shape the rotation profile.

Target Sampling - Because plasma experiments operate in many configurations, some target profiles may be unattainable in a given setup. To address this, we use the following target-sampling strategy during policy training. When replaying a reference shot from the dataset, we select another "target" shot from the same experimental session. From this target shot, we sample rotation profiles at two time points and use these two profiles to construct a step-function target. This ensures that all sampled targets are achievable within the operating conditions of the reference scenario, while maintaining target diversity.

Reward Function - We use the negative mean squared tracking error on tracking the whole rotation profile (33 dimensions per timestep), which is reconstructed from the PCA components.

$$R(t) = -\left\| rot_{target}(t) - rot(t) \right\|_2^2$$

where $rot_{target}(t)$ is target rotation at t and $rot(t)$ is the actual rotation at time t.

Sim2Real Gap and domain randomization - Although sampling across the dynamics ensemble already provides a form of domain randomization, we further add small observation noise during policy training to improve robustness. We increase the noise level until performance in the test environment begins to degrade. In practice, we add zero-mean Gaussian noise with a standard deviation of 0.1 to the observation vector.

Policy Testing - Since the true test environment is unavailable and tokamak time is limited, we evaluate the policy using the learned dynamics model with a different sampling strategy. Here, the goal is to maximize accuracy relative to the collected data. At each timestep, we sample predictions from all models in the ensemble and take their mean to estimate the next state. Empirically, this strategy provides the highest accuracy on the dynamics-model test set.

4. Results and Discussion

4.1. Benchmarking Offline RL Methods

We first present tracking results using repeated rollouts on three shots that share similar conditions to those used on the experiment day. At the tokamak facility, each experiment relies on a past shot—called the reference shot—to provide stable and well-understood operating conditions. For our experiment, the reference shot was 161409, so we evaluate on this shot along with 161410 and 161412, which were performed in the same session and follow nearly identical setups. This provides the closest reconstruction of the real experiment configuration available from our dataset.

We perform a comparative analysis of various offline RL algorithms using an open-source offline RL codebase tailored for plasma control (Chen et al., 2025a). Notably, two model-based offline RL algorithms featured in this codebase, BAMCTS (Chen et al., 2024) and ROMBRL (Chen et al., 2025b), have been evaluated for rotation profile control in their original studies, but only in simulation. The results in table 1 show the tracking accuracy of the full rotation profile in each of these shots with different types of algorithms. A detailed table showing tracking performance at different profile locations is also shown in table 3.

Goal-Conditioned Imitation Learning (GCIL) and all offline model-free RL methods are trained directly on the same offline dataset used to learn our dynamics models. For offline model-based methods, the learned dynamics model is available under the training sampling mode described earlier. We also evaluate Proximal Policy Optimization (PPO), a model-free method with respect to policy optimization, but we train it with our ensemble of probabilistic dynamics models serving as the simulator. It is also important to note that our dataset is relatively small and corresponds to a highly complex control task. This makes this setting particularly challenging for offline RL methods. Adding conservative penalties in this setting leads to overly cautious behavior, and in practice PPO outperforms these methods. Based on these observations, we select PPO as the final algorithm for deployment.

Table 1: Performance of different algorithms on Rotation Profile tracking. Root mean squared error (RMSE), values (lower is better), with standard error in parenthesis, are given for simulated runs of shots 161409, 161410, 161412 with 10 seeds each.

Category	**Algorithm**	**RMSE**
Behaviour Cloning	GCIL (Ding et al., 2019)	36.48 (0.37)
Offline Model-Free RL	CQL (Kumar et al., 2020)	80.79 (0.25)
	IQL (Kostrikov et al., 2021)	54.52 (0.78)
	TD3BC (Fujimoto and Gu, 2021)	58.73 (0.51)
	EDAC (An et al., 2021)	32.07 (0.66)
	MCQ (Lyu et al., 2022)	44.64 (0.44)
Offline Model-Based RL	COMBO (Yu et al., 2021)	51.46 (0.62)
	BAMCTS (Chen et al., 2024)	57.43 (0.71)
	ROMBRL (Chen et al., 2025b)	91.44 (0.70)
	RAMBO (Rigter et al., 2022)	40.05 (0.43)
	MOPO (Yu et al., 2020)	84.63 (0.72)
	MOBILE (Sun et al., 2023b)	32.99 (0.40)
	PPO (Schulman et al., 2017)	**29.50 (0.46)**

4.2. Experiment Results

We now present the results of online testing at the DIII-D tokamak. After training our final policy, we deployed it on the Plasma Control System (PCS) at DIII-D. Keras2c package (Conlin et al., 2021) was used to convert the policy to C code. The full deployment loop is shown in Fig.2. The tracking performance for this shot is shown in Fig.3 across several values of ψ_n, the normalized flux radius, which indicates how far a point is from the plasma center toward the edge.

For this experiment, we defined a step-function target with two changes, as illustrated in Fig. 3. The RL controller was activated at 1.5s, once the ramp-up phase had ended. At this point, the controller quickly adjusted the torque to bring the rotation profile toward the initial target. When the target changed at 3s, we observed a small increase in NB power, a slight reduction in torque, and a sharp rise in gas puffing. These changes jointly produced a decrease in the rotation profile.

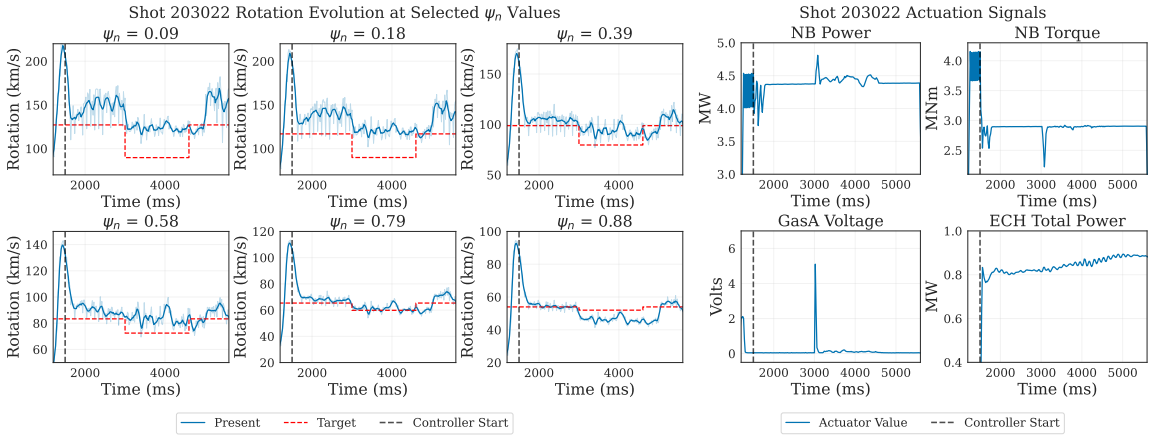

Figure 3: Real tokamak results for Shot 203022 from DIII-D. The first set of plots show value of rotation at different normalized poloidal flux values (ψ_n), such that $\psi_n = 0$ represents plasma at the core and $\psi_n = 1$ represents plasma at the outer edge. Blue represents the present value while red represents the target. Smoothened lines are shown in dark. Second set of plots shows the changes in actuator signals made by RL policy. Controller starts running at 1500ms when the flat-top phase of the experiment starts.

The decrease in torque directly influences rotation. The effect of gas injection is more complex. Around 3s, the gas valve voltage increases sharply, resulting in a higher gas flow into the plasma. Although the injected gas alone is not sufficient to produce the observed change in rotation, it can influence the profile because the newly introduced particles initially have lower momentum. While NB torque does not show a sustained change, we do notice the rotation profile to have changed significantly. This likely involves a nonlinear transient combined with a hysteresis effect that locks the rotation. Interestingly, the controller recognizes the improved state and maintains the actuator setting to sustain it. This highlights the advantage of RL in adapting to evolving plasma conditions.

At 4.5s, the target changes to a value that closely matches the existing rotation profile, with only a small increase required near the mid-radius region ($\psi_n \approx 0.5$). At this stage, the power and torque adjustments from the previous phase stop, indicating that the controller considers the profile to be close to the desired state. While this causes a small increase in rotation, a pronounced jump appears near the plasma core. We examined possible causes, including tearing instabilities and major sawtooth events, but found no clear evidence of either, and the cause of the spike observed is unclear. At approximately 5.2s, the rotation subsequently moves back toward the target value without any controller contribution. Full profile views are provided in Fig. 5 and Fig. 6.

Overall, while there is a consistent offset in the tracking values, the controller responded to target changes by coordinating the actuators in a physically meaningful way.

4.3. Simulated Experiment Results

To further analyze the experiment results and understand the effect of the sim2real gap, we also simulate the experiment shot using our test environment and the dynamics-model ensemble. The main difference between the real and simulated settings lies in the source of the rotation-profile inputs. In simulation, Zipfit profile (offline profile fitting algorithm) estimates are used, whereas, in

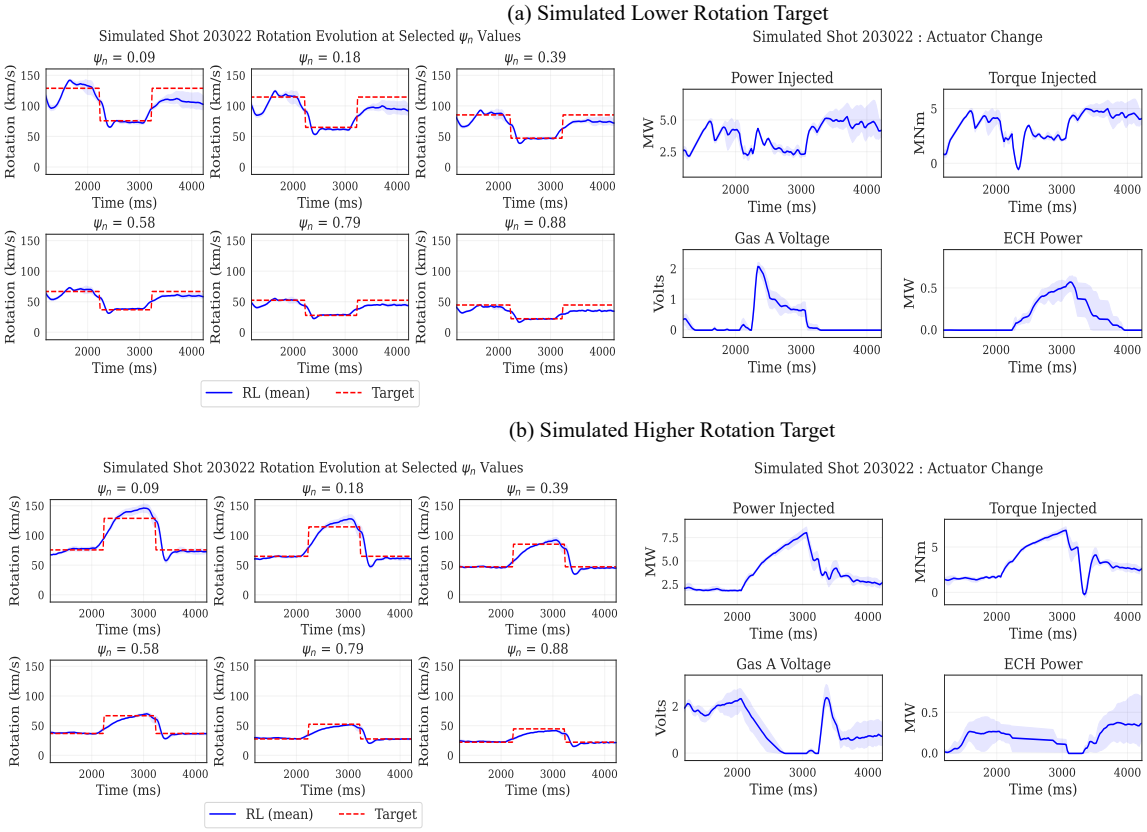

Figure 4: Simulated results for shot 203022 using the dynamics-model environment. Two target patterns are tested: (a) decreasing the rotation profile and returning, matching the experiment, and (b) increasing the profile and returning. Left plots show rotation at different normalized flux values (ψ_n), where $\psi_n = 0$ is the core and $\psi_n = 1$ the edge (blue: actual, red: target; dark lines: smoothed). Right plots show the RL-controlled actuator signals. Shaded regions denote the 5th–95th percentile over 30 seeds, with the solid line showing the mean. Both cases demonstrate strong tracking performance in the absence of the sim2real gap.

realtime, RTCAKENN profile estimates are used. We run the simulated experiment using two sets of target profiles: (a) targets that go from a higher rotation level to a lower level and back—matching the pattern used in the actual experiment, and (b) targets that go from lower rotation to higher rotation and back. The results are shown in Fig. 4.

For case (a), the actuator behavior directionally resembles that observed in the real experiment, although major differences exist in magnitudes. These differences are mainly attributed to the source signals used for the profiles, namely Zipfit in simulation and RTCAKENN in the experiment. The RTCAKENN signals are realtime signals used by the experimental policies which are noisy and exhibit high-frequency variations. This strongly impacts the control policy. Tracking performance is noticeably better in simulation because Zipfit profiles are smoother, and, since the dynamics model has no actuator delays or lags, the effects of control actions appear clearly. When the first target drop occurs at 2250 ms, torque decreases and the rotation drops correspondingly. Additional

reductions in torque and increased gas puffing help maintain the lower rotation value. ECH is also used to stabilize the profile. When the target increases again at 3250 ms, torque and power rise, gas puffing is shut off, and ECH peaks before tapering off, all contributing to a rise in rotation across the profile.

For case (b), where rotation moves from low to high and back, we observe similar coordinated behavior. When the target increases, torque increases and gas puffing decreases, leading to faster rotation. When the target drops again, torque is sharply reduced and gas is increased, bringing the rotation down. In the final stage, ECH is introduced to help maintain the lower rotation level.

In summary, the RL policy performs well in simulation and uses all actuators to track the full profile. On the tokamak, however, the policy is affected by the sim2real gap: the targets are followed, but steady-state errors persist. Even so, the actuator responses in the experiment show that the policy makes intelligent and coordinated adjustments, demonstrating its ability to control the complete rotation profile under real operating conditions.

5. Discussion and Conclusion

In this work, we show how offline RL methods—both model-free and model-based—can be used for rotation-profile control in tokamaks. The system development and experiment revealed several challenges that point toward useful directions for the RL community.

Sim2Real Gap and RL with Privileged Information. Tokamak control has a strong sim2real gap due to the shift from Zipfit signals (offline and smooth) to RTCAKENN signals (real-time and noisier). This mismatch can limit policy performance and suggests that RL methods using privileged information may be useful. Here, Zipfit profiles provide high-quality offline information, while RTCAKENN provides the restricted real-time input. Similar approaches that train with rich states and deploy with limited observations have been successful in robotics (Kumar et al., 2021; Lee et al., 2020) and could be useful here.

Actuator Failures. During the experiment session, we encountered problems with gyrotrons (ECH) and neutral-beam responses, which is common given the complexity of tokamak actuators and diagnostics. Control methods must therefore handle partial failures, saturation, and state dropouts. RL approaches explicitly designed for such conditions (Fei et al., 2020), could play an important role in making controllers more reliable in practice.

Changing Dynamics Over Time. Our dataset spans more than a decade of DIII-D operation, during which the machine underwent multiple upgrades, causing the underlying dynamics to drift over time. Prior work (Sonker et al., 2025) notes that old discharges cannot be replayed with identical commands for this reason. This non-stationarity is difficult for offline RL, which typically assumes fixed dynamics. Bootstrapped models help capture some of this variation, but not fully. Future improvements could include latent context variables for configuration-dependent changes (Shaj et al., 2022) or online policy adaptation at test time (Hansen et al., 2020).

To conclude, we developed offline RL approaches for rotation-profile control and evaluated several model-free and model-based algorithms. PPO trained with experience from a probabilistic dynamics ensemble produced the best policy, which we then deployed on the DIII-D tokamak. Although real-world testing was limited, the results demonstrate meaningful control of a previously unaddressed profile-regulation task. The probabilistic training strategy shows promising sim2real transfer, and the broader methodology offers an early step toward general-purpose, data-driven profile control in fusion plasmas.

Acknowledgments

This work would not have been possible without support from many people. We wish to thank the incredible staff at DIII-D that supported us during this experiment. We also thank Aravind Venugopal and Namrata Deka for discussions related to data processing and dynamics modeling, as well as, Yang Fu and Haomin Bao for their help in running offline RL benchmarks. This material is based upon work supported by the U.S. 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-FC02-04ER54698, DE-SC0024544 and DE-SC0015480.

References

J Abbate, E Fable, B Grierson, A Pankin, G Tardini, and E Kolemen. Large-database cross-verification and validation of tokamak transport models using baselines for comparison. *Physics of Plasmas*, 31(4), 2024.

Joseph Abbate, Rory Conlin, and Egemen Kolemen. Data-driven profile prediction for DIII-D. *Nuclear Fusion*, 61(4):046027, 2021.

G. Ambrosino and R. Albanese. Magnetic control of plasma current, position, and shape in tokamaks: a survey on modeling and control approaches. *IEEE Control Systems Magazine*, 25(5): 76–92, 2005. doi: 10.1109/MCS.2005.1512797.

Gaon An, Seungyong Moon, Jang-Hyun Kim, and Hyun Oh Song. Uncertainty-based offline reinforcement learning with diversified q-ensemble. *Advances in neural information processing systems*, 34:7436–7447, 2021.

Laszlo Bardoczi, Alexandra Dudkovskaia, NC Logan, Nathan Jordan Richner, Ashton Clair Brown, James D Callen, Robert John La Haye, and EJ Strait. Tearing stable stationary iter baseline operation in DIII-D. *Nuclear Fusion*, 65(2):026049, 2025.

Zdravko I Botev, Dirk P Kroese, Reuven Y Rubinstein, and Pierre L'ecuyer. The cross-entropy method for optimization. In *Handbook of statistics*, volume 31, pages 35–59. Elsevier, 2013.

M.D. Boyer, K.G. Erickson, B.A. Grierson, D.C. Pace, J.T. Scoville, J. Rauch, B.J. Crowley, J.R. Ferron, S.R. Haskey, D.A. Humphreys, R. Johnson, R. Nazikian, and C. Pawley. Feedback control

of stored energy and rotation with variable beam energy and perveance on DIII-D. *Nuclear Fusion*, 59(7):076004, 2019. ISSN 0029-5515, 1741-4326. doi: 10.1088/1741-4326/ab17f5.

Ian Char, Joseph Abbate, László Bardóczi, Mark Boyer, Youngseog Chung, Rory Conlin, Keith Erickson, Viraj Mehta, Nathan Richner, Egemen Kolemen, et al. Offline model-based reinforcement learning for tokamak control. In *Learning for Dynamics and Control Conference*, pages 1357–1372. PMLR, 2023.

Ian Char, Youngseog Chung, Joseph Abbate, Egemen Kolemen, and Jeff Schneider. Full shot predictions for the DIII-D tokamak via deep recurrent networks. *arXiv preprint arXiv:2404.12416*, 2024.

Jiayu Chen, Wentse Chen, and Jeff Schneider. Bayes adaptive monte carlo tree search for offline model-based reinforcement learning. *CoRR*, abs/2410.11234, 2024.

Jiayu Chen, Yang Fu, and Haomin Bao. Offline RL kit for nuclear fusion. https://github.com/LucasCJYSDL/Offline-RL-Kit-for-Nuclear-Fusion, 2025a. Accessed: 2025-11-06.

Jiayu Chen, Aravind Venugopal, and Jeff Schneider. Policy-driven world model adaptation for robust offline model-based reinforcement learning. *CoRR*, abs/2505.13709, 2025b.

Kurtland Chua, Roberto Calandra, Rowan McAllister, and Sergey Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. *Advances in neural information processing systems*, 31, 2018.

Rory Conlin, Keith Erickson, Joseph Abbate, and Egemen Kolemen. Keras2c: A library for converting keras neural networks to real-time compatible c. *Engineering Applications of Artificial Intelligence*, 100:104182, 2021.

JS DeGrassie, DR Baker, KH Burrell, CM Greenfield, YR Lin-Liu, TC Luce, CC Petty, R Prater, WW Heidbrink, and BW Rice. Plasma rotation and rf heating in DIII-D. In *AIP Conference Proceedings*, volume 485, pages 140–143. American Institute of Physics, 1999.

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.

Yiming Ding, Carlos Florensa, Pieter Abbeel, and Mariano Phielipp. Goal-conditioned imitation learning. *Advances in neural information processing systems*, 32, 2019.

Eric Emdee, Laszlo Horvath, Alessandro Bortolon, and George Wilkie. The influence of rotation and sol drifts on poloidal asymmetries of pedestal fueling. In *APS Division of Plasma Physics Meeting Abstracts*, volume 2024, pages GO06–014, 2024.

Fan Fei, Zhan Tu, Dongyan Xu, and Xinyan Deng. Learn-to-recover: Retrofitting uavs with reinforcement learning-assisted flight control under cyber-physical attacks. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7358–7364. IEEE, 2020.

Scott Fujimoto and Shixiang Shane Gu. A minimalist approach to offline reinforcement learning. *Advances in neural information processing systems*, 34:20132–20145, 2021.

Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In *International conference on machine learning*, pages 2052–2062. PMLR, 2019.

Cristian Galperti, Federico Felici, Trang Vu, Olivier Sauter, F. Carpanese, M. Kong, G. Marceca, A. Merle, A. Pau, A. Perek, F. Pesamosca, M. Baquero-Ruiz, S. Coda, J. Decker, B. Duval, M. Gospodarczyk, A. Karpushov, S. Marchioni, A. Maier, B. Marletaz, A. Segovia, B. Vincent, C. Yildiz, D. Carnevale, N. Ferron, J. Koenders, B. Kool, G. Manduchi, M. Maraschek, P. Milne, A.C. Neto, E. Poli, T. Ravensbergen, M. Reich, N. Rispoli, and F. Sartori. Overview of the TCV digital real-time plasma control system and its applications. *Fusion Engineering and Design*, 208:114640, November 2024. ISSN 09203796.

Tuomas Haarnoja, Aurick Zhou, Kristian Hartikainen, George Tucker, Sehoon Ha, Jie Tan, Vikash Kumar, Henry Zhu, Abhishek Gupta, Pieter Abbeel, et al. Soft actor-critic algorithms and applications. *arXiv preprint arXiv:1812.05905*, 2018.

Nicklas Hansen, Rishabh Jangir, Yu Sun, Guillem Alenyà, Pieter Abbeel, Alexei A Efros, Lerrel Pinto, and Xiaolong Wang. Self-supervised policy adaptation during deployment. *arXiv preprint arXiv:2007.04309*, 2020.

Michael Janner, Justin Fu, Marvin Zhang, and Sergey Levine. When to trust your model: Model-based policy optimization. *Advances in neural information processing systems*, 32, 2019.

S. Kerboua-Benlarbi, R. Nouailletas, B. Faugeras, E. Nardon, and P. Moreau. Magnetic control of west plasmas through deep reinforcement learning. *IEEE Transactions on Plasma Science*, 52 (9):3698–3703, 2024. doi: 10.1109/TPS.2024.3377811.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit q-learning. *arXiv preprint arXiv:2110.06169*, 2021.

Ashish Kumar, Zipeng Fu, Deepak Pathak, and Jitendra Malik. Rma: Rapid motor adaptation for legged robots. *arXiv preprint arXiv:2107.04034*, 2021.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. *Advances in neural information processing systems*, 33:1179–1191, 2020.

Joonho Lee, Jemin Hwangbo, Lorenz Wellhausen, Vladlen Koltun, and Marco Hutter. Learning quadrupedal locomotion over challenging terrain. *Science robotics*, 5(47):eabc5986, 2020.

Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *Journal of Machine Learning Research*, 17(39):1–40, 2016.

Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020.

NC Logan, Brian A Grierson, SR Haskey, SP Smith, O Meneghini, and D Eldon. Omfit tokamak profile data fitting and physics analysis. *Fusion Science and Technology*, 74(1-2):125–134, 2018.

Jiafei Lyu, Xiaoteng Ma, Xiu Li, and Zongqing Lu. Mildly conservative q-learning for offline reinforcement learning. *Advances in Neural Information Processing Systems*, 35:1711–1724, 2022.

Viraj Mehta, Jayson Barr, Joseph Abbate, Mark D Boyer, Ian Char, Willie Neiswanger, Egemen Kolemen, and Jeff Schneider. Automated experimental design of safe rampdowns via probabilistic machine learning. *Nuclear Fusion*, 64(4):046014, feb 2024. doi: 10.1088/1741-4326/ad22f5.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

NJ Richner, L Bardóczi, JD Callen, RJ La Haye, NC Logan, and EJ Strait. Use of differential plasma rotation to prevent disruptive tearing mode onset from 3-wave coupling. *Nuclear Fusion*, 64(10):106036, 2024.

Marc Rigter, Bruno Lacerda, and Nick Hawes. Rambo-rl: Robust adversarial model-based offline reinforcement learning. *Advances in neural information processing systems*, 35:16082–16097, 2022.

John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. Trust region policy optimization. In *International conference on machine learning*, pages 1889–1897. PMLR, 2015.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

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, 2024.

Vaisakh Shaj, Dieter Buchler, Rohit Sonker, Philipp Becker, and Gerhard Neumann. Hidden parameter recurrent state space models for changing dynamics scenarios. *arXiv preprint arXiv:2206.14697*, 2022.

Ricardo Shousha, Jaemin Seo, Keith Erickson, Zichuan Xing, SangKyeun Kim, Joseph Abbate, and Egemen Kolemen. Machine learning-based real-time kinetic profile reconstruction in DIII-D. *Nuclear Fusion*, 64(2):026006, 2023.

Rohit Sonker, Alexandre Capone, Andrew Rothstein, Hiro Josep Farre Kaga, Egemen Kolemen, and Jeff Schneider. Multi-timescale dynamics model bayesian optimization for plasma stabilization in tokamaks. In *Forty-second International Conference on Machine Learning*, 2025.

Tong Su, Tong Wu, Junbo Zhao, Anna Scaglione, and Le Xie. A review of safe reinforcement learning methods for modern power systems. *Proceedings of the IEEE*, 2025.

Yihao Sun, Jiaji Zhang, Chengxing Jia, Haoxin Lin, Junyin Ye, and Yang Yu. Model-bellman inconsistency for model-based offline reinforcement learning. In *International Conference on Machine Learning*, volume 202, pages 33177–33194, 2023a.

Yihao Sun, Jiaji Zhang, Chengxing Jia, Haoxin Lin, Junyin Ye, and Yang Yu. Model-bellman inconsistency for model-based offline reinforcement learning. In *International Conference on Machine Learning*, pages 33177–33194. PMLR, 2023b.

Brendan D. Tracey, Andrea Michi, Yuri Chervonyi, Ian Davies, Cosmin Paduraru, Nevena Lazic, Federico Felici, Timo Ewalds, Craig Donner, Cristian Galperti, Jonas Buchli, Michael Neunert, Andrea Huber, Jonathan Evens, Paula Kurylowicz, Daniel J. Mankowitz, and Martin Riedmiller. Towards practical reinforcement learning for tokamak magnetic control. *Fusion Engineering and Design*, 200:114161, 2024. ISSN 0920-3796. doi: https://doi.org/10.1016/j.fusengdes.2024.114161.

Michael L Walker, Peter De Vries, Federico Felici, and Eugenio Schuster. Introduction to tokamak plasma control. In *2020 American Control Conference (ACC)*, pages 2901–2918. IEEE, 2020.

ML Walker, DA Humphreys, and JR Ferron. Multivariable shape control development on the DIII-D tokamak. In *17th IEEE/NPSS Symposium Fusion Engineering (Cat. No. 97CH36131)*, volume 1, pages 556–559. IEEE, 1997.

M.L. Walker, J.R. Ferron, D.A. Humphreys, R.D. Johnson, J.A. Leuer, B.G. Penaflor, D.A. Piglowski, M. Ariola, A. Pironti, and E. Schuster. Next-generation plasma control in the DIII-D tokamak. *Fusion Engineering and Design*, 66-68:749–753, September 2003. ISSN 09203796. doi: 10.1016/S0920-3796(03)00295-3.

Allen M Wang, Alessandro Pau, Cristina Rea, Oswin So, Charles Dawson, Olivier Sauter, Mark D Boyer, Anna Vu, Cristian Galperti, Chuchu Fan, et al. Learning plasma dynamics and robust rampdown trajectories with predict-first experiments at TCV. *Nature Communications*, 16(1): 8877, 2025.

William Wehner, Justin Barton, and Eugenio Schuster. Toroidal rotation profile control for the DIII-D tokamak. In *2015 American Control Conference (ACC)*, pages 3664–3669. IEEE, 2015.

William Wehner, Justin Barton, and Eugenio Schuster. Combined rotation profile and plasma stored energy control for the DIII-D tokamak via mpc. In *2017 American Control Conference (ACC)*, pages 4872–4877. IEEE, 2017.

GJ Wilkie, Florian Laggner, R Hager, A Rosenthal, S-H Ku, R Michael Churchill, Laszlo Horvath, Choong Seock Chang, and Alessandro Bortolon. Reconstruction and interpretation of ionization asymmetry in magnetic confinement via synthetic diagnostics. *Nuclear Fusion*, 64(8):086028, 2024.

Grady Williams, Nolan Wagener, Brian Goldfain, Paul Drews, James M. Rehg, Byron Boots, and Evangelos A. Theodorou. Information theoretic mpc for model-based reinforcement learning. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1714–1721, 2017. doi: 10.1109/ICRA.2017.7989202.

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 control in hl-3 tokamak. *Communications Physics*, 8(1):393, 2025.

Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Y Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. Mopo: Model-based offline policy optimization. *Advances in Neural Information Processing Systems*, 33:14129–14142, 2020.

Tianhe Yu, Aviral Kumar, Rafael Rafailov, Aravind Rajeswaran, Sergey Levine, and Chelsea Finn. Combo: Conservative offline model-based policy optimization. *Advances in neural information processing systems*, 34:28954–28967, 2021.

6. Appendix

6.1. RPNN Training

6.1.1. STATE AND ACTUATOR SPACE -

The dynamics model (an ensemble of RPNN networks) models the transitions on a larger state and action space, in comparison to the policy. This is shown in table 2. Variables marked under Actuators and States are used in dynamics modeling. Only a subset of the states is provided as observations to the policy (marked under observations), and a subset of actuators is controlled by the policy (marked under actions). We do this to simplify the learning process and also because other actuators have different purposes other than controlling the selected task. Note that last action taken is provided in the observation space of the policy.

6.1.2. NETWORK ARCHITECTURE AND TRAINING DETAILS -

Network Architecture :

- **Encoder:**
  - Fully Connected (FC) layer: `input_dim` $\times$ 512
  - FC layer: 512 $\times$ 512
- **Memory Unit:**
  - Gated Recurrent Unit (GRU) block: 512 $\times$ 256
- **Decoder (with residual connections between FC layers):**
  - FC layer: 256 $\times$ 512
  - FC layers: 512 $\times$ 512 (repeated 8 times)
  - FC layer: 512 $\times$ 128
- **Output Heads:**
  - Mean head: 128 $\times$ `output_dim`
  - Log-variance head: 128 $\times$ `output_dim`

Signal Group	Signals	Actuator	State	Action	Observation
Scalar States	β_N (Normalized Plasma Pressure)	✗	✓	✗	✗
	l_i (Internal Inductance)	✗	✓	✗	✗
	Line Averaged Density	✗	✓	✗	✗
	Loop Voltage	✗	✓	✗	✗
	MHD Stored Energy	✗	✓	✗	✗
Profile States	Rotation	✗	✓	✗	✓
	Density	✗	✓	✗	✗
	Ion Temperature	✗	✓	✗	✗
	Electron Temperature	✗	✓	✗	✗
	Pressure	✗	✓	✗	✗
	Safety Factor q	✗	✓	✗	✗
Shape Variables	Elongation, Upper Triangularity, Bottom Triangularity, a_{minor}, Radial and vertical positions of magnetic axis	✓	✗	✗	✗
Neutral Beam Variables	Power Injected	✓	✗	✓	✓
	Torque Injected	✓	✗	✓	✓
Gas Puffing	GasA voltage	✓	✗	✓	✓
Electron Cyclotron Heating	ECH Total Power	✓	✗	✓	✓
Other Actuators	Current Target, Toroidal Field	✓	✗	✗	✗
Targets	Rotation Target (t), Rotation Target (t+10), Current Error Terms	✗	✗	✗	✓
Total Dimensions		**12D**	**25D**	**4D**	**20D**

Table 2: Plasma signals and how they are used as State and Actuators for dynamics modelling. Policy Observation and action space variables are also shown.

The network predicts the parameters of a probability distribution, and is trained in two stages. The First stage correponds to training with Mean Square error loss (MSE) and the second stage correponds to training only the log variance head with negative log likelihood loss. We use the Adam optimizer with a learning rate of 3×10^{-4} and a weight decay of 1×10^{-3} in both stages. A 25 model ensemble is trained by bootstrapping on the dataset. We train for 1000 epochs with early stopping (patience = 250 epochs) based on performance on a validation set comprising 10% of the total data.

6.2. Offline RL Benchmarking

Our complete set of results for all algorithms tested are provided in table 3. This includes RMSE tracking at various points along the profile from the core $\psi_n = 0$ to the edge $\psi_n = 1$ of the plasma.

Table 3: Performance of Offline RL algorithms on Rotation profile tracking error. RMSE values (with standard error in parenthesis) are shown for the whole profile tracking and also across ψ_n values which vary from 0 (core) to 1 (edge) of plasma profile (lower is better). These results were computed across shots 161409, 161410, 161412 in simulation with 10 seeds across each shot.

Algorithm	Profile	$\psi_n = 0.09$	$\psi_n = 0.18$	$\psi_n = 0.39$	$\psi_n = 0.58$	$\psi_n = 0.79$	$\psi_n = 0.88$
TD3BC	58.73 (0.51)	62.64 (3.17)	57.54 (2.62)	49.05 (2.11)	43.73 (1.85)	39.36 (1.39)	24.85 (0.39)
PPO	29.5 (0.46)	45.77 (2.17)	39.05 (1.42)	30.16 (0.77)	26.84 (0.67)	26.19 (0.62)	21.12 (0.29)
GCIL	36.48 (0.37)	50.15 (2.09)	43.8 (1.87)	35.32 (1.38)	31.86 (1.06)	30.71 (0.83)	23.47 (0.26)
CQL	80.79 (0.25)	73.83 (1.13)	67.58 (0.81)	57.14 (1.03)	50.84 (0.97)	46.04 (0.59)	29.29 (0.21)
EDAC	32.07 (0.66)	47.09 (3.85)	41.58 (2.8)	34.2 (1.65)	30.29 (1.3)	27.39 (1.32)	19.31 (0.62)
MCQ	44.64 (0.44)	55.89 (2.59)	49.94 (2.53)	40.48 (2.02)	35.67 (1.45)	33.1 (1.07)	23.11 (0.42)
COMBO	51.46 (0.62)	59.96 (3.57)	53.09 (2.6)	43.05 (1.55)	38.49 (1.26)	36.07 (1.3)	24.96 (0.72)
IQL	54.52 (0.78)	61.46 (5.43)	55.12 (4.94)	45.0 (3.24)	39.98 (2.2)	37.69 (1.79)	26.84 (0.91)
MOPO	84.63 (0.72)	76.81 (4.38)	68.82 (3.57)	56.15 (2.37)	49.78 (1.84)	46.42 (1.7)	31.93 (1.02)
MOBILE	32.99 (0.4)	45.64 (2.67)	43.0 (1.79)	38.02 (0.98)	33.77 (0.77)	29.82 (0.55)	20.13 (0.28)
BAMCTS	57.43 (0.71)	62.69 (4.0)	56.27 (2.93)	46.74 (2.12)	41.86 (1.91)	38.61 (1.64)	25.63 (0.54)
RAMBO	40.05 (0.43)	51.3 (2.52)	45.99 (2.03)	39.14 (1.49)	36.14 (1.33)	34.28 (1.26)	24.43 (0.62)
ROMBRL	91.44 (0.7)	79.59 (4.18)	72.05 (3.26)	59.71 (2.16)	52.56 (1.7)	47.16 (1.38)	28.88 (0.57)

7. Profile Views

In this section, we present full rotation profile views across time. Fig.5 shows the profile variation when the first target change occurs. This target change corresponds to a decrease in profile at t = 3s. Fig.6 shows the profile changes that occur during the second target change at t = 4.5 s. At this point, more variations occur in the plasma, which can be observed in the figure. The rise in core rotation followed by a drop is attributed to unknown plasma effects. We examined the presence of tearing mode activity and large sawtooth instabilities, however, no clear evidence of either was detected.

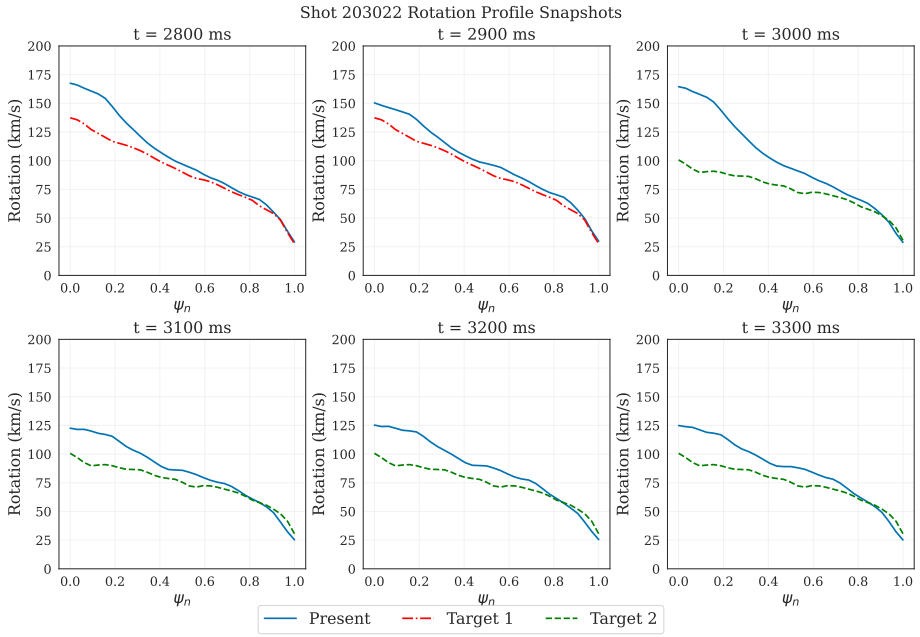

Figure 5: Full RTCAKENN Rotation profile at various time slices during the first target change. The change in target line (red to green) occurs at t=3s. The real rotation profile (blue) starts moving down to get closer to the updated target.

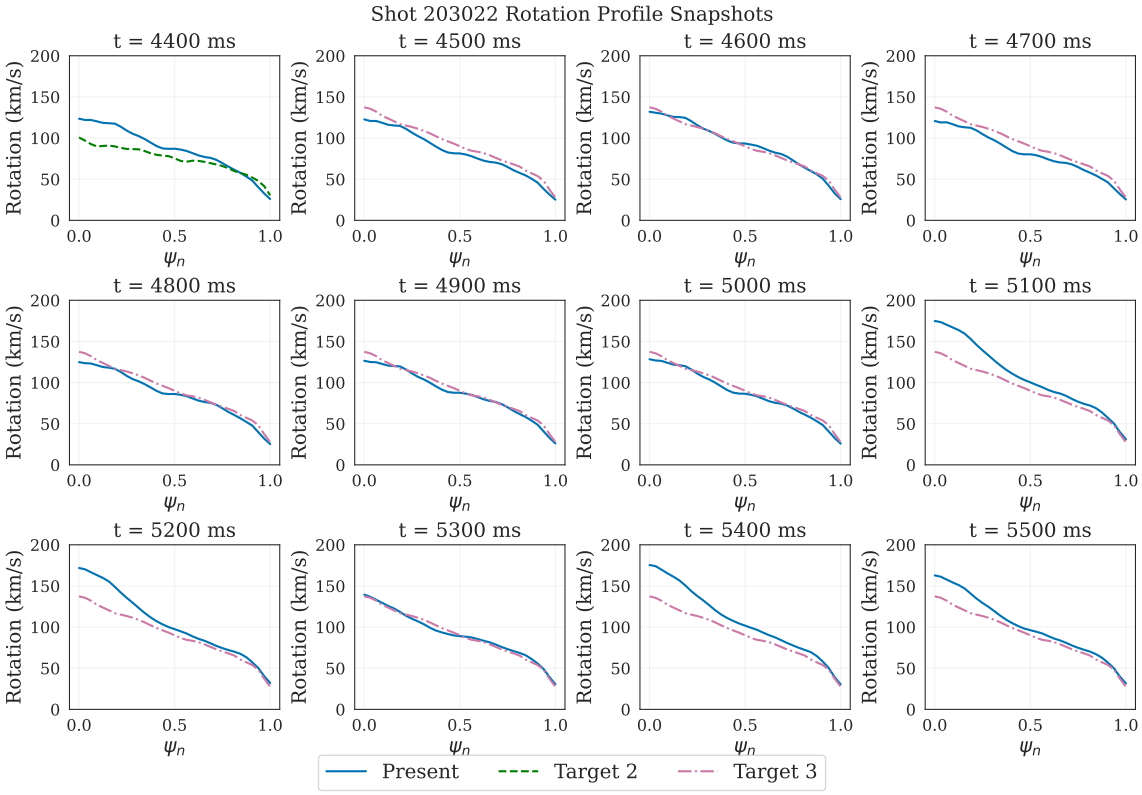

Figure 6: Full RTCAKENN rotation profiles at various time slices during the second target change. The change in target line (green to purple) occurs at t=4.5s. This new target matches the present profile well, hence drastic changes in actuators are not expected. A sudden rise in core rotation is observed, which later drops. This is attributed to either unknown plasma effects or noise in signals.

8. Gas Flow

In this section, plots showing the gas flow rate into the plasma are shown in Fig. 7. These plots show the actual amount of gas flowing into the plasma, corresponding well to the gas valve voltage, which the policy controls.

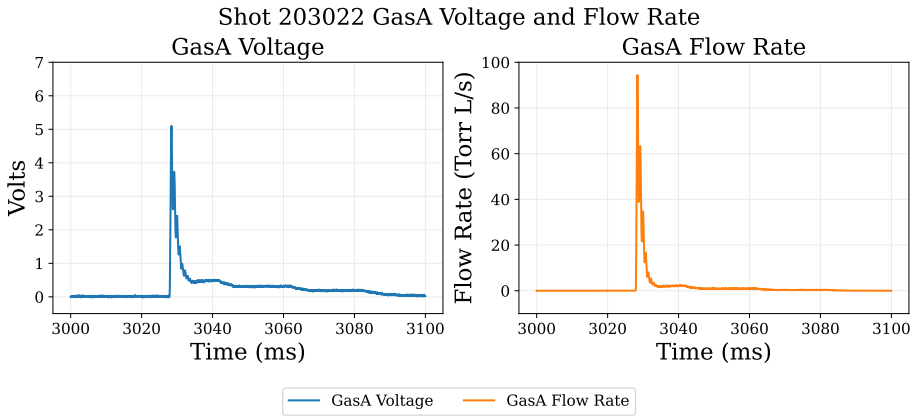

Figure 7: Gas Voltage and the corresponding gas flow rate increase at approx t=3s, corresponding to the target change.