DYNABO: DYNAMIC MODEL BAYESIAN OPTIMIZATION FOR TOKAMAK CONTROL

Anonymous authors
Paper under double-blind review

ABSTRACT

Despite recent advances, state-of-the-art machine learning algorithms struggle considerably with control problems where data is scarce relative to model complexity. This problem is further exacerbated if the system changes over time, making past measurements less useful. While tools from reinforcement learning, supervised learning, and Bayesian optimization alleviate some of these issues, they do not address all of them at once. With these drawbacks in mind, we present a multi-scale Bayesian optimization for fast and data-efficient decision-making. Our pipeline combines a high-frequency data-driven dynamics model with a low-frequency Gaussian process, resulting in a high-level model with a prior that is specifically tailored to the dynamics model setting. By updating the Gaussian process during Bayesian optimization, our method adapts rapidly to new data points, allowing us to quickly process current high-quality data which is more representative of the system than past data. We apply our method to avoid tearing instabilities in a tokamak plasma, a control problem where modeling is difficult, and hardware changes potentially between experiments. Our approach is validated through offline testing on historical data and live experiments on the DIII-D tokamak. On the historical data, we show that our method outperforms a naive decision-making algorithm based exclusively on a recurrent neural network and past data. The live experiment corresponds to a high performance plasma scenario with very high likelihood of instabilities. Despite this base configuration, we achieve a 75% success rate in the live experiment, which represents an improvement of over 300% compared to historical data.

1 INTRODUCTION

Controlling real-world systems is generally a difficult task, even when powerful machine-learning tools are employed: nonlinearities are often pronounced, data is scarce, and safety issues impose severe limitations. A prime example of these issues is tokamak control, where good models are unavailable, safety is paramount, and instabilities are notoriously hard to control. These issues are further complicated by the fact that hardware configurations in tokamak change on a regular basis, making a model trained on past data even less reliable. However, despite these challenges, designing good control policies for tokamaks is highly desirable due to their promise to generate abundant clean energy via nuclear fusion.

In many real-world settings, model-free reinforcement learning is a promising solution and has seen successful applications (He et al., 2024; Kumar et al., 2021; Lee et al., 2020). However, most of these methods rely on a prohibitive amount of policy rollouts for training, which is typically only achievable with reliable simulation environments. In complex environments like tokamaks, this is particularly problematic, as operation costs typically only permit a handful of rollouts, and existing simulators do not reflect the true dynamics for many aspects of the plasma (Char et al., 2023a). Offline RL, seeks to overcome these issues by directly learning a policy from offline data which conservatively stays within bounds of observed data (Levine et al., 2020). However, the performance of offline RL methods depends crucially on high-quality expert data that contains advantageous states. If these are not present, then offline RL can suffer from extrapolation errors (Fujimoto et al., 2019). This is a major drawback for tokamak control, where significant exploration and improvement are still required to achieve energy production. Moreover, even offline RL is affected by the sim2real problem which is described in detail below.

Alternatively, model-based reinforcement learning offers a solution where dynamics models are trained from historic data and rollouts from the model are then used for policy learning or planning (Deisenroth & Rasmussen, 2011; Chua et al., 2018; Kaiser et al., 2019). In the past, machine learning algorithms have been used to directly model plasma dynamics (Char et al., 2023b; Abbate et al., 2021; Boyer et al., 2021). Reinforcement learning policies have also been trained in models trained solely on fusion data (Char et al., 2023a; Wakatsuki et al., 2023; Degrave et al., 2022). However, the performance of these approaches crucially hinges on the assumption that the data faithfully captures the model at test time. This is problematic in the case of tokamak dynamics, where time-dependent model changes cannot be neglected. Though this issue can be potentially addressed by updating the model with new data, the scarcity of experiments implies that too little data is typically produced to reliably update the model.

In low-dimensional settings, the obstacles posed by conventional RL methods can potentially be addressed by Bayesian optimization (BO). BO is a data-efficient tool for optimizing black box functions (Garnett, 2023). By quantifying model uncertainty, BO achieves a tradeoff between exploration and exploitation, leading to fast convergence in many practical settings (Shalloo et al., 2020; Shields et al., 2021). In the case of tokamak control, BO has been used, e.g., to control the rampdown of a real tokamak (Mehta et al., 2024), and to control neutral beams in a tokamak simulator (Char et al., 2019). However, the work of Mehta et al. (2024) does not address critical plasma instabilities, whereas Char et al. (2019) relies on a simulator. Moreover, these methods use a poorly specified prior and require an extensive amount of experiments to perform well.

Motivated by the strengths and shortcomings of existing machine learning-based approaches for tokamak control, we design a novel approach that combines a dynamic model predictor and Bayesian Optimization. Our approach employs a multi-scale approach: a recurrent probabilistic neural network models the high-frequency model dynamics, while a Gaussian process models the effect of low-frequency marginal statistics on the dynamics. After adequate pre-processing, we use historical data to train both models, where the dynamic model serves as a prior for the Gaussian process. Additionally, by leveraging physics-informed assumptions, we design a low-dimensional state space for the Gaussian process. This naturally leads to a contextual Bayesian optimization algorithm tailored to the task at hand, allowing it to find stabilizing actions in a highly data-efficient manner. Moreover, due to its ability to perform fast updates, it allows us to efficiently leverage small batches of data collected during experiments to best inform new decisions on the fly.

We test our approach on a large dataset from past tokamak experiments, where we can quickly identify stable configurations, outperforming a naive approach based exclusively on the recurrent neural network model. Furthermore, we apply our approach to find stabilizing actions for a high performing plasma scenario in the DIII-D tokamak. High performing plasma scenarios need to maintain high temperature and pressures for increased energy, hence, they are more unstable. Our method was able to find stabilizing ECH actuator values in six of eight experiments despite changes to other actuators, a 300% improvement compared to historical experiments with the same configuration.

Our paper is structured as follows: first, we provide some necessary background to nuclear fusion, and define our problem mathematically. Then we discuss our complete pipeline and methodology, followed by the results and analysis on offline historical data and live experiments on a Tokamak reactor. Finally, we provide conclusions and discuss opportunities for future work. Additional details are provided in the Appendix.

2 Background and Problem Statement

In this section, we first provide some background on nuclear fusion and then present the formal problem statement.

2.1 Nuclear Fusion

Nuclear fusion is seen as a promising solution for clean, limitless energy, producing no high-level radioactive waste. Among the fusion technologies, tokamaks are the most advanced, using magnetic fields to confine hot plasma to enable fusion conditions. Many countries have invested in tokamak research facilities and currently more 35 nations are collaborating to build ITER, a global project

aiming to demonstrate the viability of large-scale commercial fusion reactors (Mohamed et al., 2024; Shimada et al., 2007).

However, one of the key challenges in tokamak development is plasma disruptions, which can cause severe damage to reactor walls and components, particularly in larger reactors like ITER (Schuller, 1995; Lehnen et al., 2015). These disruptions often stem from tearing mode instabilities (or tearing modes), where magnetic islands form, leading to energy loss and instability. Electron Cyclotron Heating (ECH) is one of the most effective methods to counteract tearing instability by driving localized currents at the site of instability (Gantenbein et al., 2000; Kolemen et al., 2014).

Prior work has been done on avoiding tearing instability with predictive models using real-time control (Fu et al., 2020) and reinforcement learning (Seo et al., 2024). However, these methods reduce neutral beam power and add torque to stabilize the plasma . This is not feasible, as reducing beam power leads to lower confinement energy, in turn decreasing the total energy output of the tokamak. On the other hand, adding torque to large tokamaks is another challenging issue. Use of ECH to avoid tearing instability directly eliminates the causes of these instabilities. This is also why future reactors will heavily depend on ECH for stability, eg. ITER will have over 40 gyrotrons to deliver ECH as a primary method to stabilize the plasma.

In this work, we aim to control ECH profiles to avoid tearing instability (or modes) in high q_{min} tokamak scenarios. An ECH profile represents the heating achieved by the gyrotrons across the cross section of the plasma. This can be seen in fig 3. High q_{min} is a scenario that supports long duration steady-state plasma operations, making it crucial for future commercial fusion reactors. We also focus our attention on 2-1 tearing instability, a type which is the most common and significantly disruptive.

2.2 Problem Statement

We treat the tokamak dynamics as an unknown discrete-time stochastic system

$$s_{t+1} \sim \Pi_{s_t, a_t}, \tag{1}$$

with states $s_t \in \mathcal{S}$ and actions $a_t \in \mathcal{A}$, and the probability of a tearing mode occurring follows a Bernoulli distribution, parameterized by the tokamak states and actions

$$T_t \sim \text{Bernoulli}(p(s_t, a_t)). \tag{2}$$

Of the state variables describing the plasma, the most important for our approach is the normalized plasma pressure $\beta_{N,t} \in s_t$. A full description of the state space is given in the appendix. The action vector can be decomposed into three different sub-vectors

$$a_t \coloneqq \left[a_t^f,\ a_t^c,\ a_t^g\right] \tag{3}$$

as follows. The actions a_t^f correspond to feedforward inputs specified before the experiment. These correspond, e.g., to gas flows, plasma density, and shape controls. They are typically picked manually based on the success of previous experiments. The actions a_t^c are part of a feedback control loop that aims to stabilize the normalized plasma pressure $\beta_{N,t} \in s_t$, arguably one of the most important quantities since it measures the efficiency of plasma confinement relative to the magnetic field strength. The third set of actions a_t^g corresponds to gyrotron angles, operated at constant power, which we use to keep the tearing instability from occurring. The gyrotrons operate on the plasma by generating an ECH profile $a_t^{\text{ech}} = h(a_t^g)$. Unlike a_t^f and a_t^c, the number of gyrotrons, i.e., the dimension of a_t^g, potentially changes between each individual experiment. This is due to various reasons, e.g., due to hardware issues or because some gyrotrons might be required for other tasks, such as elm suppression or density control (Hu et al., 2024; Ono et al., 2024).

This paper considers the case where the gyrotron angles a_t^g are kept fixed throughout each experiment roullout, i.e., $a_0^g = a_1^g = ... = a_\tau^g =: a^g$, where τ is the length of the rollout horizon. This is a common operating mode and also a design choice, which we make because we need to search as efficiently as possible within the action space, an impossible task if its dimension is too large. The feedforward actions a_t^f and the target β_N, which defines the set-point for a_t^c, are specified beforehand and can change between rollouts. Our goal is then to select a^g separately for each experiment such that the probability of encountering a tearing mode $T_t = 1$ is minimized over the full rollout horizon.

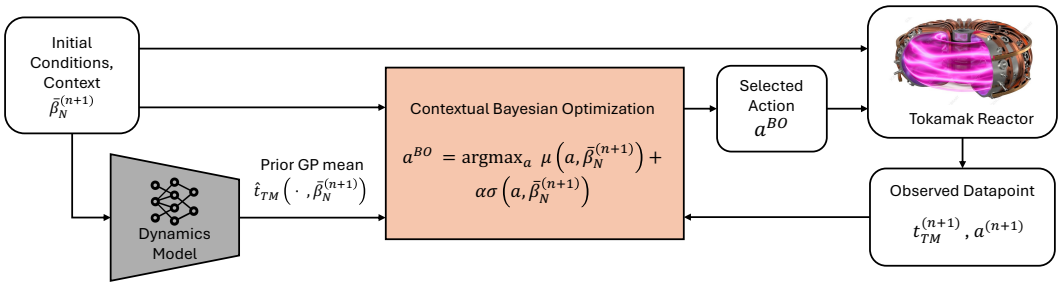

Figure 1: Overall Pipeline to generate trajectory actions. Initial conditions and feedforward actuators are used by RPNN to generate rollouts through which we compute the prior mean of the objective function (time to tearing instability). Our Bayesian optimization algorithm uses this to optimize for actions (ECH). Noisy outputs from the Tokamak are then to update the Gaussian process model used for Bayesian optimization.

3 METHODOLOGY

Our complete pipeline is illustrated in Fig. 1. On a high level, the process is as follows - We model the system at two different time scales to inform the choice of actuator commands for each experiment. At a smaller, more granular time scale, we use a recurrent probabilistic neural network model (RPNN) to estimate the high-frequency behavior during each experiment. The coarser model corresponds to a Gaussian process model and is trained to predict the behavior of the system based on marginal statistics from experimental observations and RPNN predictions of the objective function, which act as a prior mean. In this case, the objective function is the time-to-tearing instability. Given the target values (the normalized plasma pressure β_N), we leverage the Gaussian process to select actions (ECH profiles) in a low-dimensional space via Bayesian optimization. The resulting action (heating profile) is then applied to the tokamak. In the following sections, we describe the high-frequency RPNN, then the GP, and end with the full Bayesian optimization pipeline.

3.1 RECURRENT PROBABILISTIC NEURAL NETWORKS AND BINARY CLASSIFCATION

We employ a Recurrent Probabilistic Neural Network (RPNN) to model the high-frequency behavior of the tokamak. An RPNN has a Gated Rectifier Unit (GRU) cell, which stores information about past states and actions. The advantage of including a memory unit is that it allows us to model any unobserved variables that influence the state. To bypass the issue that the number of gyrotrons differs for each rollout in the training dataset, we assume that the resulting heating profiles a_t^{ech} can be controlled directly, allowing us to disregard a_t^g both in training and testing. When carrying out experiments, we then project a_t^{ech} onto a_t^g, which can be done for an arbitrary number of gyrotrons, i.e., for an arbitrary dimension of a_t^g.

Given s_t and a_t as inputs, our RPNN outputs a distribution over s_{t+1} as mean μ, variance Σ^2. The mean and variance specify a multivariate normal distribution, which we employ to approximate the system dynamics

$$\mathcal{N}(\eta(s_t, a_t), \Sigma^2(s_t, a_t)) \approx \Pi_{s_t, a_t}. \tag{4}$$

In addition to the RPNN, we train a classifier, which we call the tearing mode predictor h to predict the probability of a tearing mode occurring

$$h(s_t, a_t) \approx \text{Bernoulli}(p(s_t, a_t)). \tag{5}$$

3.2 GAUSSIAN PROCESS MODEL

Exclusively using RPNN for experimental design is challenging for various reasons. Although the RPNN accurately captures some of the tokamak behavior, the resulting predictions often exhibit significant errors, largely due to the sim2real gap caused by time-dependent fluctuations in the environment variables, e.g., due to maintenance or hardware changes provoked by previous experiments.

Furthermore, retraining the RPNN between experiments and using it to select gyrotron angles a^g is virtually impossible because the newly collected data is too small and because we only have a couple of minutes between experiments.

We address the abovementioned issues by employing a Gaussian process (GP) model, a nonparametric model that is very data-efficient, especially in low-dimensional spaces Deisenroth & Rasmussen (2011). A GP corresponds to an infinite collection of random variables, of which any finite number is jointly normally distributed. To fully leverage the strengths of GP models, we need to carefully summarize the information collected between experiments before training the GP. This is done as follows.

First, we make the assumption that the normalized plasma pressure β_N is independent of the ECH profile a_q. This is a reasonable assumption because β_N is largely determined by neutral beams, which are controlled through the feedback variables a_t^c. We then approximate the feedforward and feedback control actions $a_t^f, \dots, a_\tau^f$ and $a_1^c, \dots, a_\tau^c$ by assuming that they are uniquely specified by the target normalized plasma pressure, denoted by $\bar{\beta}_N$. This choice is partly justified because the feedforward and feedback control actions are often primarily informed by a target normalized plasma pressure. A further approximation we make is to project the ECH profile a^{ech} to a Gaussian curve, parametrized by the three-dimensional vector a_q containing the center, width, and height of the Gaussian curve. Finally, we employ the GP to predict the time-to-tearing mode t_{TM}, which we use as a proxy for the probability of a tearing mode occurring. The rationale behind this choice is twofold. First, a scenario where tearing instability occur late implies a higher degree of stability than a scenario where they occur earlier. Moreover, it allows us to use the GP in a regression setting, where GPs are strongest and best understood. The GP inputs are thus $\bar{\beta}_N$ and a_q, whereas the output is t_{TM}.

The GP is fully specified by a prior mean function m and a kernel k that specifies the similarity between training inputs. In this work, we employ a squared-exponential kernel k, which is appropriate for approximating most continuous functions. The mean function m corresponds to the average $\hat{t}_{\mathsf{TM}}$ predicted by the RPNN and tearing mode predictor,

$$\hat{t}_{\mathsf{TM}}(\bar{\beta}_N, a_q) \coloneqq \mathbb{E}\left(\arg\min_t T_t \;\middle|\; T_t = 1,\;\; T_t \sim h(s_t, a_t), \quad s_{t+1} \sim \mathcal{N}\left(\eta(s_t, a_t), \Sigma^2(s_t, a_t)\right) \right), \tag{6}$$

where we use the Gaussian curve specified by a_q to choose the ECH component of the actions $a_1, \dots, a_\tau$. The feedforward and control actions a_t^c and a_t^f components of the actions are chosen based on the target $\bar{\beta}_N$ for the experiment. Given training data,

$$\mathcal{D}_n = \{\bar{\beta}_N^{(i)}, a_q^{(i)}, t_{\mathsf{TM}}^{(i)}\}_{i=1,\dots,n},$$

obtained after appropriate pre-processing, we can compute the posterior distribution of t_{TM} for arbitrary test inputs $\bar{\beta}_N^*, a_q^*$, which corresponds to a normal distribution mean and covariance

$$\mu_n(\bar{\beta}_N^*, a_q^*) = \hat{t}_{\mathsf{TM}}(\bar{\beta}_N^*, a_q^*) + k_*^\top (K + \sigma^2 I)^{-1} \Delta_n, \tag{7}$$

$$\sigma_n^2(\bar{\beta}_N^*, a_q^*) = k_{**} - k_*^\top (K + \sigma_{\text{no}}^2 I)^{-1} k_* + \sigma_{\text{no}}^2, \tag{8}$$

where σ_{no}^2 is the noise variance, $[k_*]_i = k(\bar{\beta}_N^*, a_q^*, \bar{\beta}_N^{(i)}, a_q^{(i)})$, $[K]_{ij} = k(\bar{\beta}_N^{(i)}, a_q^{(i)}, \bar{\beta}_N^{(j)}, a_q^{(j)})$, $k_{**} = k(\bar{\beta}_N^*, a_q^*, \bar{\beta}_N^*, a_q^*)$. The vector $[\Delta_n]_i = t_{\mathsf{TM}}^{(i)} - \hat{t}_{\mathsf{TM}}(\bar{\beta}_N^{(i)}, a_q^{(i)})$ contains the difference between the observed and the predicted time-to-tearing mode. In practice, the posterior variance σ_n^2 is typically small when evaluated in distribution and larger when out of distribution. Hence, intuitively, the posterior GP mean μ_n can be viewed as the predictive model, whereas σ_n^2 quantifies model uncertainty. This distinction for understanding Bayesian optimization, which is introduced in the next section.

3.3 Contextual Bayesian Optimization with Noisy Inputs

Contextual Bayesian optimization is a data-efficient tool that leverages GPs to optimize black-box functions. Given a context that specifies the environment, it optimizes an acquisition function that carefully balances exploration versus exploitation. By recursively updating the acquisition function

after every observation, it gradually becomes more confident about its predictions, resulting in convergence. In every experiment, we treat the target normalized plasma pressure $\bar{\beta}_N^{(n+1))}$, specified before the experiment, as the context and choose the ECH profile by optimizing the so-called upper confidence bound (UCB) acquisition function

$$a_q^{\text{BO}} = \arg\max_{a_q} \mu_n(a_q, \bar{\beta}_N^{(n+1))}) + \alpha\sigma_n(a_q, \bar{\beta}_N^{(n+1)}), \tag{9}$$

where α balances exploration and exploitation. In conventional BO methods, the next step consists of setting $a_q^{(n+1)} = a_q^{\text{BO}}$, measuring the time-to-tearing mode $t_{\mathsf{TM}}^{(i+1)}$, and updating the GP accordingly. However, in our setting there is the added challenge that the predicted plasma the desired ECH profile corresponding to a_q^{BO} is not reproduced exactly. This is due to the potentially changing number of available gyrotrons, actuator noise, and unmodeled disturbances. Number of gyrotrons is variable from experiment-to-experiment. To alleviate this, we instead measure the ECH profile obtained during the experiment and use it to determine $a_q^{(n+1)}$ before updating the GP model. Formally, this is equivalent to standard contextual BO where the GP inputs a_q in Equation 9 are perturbed by unknown noise.

4 Results

In this section, we present results from offline tests using historical data, followed by results from experiments at the General Atomics DIII-D Tokamak Fusion Facility. In all experiments, we use a fixed RPNN, trained using $15,000$ one-step state transition observations collected between 2010 and 2019 at the DIII-D tokamak.

Through our analysis of offline and online experiments, we aim to answer the following questions:

1. Is our Bayesian Optimization-based algorithm better than an open-loop planner exclusively using the RPNN dynamics model and tearing mode predictor?
2. Can Bayesian Optimization find heating profiles a_q that avoid tearing instability in a real experimental setting where we have changing configurations and only a handful of trials?

We address question 1 by using historical data from the DIII-D tokamak, which we split into a small training data set and a large holdout dataset that can be queried. Afterward, we address question 2 with results from live experiments on the DIII-D Tokamak. As we show in the following, both questions have an affirmative answer.

4.1 Offline Data Analysis

To confirm if Bayesian optimization outperforms an RPNN-based planner, we set up an artificial experimental environment using historical data $\mathcal{D}^H$ collected between 2012 and 2023, which features a varying number of active gyrotrons. The data set $\mathcal{D}^H$ is described in detail in A.1. Each element in $\mathcal{D}^H$ corresponds to an experiment with a different configuration, i.e., a potentially different target normalized plasma pressure β_N. The task is then to find ECH values that avoid tearing instabilities.

The elements in the data set $\mathcal{D}^H$ all have different configurations, each one corresponding to a different $\bar{\beta}_N$. To simulate an environment where we can choose different ECH profiles a_q, given a constant, pre-specified, $\bar{\beta}_N$ we proceed as follows. We split $\mathcal{D}^H$ into multiple bins, such that each bin i contains all data corresponding to an average normalized plasma pressure within the interval $\bar{\beta}_N \in (\bar{\beta}_N - \epsilon, \bar{\beta}_N + \epsilon)$. We select $\epsilon = 0.04$, and the bin centers $\bar{\beta}_N^i < \bar{\beta}_N^{i+1}$ such that the full data set $\mathcal{D}^H$ is contained within the bins. We then sample a bin randomly and allow an arbitrary data point within it to be selected. Since ϵ is small, this corresponds to approximately choosing a_q from a finite set given a constant β_N.

When employing our approach, we first draw $n = 10$ training samples from $\mathcal{D}^H$ to fit the GP hyperparameters and to compute the GP posterior. We then perform BO using the setup described above, iteratively updating the GP and the UCB acquisition function at every step. In the case of the RPNN planner, we simply select the element in the bin with the highest predicted time-to-tearing mode at every iteration.

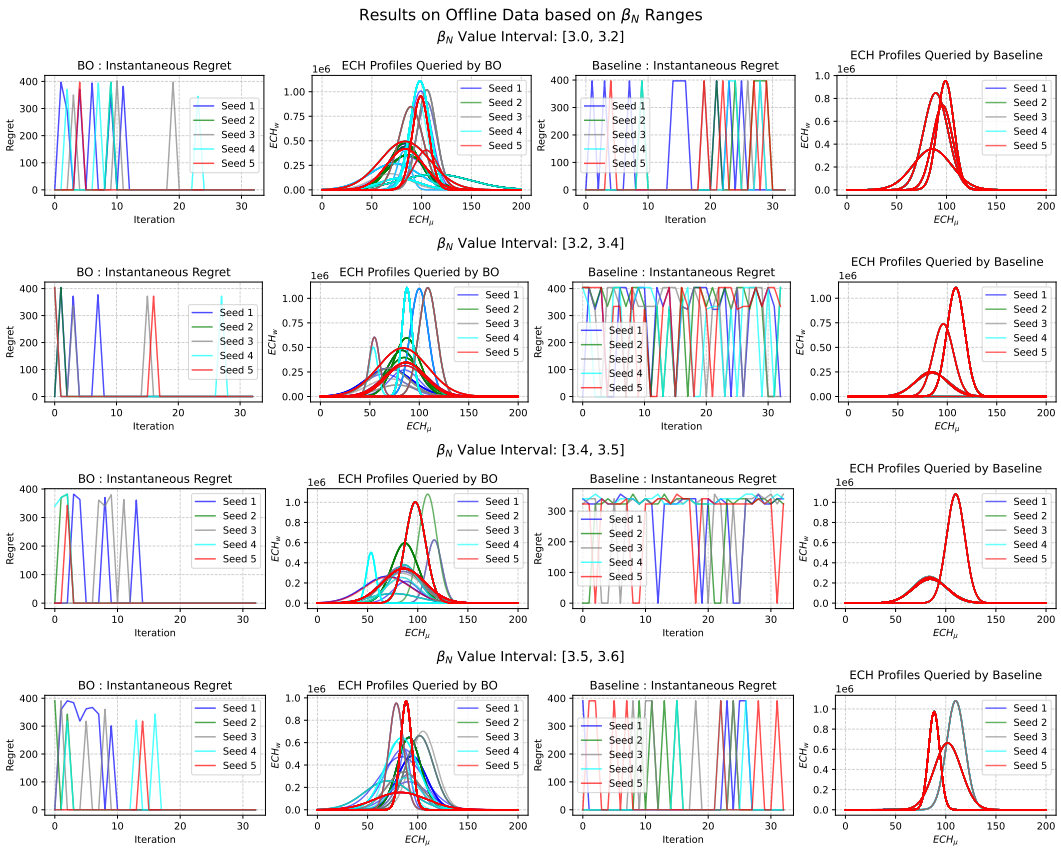

Figure 2: Comparison of results on offline data with Bayesian Optimization vs Prior model (baseline) optimization. The BO algorithm is on the path to convergence however, the prior model baseline does not have a model update step, hence it cannot adapt to new data. Moreover, the BO algorithm explores more, which further helps in identifying ECH settings which perform well.

In Fig. 2, we depict the regret

$$\text{Regret}_i = \tau - t_{\text{TM}}^{(i)} \tag{10}$$

for four ranges of $\bar{\beta}_N$, where we subsumed multiple bins into a single figure. The results obtained with our approach are depicted on the left-hand side of Fig. 2, and those of the RPNN-based planner are on the right-hand side. In most cases, our algorithm is able to find an ECH value a_q that stabilizes the plasma for any value of $\bar{\beta}_N$. In all seeds, not more than a single instability is encountered after $n = 15$ iterations. This clearly illustrates the data efficiency of our approach despite the complexity of the setting. By contrast, the RPNN is seldom able to find stabilizing ECH profiles. This is partly because the RPNN is not accurate enough to faithfully predict tearing instabilities, but also because the planner fully trusts each prediction and does not get updated after every iteration. This does not encourage exploration, which leads to the model picking a poor ECH profile multiple times, even after it has seen it fail. The lack of exploration is also illustrated in Fig. 2, where our approach exhibits considerably more variety in the queried ECH profiles than the RPNN approach.

4.2 DIII-D Tokamak Experiments

We tested our algorithm at the General Atomics DIII-D Tokamak during a two-hour time window allocated to us during the FY24 campaign. Getting time allocated on the reactor is a competitive process, with applicants from across the globe, thus limiting the number of experiments one can perform on the actual device. Each experiment run involves applying actions to the plasma, known as a 'shot'. Each shot is then assigned a unique shot number.

Experiment ID (Shotnumber)	Target $\bar{\beta}_N$	Projected ECH Profile			GP Variance	Tearing Instability Avoided
		Center	Width	Height		
xxx99	3.37	0.45	0.14	0.85	0.239	**Yes**
xxx01	3.27	0.43	0.11	1.30	0.179	**Yes**
xxx02	3.27	0.44	0.10	1.32	0.170	No
xxx03	3.27	0.39	0.18	0.80	0.380	**Yes**
xxx04	3.27	0.42	0.14	0.99	0.044	**Yes**
xxx05	3.14	0.39	0.20	0.54	0.299	No
xxx06	3.45	0.43	0.18	0.62	0.066	**Yes**
xxx07	3.43	0.42	0.19	0.62	0.047	**Yes**

Table 1: Results from two-hour experiment session at the DIII-D Tokamak with a high performance plasma configuration. Our approach avoids tearing instability in 6/8 runs. In a past experiment with these conditions, tearing instabilities were present in 10/12 runs. The ECH profile selected by our approach is a Gaussian curve parameterized by its center, width, and height. The location of the center and width is given in normalized plasma radius, and the height is in MW/m^3. Tearing instability here refers specifically to 2-1 tearing instability. Full experiment IDs redacted for anonymity.

To make the most of our time and be able to make significant statements about results, we opt for a pre-specified set of feedforward actuators a_t^f that is highly unstable, having a historical rate of tearing mode occurrences of 86%. During the experiment, we perform 8 BO iterations with our algorithm. The selected heating profiles a_q^{BO} are converted to gyrotron angles and entered into the Plasma Control System, the interface that controls the tokamak. After a few seconds of maintaining the plasma, we ramp down actuators and terminate the shot.

We start our experiments by recreating a high performing historical high q_{min} experiment that has a tearing instability. For this, we recreate the conditions in shot 180636, an unstable high q_{min} plasma trajectory conducted previously at DIII-D. During the experiments with this configuration, which correspond to the past shots 180636 to 180647, tearing instabilities were seen in 10/12 shots. We recreate these conditions and test if our method is able to avoid these instabilities. In each of our experiments, we run a similar configuration and vary the parameters slightly, resulting in a different $\bar{\beta}_N$. The details for each of the 8 shots carried out using our approach are shown in Table 1. Our algorithm was able to successfully avoid tearing instabilities in 6 out of 8 shots.

5 Conclusion

In this work, motivated by the challenges of tokamak control, we develop a multi-scale modeling approach for making decisions on the fly using a handful of data. Our pipeline leverages a high-frequency neural network model of the system dynamics and a Gaussian process that makes predictions based on marginal statistics. Together, both models form a Bayesian optimization algorithm tailored to the task at hand and can quickly identify stabilizing control actions. This is achieved by making decisions on the fly based on newly collected data. Our method outperforms a naive planner based exclusively on the dynamic neural network model, mainly due to better exploration capability on the part of our approach. Moreover, our method shows promise in live experiments on the DIII-D Fusion reactor. During the experiments, our approach successfully avoided tearing instability in 6/8 runs despite highly unstable conditions, representing an improvement of over 300% percent compared to past experiments.

Our work illustrates the potential of combining complex high-frequency and low-frequency models to improve performance on the fly based on incoming data. In the field of nuclear fusion, the need for similar methods will increase in the future, as new and larger reactors such as ITER become operational, and a significant gap between existing and new models needs to be bridged with very little data. This is the case not only for the stabilization setting considered in this paper but also for settings such as ramp-up design, where a different set of actuators is considered. Moreover, we believe this approach could be of interest to several other applications where the discrepancy between past and present data is considerable, which is often the case in practice.

REFERENCES

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

Mark Boyer, Josiah Wai, Mitchell Clement, Egemen Kolemen, Ian Char, Youngseog Chung, Willie Neiswanger, and Jeff Schneider. Machine learning for tokamak scenario optimization: combining accelerating physics models and empirical models. In *APS Division of Plasma Physics Meeting Abstracts*, volume 2021, pp. PP11–164, 2021.

Ian Char, Youngseog Chung, Willie Neiswanger, Kirthevasan Kandasamy, Andrew Oakleigh Nelson, Mark Boyer, Egemen Kolemen, and Jeff Schneider. Offline contextual bayesian optimization. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (eds.), *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019. URL https://proceedings.neurips.cc/paper_files/paper/2019/file/7876acb66640bad41f1e1371ef30c180-Paper.pdf.

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*, pp. 1357–1372. PMLR, 2023a.

Ian Char, Youngseog Chung, Joseph Abbate, Egemen Kolemen, and Jeff Schneider. Full shot predictions for the diii-d tokamak via deep recurrent networks. In *APS Division of Plasma Physics Meeting Abstracts*, volume 2023, pp. UP11–096, 2023b.

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.

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.

Marc Deisenroth and Carl E Rasmussen. Pilco: A model-based and data-efficient approach to policy search. In *Proceedings of the 28th International Conference on machine learning (ICML-11)*, pp. 465–472, 2011.

Yichen Fu, David Eldon, Keith Erickson, Kornee Kleijwegt, Leonard Lupin-Jimenez, Mark D Boyer, Nick Eidietis, Nathaniel Barbour, Olivier Izacard, and Egemen Kolemen. Machine learning control for disruption and tearing mode avoidance. *Physics of Plasmas*, 27(2), 2020.

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

Gunter Gantenbein, H Zohm, G Giruzzi, S Günter, F Leuterer, M Maraschek, J Meskat, Q Yu, ASDEX Upgrade Team, et al. Complete suppression of neoclassical tearing modes with current drive at the electron-cyclotron-resonance frequency in asdex upgrade tokamak. *Physical Review Letters*, 85(6):1242, 2000.

Roman Garnett. *Bayesian optimization*. Cambridge University Press, 2023.

Tairan He, Chong Zhang, Wenli Xiao, Guanqi He, Changliu Liu, and Guanya Shi. Agile but safe: Learning collision-free high-speed legged locomotion. *arXiv preprint arXiv:2401.17583*, 2024.

Qiming M Hu, Nikolas C Logan, Qingquan Yu, and Alessandro Bortolon. Effects of edge-localized electron cyclotron current drive on edge-localized mode suppression by resonant magnetic perturbations in diii-d. *Nuclear Fusion*, 64(4):046027, 2024.

Lukasz Kaiser, Mohammad Babaeizadeh, Piotr Milos, Blazej Osinski, Roy H Campbell, Konrad Czechowski, Dumitru Erhan, Chelsea Finn, Piotr Kozakowski, Sergey Levine, et al. Model-based reinforcement learning for atari. *arXiv preprint arXiv:1903.00374*, 2019.

Egemen Kolemen, AS Welander, RJ La Haye, NW Eidietis, DA Humphreys, J Lohr, V Noraky, BG Penaflor, R Prater, and F Turco. State-of-the-art neoclassical tearing mode control in diii-d using real-time steerable electron cyclotron current drive launchers. *Nuclear Fusion*, 54(7): 073020, 2014.

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

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

Michael Lehnen, K Aleynikova, PB Aleynikov, DJ Campbell, P Drewelow, NW Eidietis, Yu Gasparyan, RS Granetz, Y Gribov, N Hartmann, et al. Disruptions in iter and strategies for their control and mitigation. *Journal of Nuclear materials*, 463:39–48, 2015.

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.

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, 2024.

O. Meneghini, S.P. Smith, L.L. Lao, O. Izacard, Q. Ren, J.M. Park, J. Candy, Z. Wang, C.J. Luna, V.A. Izzo, B.A. Grierson, P.B. Snyder, C. Holland, J. Penna, G. Lu, P. Raum, A. McCubbin, D.M. Orlov, E.A. Belli, N.M. Ferraro, R. Prater, T.H. Osborne, A.D. Turnbull, and G.M. Staebler. Integrated modeling applications for tokamak experiments with omfit. *Nuclear Fusion*, 55(8):083008, 2015. URL http://iopscience.iop.org/article/10.1088/0029-5515/55/8/083008/meta.

Mustakimah Mohamed, Nur Diyana Zakuan, Tengku Nur Adibah Tengku Hassan, Serene Sow Mun Lock, and Azmi Mohd Shariff. Global development and readiness of nuclear fusion technology as the alternative source for clean energy supply. *Sustainability*, 16(10):4089, 2024.

Masayuki Ono, John W Berkery, Nicola Bertelli, Syunichi Shiraiwa, Luis Delgado-Aparicio, Jonathan E Menard, Álvaro Sánchez-Villar, Kajal Shah, Vladimir F Shevchenko, Hiroshi Idei, et al. Efficient eccd non-inductive plasma current start-up, ramp-up, and sustainment for an st fusion reactor. *Nuclear Fusion*, 2024.

FC Schuller. Disruptions in tokamaks. *Plasma Physics and Controlled Fusion*, 37(11A):A135, 1995.

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.

RJ Shalloo, SJD Dann, J-N Gruse, CID Underwood, AF Antoine, Christopher Arran, Michael Backhouse, CD Baird, MD Balcazar, Nicholas Bourgeois, et al. Automation and control of laser wakefield accelerators using bayesian optimization. *Nature communications*, 11(1):6355, 2020.

Benjamin J Shields, Jason Stevens, Jun Li, Marvin Parasram, Farhan Damani, Jesus I Martinez Alvarado, Jacob M Janey, Ryan P Adams, and Abigail G Doyle. Bayesian reaction optimization as a tool for chemical synthesis. *Nature*, 590(7844):89–96, 2021.

M Shimada, DJ Campbell, V Mukhovatov, M Fujiwara, N Kirneva, K Lackner, M Nagami, VD Pustovitov, N Uckan, J Wesley, et al. Overview and summary. *Nuclear Fusion*, 47(6):S1, 2007.

T Wakatsuki, M Yoshida, E Narita, T Suzuki, and N Hayashi. Simultaneous control of safety factor profile and normalized beta for jt-60sa using reinforcement learning. *Nuclear Fusion*, 63(7): 076017, 2023.

A APPENDIX

A.1 DATASET

Plasma trajectories on a Tokamak consists of three phases. The ramp-up phase, where the gases are heated and pressure is increased to generate the plasma state where fusion occurs. During this phase, the normalized plasma pressure β_N rises. Then, we enter the flat-top phase, where the plasma pressure β_N is sustained, allowing fusion to occur. In this phase, β_N is mostly constant and the aim to maintain this state without instabilities. Finally, the actuators are gradually ramped down and the plasma is safely terminated as the shot concludes. In this paper, we stay in the flat top phase and aim to stabilize it. To create our dataset, we hence use only flat top data and only control actuators during this phase of the experiment.

Our complete dataset contains of $\sim$ 15000 plasma trajectories from historical experiments at DIII-D Tokamak. The data contains signals from different diagnostics have different dimensions and spatial resolutions, and the availability and target positions of each channel vary depending on the discharge condition. Therefore, the measured signals are preprocessed into structured data of the same dimension and spatial resolution using the profile reconstruction and equilibrium fitting (EFIT). These shots contain many different signals, some of which are described below. The dataset consists of scalar signals defined at every timestep and profile signals which are defined along 33 or 200 points along the radius of the plasma cross-section. These consists of temperature, ion temperature, pressure, rotation, safety (Q) factor and density. For these signals we first convert them into PCA components and select the top components which are able to explain 99% of the variance in data. The Electron Cyclotron Heating (ECH) profile we choose to control, is also defined at 200 points along the plasma radius. PCA is unable to describe ECH profiles, however they can described well by a Gaussian curve and are hence parameterized by the center, width and amplitude of the curve. These 3 parameters form our parameterization a_q of the ECH profiles. The model state space s_t is shown in table 2 while the actuator space a_t is shown in table 3.

State Variables	Additional Processing/Details
Normalized Plasma Pressure β_N	-
Line averaged density	-
Loop voltage	-
Confinement Energy	-
Temperature Profile	Decomposed to 4 PCA components
Ion Temperature Profile	Decomposed to 4 PCA components
Density Profile	Decomponsed to 4 PCA components
Rotation Profile	Decomposed to 4 PCA Components
Pressure Profile	Decomposed to 2 PCA components
Q Profile (safety factor)	Decomposed to 2 PCA components

Table 2: Plasma Features used as state space for RPNN model.

Actuator Variables	Additional Processing/Details
Power Injected	-
Torque Injected	-
Target Current	-
Target Density	-
Magnetic Field	-
Total Deutrium	-
Gas Controls	6 Scalars
ECH	Decomposed to Gaussian curve with mean, stddev, amplitude (μ, σ, w)

Table 3: Plasma Features used as actuator space of the RPNN model.

For training the RPNN, we utilize this data set with data points every 20 ms in time intervals with trajectories having an average length of 5 seconds. The RPNN is trained to predict Δs_{t+1} given (s_t, a_t). We add tearing mode labels to this dataset and train a random forest classifier to predict the probability of tearing modes at every time step. We tried incorporating tearing

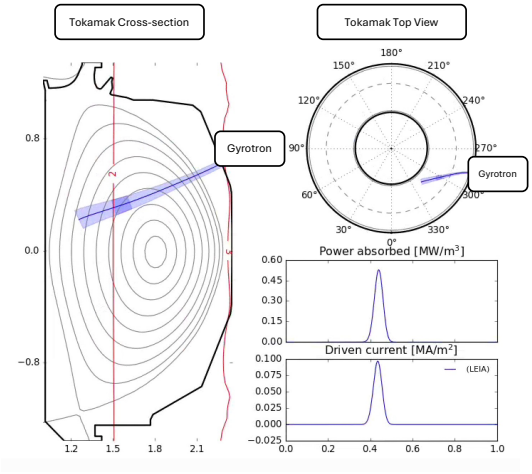

Figure 3: Gyrotron action on the Plasma inside the Tokamak. The bottom 2 curves indicate the power absorbed (heating profile) and current driven in the plasma from the centre to outer region of the plasma.

mode predictions inside the RPNN network however, we did not get good results. This is likely due to the formation of spurious correlations and causality issues formed by introducing tearing modes into the dataset.

To create the dataset for offline testing $\mathcal{D}^H$, we first limit ourselves to High q_{min} trajectories having high β_N values in the range [3.0, 3.6] and the center of the parameterized gaussian curve is between $(0.2, 0.75)$ in normalized radius. A figure of how the heating profile looks is shown in Fig. 3. These constraints follow our experiment conditions. This leaves us with 125 trajectories. We subsequently convert this data from a time-step scale to a trajectory level scale. We take average β_N of the flat-top phase of the trajectory. For ECH profile a_q, we take a mean of all profiles in the flat-top phase of the experiment. This is the phase where the high-energy plasma state is maintained. We thus get the dataset D^H where D_i^H consists of triplet $({\beta_N}^i, a_q^i, t_{\mathsf{TM}}^i)$ i.e. the normalized plasma pressure, parameterized ECH profile and the observed time-to-tearing mode. This dataset is used for offline testing. For online testing, we use this as training set for the Gaussian Process.

A.2 Approximating the Prior

The historical data used to train the RPNN and the GP does not contain the target normalized plasma pressure $\bar{\beta}_N$. Instead, it only contains the actions a_t achieved during the shot. Similarly, the RPNN is trained exclusively on the actions, and not on $\bar{\beta}_N$, hence a direct mapping from $\bar{\beta}_N$ does not take place in the RPNN. In the experiments, we address these issues as follows. In the historical data, we set $\bar{\beta}_N$ to be equal to the average normalized plasma pressure, i.e.,

$$\bar{\beta}_N^{(i)} \approx \sum_{t=1}^{\tau} {\beta_{N,t}}^{(i)}. \tag{11}$$

This is a reasonable assumption since the target $\bar{\beta}_N$ is mostly achieved in practice. We then approximate the time-to-tearing mode $\hat{t}_{\text{TM}}(\bar{\beta}_N, a_q)$ predicted by the RPNN given $\bar{\beta}_N$ and a_q as follows. We first use a_q to compute the actions a_t^{ech}. We then compute the remaining actions a_t^c and a_t^f by sampling full rollouts from the historical data and setting a_t^c and a_t^f equal to the corresponding actions. We then look at the resulting average normalized plasma pressure and set it equal to $\bar{\beta}_N^{(i)}$. We do this for all ECH actions a_q within a $10 \times 10 \times 10$ grid within the space of ECH parameters, specified by the historically largest and smallest parameter values in the historical data set. We then separate the results into bins that have the same value of $\bar{\beta}_N^{(i)}$ up to a margin of $\epsilon = 0.04$, and average over all tearing modes within that bin, yielding $\hat{t}_{\text{TM}}(\bar{\beta}_N, a_q)$. At test time, we project all points to the closest point on the grid, both when performing queries and before updating the GP model.

A.3 Conversion of ECH Profile to Gyrotron Angles

Even though we selected ECH profiles as our action space, the Plasma Control System (PCS) at DIID tokamak expects the output to be Gyrotron angles, which denote locations where they will be aimed. To make this conversion, we used OMFIT software (Meneghini et al., 2015). We selected ECH profiles as our action space instead of gyrotron angles because at experiment time one does not know how many gyrotrons are available. With this choice of action space, we ensure our method is agnostic of number of gyrotrons.