Extended Abstract

Motivation Particle accelerators enable scientific discoveries at the frontiers of high-energy physics, materials science, and medicine. (Wilson, 2001) To make the most of limited and expensive beam time, accelerator control algorithms must be robust and converge rapidly. Significant progress has been made in automated beam tuning algorithms, primarily using Bayesian optimization and model-free reinforcement learning (RL). (Edelen and Huang, 2024) However, Bayesian optimization with static Gaussian processes is unable to account for model drifts, and online training with model-free RL requires costly samples during inference. This project uses model-based reinforcement learning (MBRL) with learned surrogate models to enable accelerator control in complex and drifting systems with nonlinear dynamics.

Method The work focuses on the problem of beam control within the photoinjector of the FACET-II facility at SLAC. Two different surrogate methods were implemented and evaluated: a multilayer perceptron (MLP) which maps from the photoinjector state parameters to the transverse emittance measurement (the metric of interest in the experiment) and a normalizing flow which learns the distribution of the electron bunch in phase space exiting the photoinjector. Ground truth data for training the surrogates was generated from IMPACT-T, a particle-in-cell beam physics code. (Qiang et al., 2006) Two differentiable MBRL algorithms were implemented and evaluated on these surrogates: backpropagation through time (BPTT), which evaluates policy gradients via differentiating through the full model rollout, and short horizon actor-critic (SHAC), which truncates the model rollout and adds a learned terminal value estimate. (Xu et al., 2022) Proximal policy optimization (PPO) was included as a baseline for model-free RL. (Schulman et al., 2017)

Experiments Both surrogate models were trained on data generated from IMPACT-T over a Latin hypercube sample of photoinjector parameters. The MBRL algorithms were trained via backpropagation of the objective through the surrogate model rollout in each episode. The first objective tested was minimization of the transverse emittance ε_{4D}, which was run with both the MLP and normalizing flow surrogate models. Additionally, to evaluate the capabilities of the normalizing flow model, beam shape control was added as an objective. Beam shape setpoints were ramped and stepped dynamically to add an additional challenge for the RL algorithms, which were trained in curriculum learning on the new set point-conditioned version of the problem.

Results Both surrogate models trained well on their respective objectives, with the MLP attaining a mean absolute percent error (MAPE) of 2.7% on ε_{4D} and the normalizing flow attaining a MAPE of 5.8% (even while simulating each electron macroparticle in the bunch independently). All three algorithms (BPTT, SHAC, PPO) performed comparably well on the emittance minimization task, attaining median final emittances $< 3 \times 10^{-11}$ m^2 on evaluation against IMPACT-T after training on the surrogate. However, BPTT trained over $5\times$ faster than PPO, assisted by gradient evaluation times and no critic training. Results on the flow-based surrogate were noisier, and SHAC in particular suffered from performance instabilities. In the beam shape tracking test, SHAC and BPTT outperformed PPO with $2-3\times$ less tilt error. SHAC performed better on smooth ramp trajectories, while BPTT performed better on stepped set point changes. All policies transferred well, even to higher-fidelity IMPACT-T evaluations than used in surrogate training.

Discussion The results demonstrate the ability of the surrogate architectures to capture the relevant physics and train policies which transferred without degradation to the IMPACT-T numerical model. Differentiable MBRL methods scaled better than PPO in terms of compute time and performance, particularly on the more challenging shape tracking task. The results motivate additional work on assessing the robustness of these conclusions to higher-resolution simulation and more realistic diagnostic information. Future steps include interfacing the normalizing flow simulation with differentiable beam dynamics codes for start-to-end simulation and incorporating more challenging control objectives such as two-bunch operation.

Conclusion MBRL with learned surrogates is shown to be a promising avenue for developing particle accelerator controllers for time-varying and nonlinear beam dynamics.

Model-Based Reinforcement Learning for Particle Accelerator Control

Ryan Wu
Department of Mechanical Engineering
Stanford University
rwu4@stanford.edu

Abstract

Particle accelerator control is a high-dimensional and nonlinear problem, with many tuning procedures requiring intensive expert operator intervention. Previous work has demonstrated the applicability of model-free Bayesian optimization and reinforcement learning (RL) strategies for accelerator tuning. This project uses surrogate modeling techniques to extend that previous work to RL for photoinjector control for FACET-II at SLAC. Model-based RL methods are implemented and tested using different surrogate model formulations, demonstrating improved training time (up to 5 times faster) and performance compared to baseline model-free RL methods on emittance reduction and beam shape tuning. When used in conjunction with differentiable accelerator simulation models, the techniques developed in this work may be extensible to model-based RL in start-to-end control of accelerators with complex setpoints and drifting beamline parameters.

1 Introduction

Particle accelerators are powerful tools for science, allowing users to probe high-energy physics, examine molecular structure, and develop new medical interventions. (Wilson, 2001) Advancements in particle accelerator performance have pushed forward the boundaries of theoretical physics and materials science, and miniaturized accelerators have been commercially deployed in applications such as cancer treatment, space environment engineering, and industrial inspection. Modern particle accelerators work in a high-dimensional operating space with hundreds of control parameters and thousands of observables. (Edelen and Huang, 2024) Beam time on particle accelerator experiments is heavily oversubscribed and operating costs are on the order of a million dollars per day, motivating rapid and reliable tuning and control to ensure high-quality beam performance to support user science. The combination of these two parameters motivates study into robust control techniques to navigate the complex and dynamic objective space of particle accelerator tuning.

This project focuses specifically on the Facility for Advanced Accelerator Experimental Tests-II (FACET-II) at SLAC National Accelerator Lab. FACET-II is an electron linear accelerator experiment, with the objective of maturing technologies for future accelerator concepts such as plasma wakefield acceleration. (Yakimenko et al., 2019) Another objective of FACET-II is maturing the machine learning techniques necessary to support complex accelerator operation, which aligns closely with the objective of this research. In FACET-II, electrons are injected from a laser striking a photocathode within a photoinjector assembly, which are then accelerated through a series of radiofrequency (RF) cavities and focused with magnetic optics towards the experimental interaction point. These beamline components provide hundreds of degrees of freedom for operators to customize the beam to the experimenter's needs. (Edelen et al., 2018) However, particularly for challenging beam scenarios such as two-bunch operation, tuning these parameters consumes hours of valuable beamtime, with setpoints drifting dynamically due to changes in accelerator geometry with temperature and hysteresis. (Storey et al., 2024)

Stanford CS224R 2026 Final Report

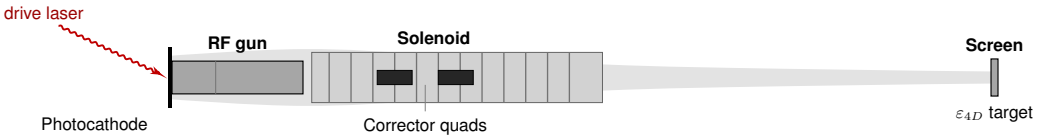

Figure 1: Schematic of FACET-II photoinjector.

This project will focus on tuning of the photoinjector, since nonlinear space charge effects play a major role in electron dynamics in this segment and necessitate the creation of new modeling approaches to efficiently capture that system behavior. A model of the photoinjector is shown in figure 1.

These challenges in high-dimensional control within the data-rich accelerator environment motivate reinforcement learning approaches. A typical accelerator experiment at SLAC (LCLS-II) generates data at a rate of 1 TB/s, providing a significant amount of information to infer system state and train dynamics models, and far more than can be manually interpreted by a human operator. (Mishra et al., 2025) Additionally, physics codes at a hierarchy of fidelities provide a set of environments for offline training, including with gradient information in certain cases. (Sagan, 2006; Qiang et al., 2006; Kaiser et al., 2024) Since evaluations on real accelerator controls are extremely limited, these codes can help bridge the sim2real gap and accelerate agent learning in-the-loop.

In light of these needs and opportunities in accelerator tuning, this project proposes using model-based reinforcement learning (MBRL) to design robust controllers for FACET-II accelerator components. Section 2 reviews prior work in autonomous accelerator tuning and MBRL, and section 3 proposes an implementation with differentiable MBRL over a photoinjector surrogate. Sections 4 and 5 overview the performance of the proposed approaches on this tuning task, and sections 6 and 7 discuss the implications and next steps from this project.

2 Related Work

Previous work from the SLAC Accelerator ML group has demonstrated the capability of machine learning models to provide insight into beam behavior. (Edelen and Huang, 2024) These models have also been successfully combined with Bayesian optimization (BO) to reduce tuning time and improve performance during experimental operation. (Edelen and Huang, 2024; Boltz et al., 2025) While the full electron bunch dynamics exist in six-dimensional position–momentum phase space (x, y, z, p_x, p_y, p_z), many ML efforts in accelerators to date have focused on summary beam parameters (eg. centroids and emittances) or lower-dimensional phase-space reconstructions. For example, a recent latent-space ML model forecasts the 15 unique 2D projections of a 6D phase-space distribution. (Rautela et al., 2024) Full control of the beam for applications such as plasma wakefield acceleration requires manipulation in this 6D phase space, beyond what 2D slices can offer. (Roussel et al., 2024)

Additionally, Bayesian optimization in accelerator tuning has faced challenges in distribution drift, as the accelerator shifts in alignment due to temperature and continued operation. (Mishra et al., 2026) Reinforcement learning can more robustly manage this uncertainty to enable more responsive tuning as accelerator conditions shift. Previous work in RL has included model-free implementations at CERN (Kain et al., 2020), which demonstrated good performance but required hundreds of online rollout episodes if tested without offline pretraining. MBRL has been demonstrated with differentiable linear beamline models (Kaiser et al., 2024), as discussed subsequently. This work seeks to build upon and eventually interface with that work, by focusing on the nonlinear photoinjector regime where differentiable models are not yet available.

This proposal is enabled by recent improvements in modeling and diagnostics to rapidly provide online feedback to the human controller and RL algorithm. Differentiable and GPU-native implementations of beam physics codes allow rapid simulations of accelerator configurations with gradient information available. (Kaiser et al., 2024; Gonzalez-Aguilera et al., 2023; Signorelli et al., 2026) Automated diagnostics also provide a better picture of the full 6D phase space during online operation, providing high-resolution feedback to the controller. (Roussel et al., 2026) These developments, along with a normalizing flow surrogate for photoinjector dynamics developed in a past project, enable rapid training of candidate RL models and clear visibility into controller performance once deployed.

3 Method

3.1 Markov Decision Process Definition

The control Markov decision process (MDP) of interest is defined as follows. The state space $\mathcal{S}$ is defined by 6 particle distribution components for the initial photocathode output (laser spot size, truncation radius, mean transverse energy, pulse duration, and two transverse asymmetry components) and 5 control knobs (solenoid field, quadrupole gradient, skew quadrupole gradient, RF gun field scale, and RF gun phase). Refer to figure 1 for the photoinjector geometry. The action space $\mathcal{A}$ is Δ change commands on the 5 control knobs with box bounds. The actor observes the current state of the 5 control knobs as well as the objective, which is ε_{4D} (defined below) for the first set of experiments. For the emittance objective, the reward is defined as the time-integrated z-scored $-\log(\varepsilon_{4D})$. The initial photocathode output distribution is not observable by the actor, making the problem partially observable.

3.2 Surrogate Model Fitting

Electron bunch dynamics within the low-energy photoinjector regime are heavily influenced by space charge effects, making linear optics propagation assumptions inaccurate. (Qiang et al., 2006) As such, surrogate models are necessary to capture the relevant dynamics for training. In this project, two surrogate models were attempted, a simple multilayer perceptron (MLP) model for dedicated single-objective targeting, as well as a flow-matching model that models the full phase space evolution of the electron bunch.

3.2.1 MLP

The first optimization objective is to minimize the transverse emittance of the beam. The normalized transverse emittance ε_{4D} is defined as:

$$\varepsilon_{4D} = \frac{1}{(mc)^2}\sqrt{\det(\Sigma_{4\times4})}, \tag{1}$$

where the transverse beam covariance matrix $\Sigma_{4\times4}$ is defined as:

$$\Sigma_{4\times4} = \begin{bmatrix} \langle x^2\rangle & \langle xp_x\rangle & \langle xy\rangle & \langle xp_y\rangle \\ \langle p_x x\rangle & \langle p_x^2\rangle & \langle p_x y\rangle & \langle p_x p_y\rangle \\ \langle yx\rangle & \langle yp_x\rangle & \langle y^2\rangle & \langle yp_y\rangle \\ \langle p_y x\rangle & \langle p_y p_x\rangle & \langle p_y y\rangle & \langle p_y^2\rangle \end{bmatrix}, \tag{2}$$

where a variable such as $\langle xp_y\rangle$ defines the second-order central moment of the product of the x position and vertical momentum p_y of the particles in the beam. (Wiedemann, 2015) The standard convention is observed where z is defined as the longitudinal direction along the propagation of the beam. The transverse emittance is analogous to a volume occupied by the beam in position-momentum phase space. Controlling emittance growth is an important requirement for generating high-quality beams for applications such as accelerator-driven light sources and colliders. (Ayoub Miskovich et al., 2024)

The surrogate directly maps between the 11 state space parameters to z-scored $\log(\varepsilon_{4D})$, using a dense MLP architecture trained with mean squared error regression. The model is adequate for predicting any single given variable of interest and exposes gradient information for first-order MBRL strategies to exploit. However, the surrogate does not capture the full 6D phase space dynamics of the particle bunch and cannot be reused for new objectives of interest. This gap motivates a dynamics model which can describe the full phase-space evolution of the electron bunch.

3.2.2 Normalizing Flow Model

The bunch of electrons within the accelerator can be defined as a distribution in 6D space. This enables the use of normalizing flow models to describe the evolution of this distribution through the photoinjector, enabling arbitrary sampling of posterior distributions over the bunch. Normalizing

flows learn an invertible map between the output distribution and a latent distribution (Gaussian in this case). This project uses the real-valued non-volume preserving flow (Dinh et al., 2016), which partitions the state vector into two components, where one component is used as input into a neural network to generate an affine scale and shift function for the other half. The process is repeated with alternating masks to transform all dimensions of the distribution.

Specifically, the state vector for a given electron is defined as the 6D vector (x, y, z, p_x, p_y, p_z). Since the particle distribution is conditioned upon an 11D state vector, this state is encoded with a separate MLP and appended to the input to the scale and shift functions. The model is trained on a combination of objective functions, including negative log-likelihood loss for the particle positions (using the inverse map) and emittance (ε_{4D}) accuracy. This architecture admits gradients from arbitrary properties computed over the output particle bunch using the reparameterization trick. This is used in training to include emittance as a loss function and in MBRL to take derivatives of bunch properties with respect to control knobs.

3.3 Differentiable MBRL

To take advantage of these differentiable beam surrogate models, two separate first order MBRL schemes were considered, backpropagation through time (BPTT) and short horizon actor-critic (SHAC). Using the MDP defined previously with a finite task horizon H, BPTT computes a discounted reward loss as:

$$\mathcal{L}_\theta = -\frac{1}{NH}\sum_{i=1}^{N}\left[\sum_{t=t_0}^{t_0+H}\gamma^{t-t_0}\mathcal{R}(\mathbf{s}_t^i, \mathbf{a}_t^i)\right] \tag{3}$$

for N rollouts, a discount rate γ, and a reward function $\mathcal{R}$ defined as the beam property of interest. (Mozer, 2013) Gradients with respect to the actor policy parameters θ can be taken as $\frac{\partial \mathcal{L}_\theta}{\partial \theta}$ via backpropagating through the surrogate model. Following gradient computation, one step of Adam is used to update the policy parameters. (Kingma and Ba, 2017)

SHAC was proposed in Xu et al. (2022) to resolve some observed instabilities in BPTT over long task horizons which can make the loss landscape noisy. Rather than backpropagate gradients over the full task horizon H, SHAC introduces a short horizon h, beyond which rewards are approximated via a value function V_ϕ:

$$\mathcal{L}_\theta = -\frac{1}{Nh}\sum_{i=1}^{N}\left[\left(\sum_{t=t_0}^{t_0+h-1}\gamma^{t-t_0}\mathcal{R}(\mathbf{s}_t^i, \mathbf{a}_t^i)\right) + \gamma^h V_\phi(\mathbf{s}_{t_0+h}^i)\right]. \tag{4}$$

V_ϕ is fit via mean squared error regression against an estimated value function $\tilde{V}$ which is fit using a td-λ formulation. To stabilize training, the target value function $V_{\phi'}$ is updated via an exponential moving average (Polyak averaging), controlled by a parameter α: $V_{\phi'} \leftarrow \alpha V_{\phi'} + (1-\alpha)V_\phi$. When computing policy loss $\mathcal{L}_\theta$, the target value function $V_{\phi'}$ is used.

The full training loop for surrogate model and MBRL training is summarized in figure 2. Hyperparameters for BPTT and SHAC are generally kept the same as the implementations described in Xu et al. (2022), with exceptions mentioned explicitly below. Rollout episodes are 64 steps long, with Δ actions capped to 5% of each control knob range. SHAC was implemented with a rollout horizon of 16 steps.

As a baseline comparison against MBRL implementations, proximal policy optimization (PPO) was used. (Schulman et al., 2017) PPO is a model-free reinforcement learning algorithm which uses an on-policy actor-critic scheme. PPO was selected as a baseline due to the improved stability and reduced hyperparameter tuning which could affect performance when used as a comparison. The implementation of PPO provided in StableBaselines3 was used for testing. (Raffin et al., 2021)

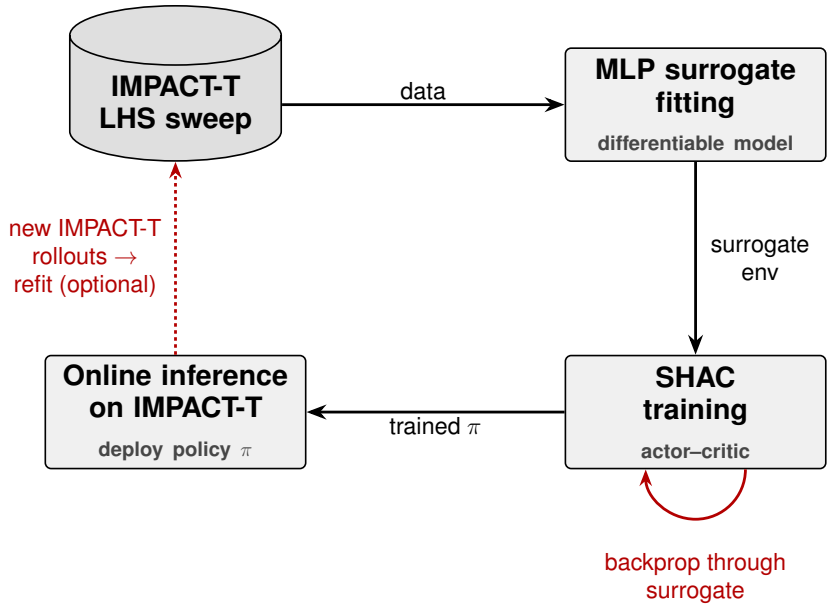

Figure 2: Schematic of training loop for MLP surrogate with SHAC MBRL algorithm.

4 Experimental Setup

4.1 IMPACT-T Data Generation

To generate sample data for training the surrogate models, the IMPACT-T code was used. (Qiang et al., 2006) Beam dynamics is heavily impacted by the resolution of the particle-in-cell simulation, both in the number of macroparticles and mesh cells. To evaluate model transferability and convergence with simulation fidelity, two simulation fidelities were used: 2000 particles in an $8 \times 8 \times 8$ mesh and 20000 particles in a $32 \times 32 \times 32$ mesh. In this project, the former configuration will be referred to as the low resolution and the latter as the medium resolution. Future work would focus on extending this to higher resolution to capture microbunching instabilities and detailed phase-space dynamics.

Box constraints were set for the 11 state space parameters, and Latin hypercube sampling (LHS) over those parameters was performed using the Xopt orchestrator wrapping IMPACT-T. (Roussel et al., 2023) 10000 samples were drawn from the sampler and evaluated at each resolution. The phase space distribution of the particles was reconstructed around the location of the measurement screen PR10241 in FACET-II, where the surrogate loss and control objectives were defined.

4.2 Control Objectives

The initial control problem, as described previously, is minimizing transverse emittance ε_{4D} at the PR10241 measurement screen. When running in IMPACT-T, the statistic was computed over the full particle bunch at the measurement screen. For the MLP surrogate, ε_{4D} was directly predicted as a function of the photoinjector state variables. For the flow surrogate, since an arbitrary number of posterior samples could be drawn from the output distribution, different particle counts were tested for computing ε_{4D} and other metrics of interest. 512 particles were found to be sufficient for model convergence with the flow surrogate and is the sample count used in the remainder of the project.

To demonstrate the versatility of the flow surrogate, an additional set of objectives were implemented and tested. These objectives included normalized 2D transverse emittances ε_x and ε_y, beam spot sizes $\sigma_{x,y,z}$ in each axis, particle energy, and transverse projected beam shape. The projected beam shape was used for demonstration purposes for algorithm training. For the shape objective, the transverse beam profile is modeled as an ellipse, with spatial second moments σ_x, σ_y, and σ_{xy}. A normalized linear Stokes parameter shape vector $\mathbf{s} = [s_1, s_2]$ is defined as $s_1 = \frac{\sigma_x^2 - \sigma_y^2}{\sigma_x^2 + \sigma_y^2}$ and $s_2 = \frac{2\sigma_{xy}}{\sigma_x^2 + \sigma_y^2}$. The first term captures a normalized elongation while the second captures shear. This formulation allows

Model	Training time (s)	Surrogate ε_{4D}	IMPACT-T ε_{4D}
PPO	218 ± 10	2.230	2.45 ± 1.51
SHAC	164 ± 2	2.208	2.46 ± 1.47
BPTT	$\mathbf{37 \pm 1}$	2.207	2.57 ± 1.46

Table 1: Training cost vs. median terminal 4D emittance (units of 10^{-11} m^2). All training performed on one A100 GPU. Surrogate parameters are measured over 3 seeds and 256 rollouts. IMPACT-T parameters are computed over 20 random initial particle distributions.

the definition of a unique beam eigen-aspect ratio and angle objective without the singularity or angle-wrapping challenges induced by directly computing aspect ratio and angle.

Note that this transverse shape control objective is purely for demonstration purposes and does not serve an accelerator physics objective, as the momentum-space vectors are not controlled in the optimization. Provided additional quadrupoles to control, it may be possible to extend this optimization to a round-to-flat beam transform, such as in Cropp V et al. (2019). However, this objective is not a round-to-flat transform.

As an additional challenge, the beam shape parameter was encoded as a moving target, transforming the problem into a goal-conditioned problem. The target $\mathbf{s}$ vector is appended to the state vector observed by the actor, and policy loss is computed against the moving shape objective.

5 Results

5.1 Surrogate Performance

The MLP emittance surrogate performed well on both the low and medium resolution datasets, attaining an R^2 of 0.995 and mean absolute percent error (MAPE) of 2.7%.

The normalizing flow surrogate was only trained on the low resolution dataset due to computational cost. The training landscape was noisier due to the discrete particle-level negative log likelihood objective, but the surrogate was still able to attain a MAPE of 5.8%. More importantly, the surrogate was able to qualitatively and quantitatively replicate the 6D phase space distribution of the bunch, shown in figure 3. This provides confidence in accurate downstream predictions of bunch statistics for tasks like beam shape control.

5.2 Algorithm Comparison: Emittance

All algorithms were able to attain comparable results in terms of emittance minimization, albeit with significantly different training times, attributable to the information provided by model gradients to the MBRL methods. Results over three seeds are summarized in table 1. BPTT trained 5 times faster than PPO and 4 times faster than SHAC, which is attributable to the lack of a critic function in the training loop. The trained actors also demonstrate strong zero-shot transfer from the surrogate model to IMPACT-T, where performance degradation was well under the statistical variability due to changes in the initial particle distribution parameters.

Training curves for RL training against the MLP emittance surrogate are shown in figure 4. The dynamics for the emittance objective appear smooth and differentiable, since BPTT converges rapidly and the regularizing effect of the short horizon in SHAC does not appear to improve performance. PPO takes more environment steps to converge, consistent with the lack of gradient information available.

This analysis was repeated with emittance computed over the flow surrogate. Since the emittance is a derived property of the sampled particle bunch and emittance MAPE is higher in the flow surrogate compared to the MLP, the training process using the flow surrogate was also slightly more unstable. This manifested specifically in degraded performance from SHAC, which was resolved after the Polyak averaging parameter α was increased from 0.2 to 0.95 to reduce variability in the critic model. After that change, the training convergence curves are shown in figure 5. Variability in episode reward is greater and SHAC performance still degrades after a certain number of steps. Additional parameter tuning is likely necessary to improve SHAC performance and stability in this environment.

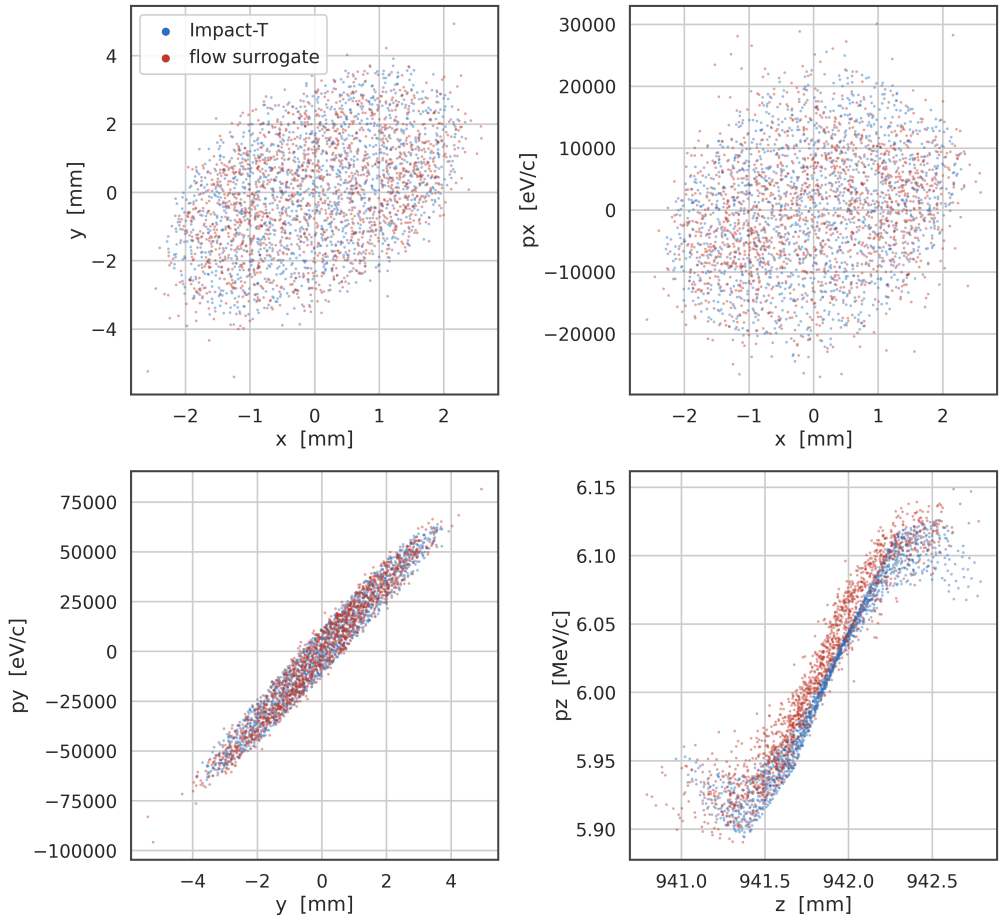

Figure 3: Phase-space slices of beam distribution compared between IMPACT-T and the normalizing flow surrogate. Slices are (clockwise from top left): $x - y$, $x - p_x$, $z - p_z$, $y - p_y$.

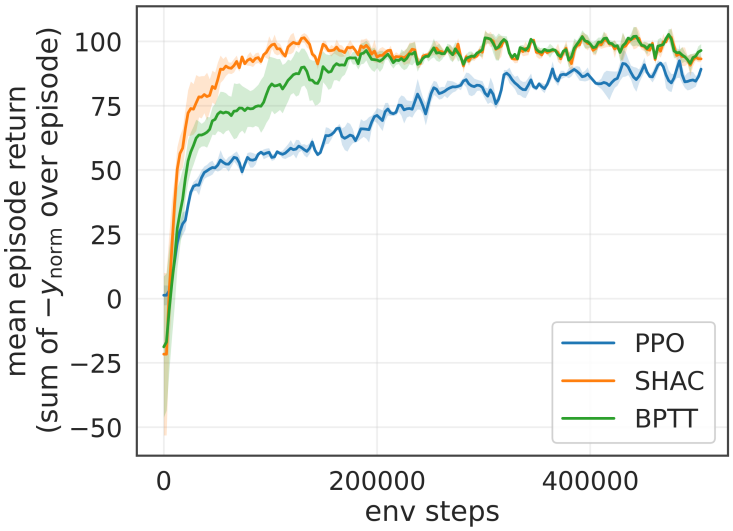

Figure 4: Convergence plots of PPO, SHAC, and BPTT on the MLP surrogate emittance objective.

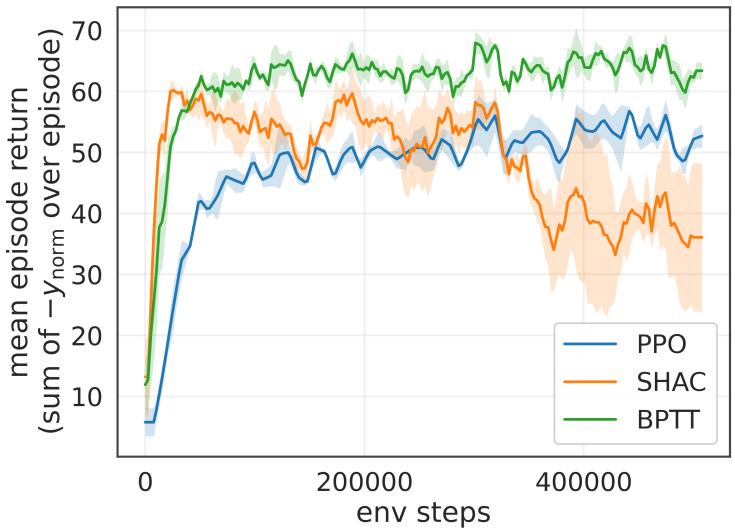

Figure 5: Convergence plots of PPO, SHAC, and BPTT on the flow surrogate emittance objective.

Model	Tilt rotation	Aspect ramp	Staircase
PPO	3.23 / 3.04	2.36 / 2.22	21.59 / 21.42
SHAC	**0.80** / **0.81**	**0.60** / **0.48**	13.03 / 12.69
BPTT	0.92 / 1.11	0.91 / 1.04	**9.16** / **8.48**

Table 2: Settled mean absolute tilt error (degrees) for the goal-conditioned moving-target shape controller on three held-out setpoint schedules. Each cell is **low-fidelity / medium-fidelity**. The settled window excludes the first 8 initialization-recovery steps. Bold = best IMPACT-T per schedule.

In IMPACT-T testing, the models trained on the normalizing flow-based emittance surrogate returned similar results to those shown in table 1.

5.3 Algorithm Comparison: Beam Shape Control

Since the emittance objective appears to perform universally well between the RL model architectures, a beam shape control objective was included to evaluate goal-conditioned control against moving targets. The goal-conditioned versions of the models were trained in a curriculum learning manner, first targeting static shape set points, then piecewise step changes, then continuous ramps. The same amount of environment steps and the same curriculum was used between all models. Results were evaluated on IMPACT-T at both the low and medium resolutions. This scheme also quantifies whether the normalizing flow surrogate (trained at low resolution) could adequately transfer to a higher fidelity simulation.

Three evaluation trajectories were used: rotating the ellipse at constant aspect ratio (tilt rotation), changing the aspect ratio at constant tilt (aspect ramp), and a piecewise series of step changes between aspect/tilt setpoints (staircase). The results are shown in table 2. The first-order MBRL implementations performed better than PPO on all three cases, showing that the gradient and model information had a significant effect on improving model performance. SHAC performed better than BPTT on the continuous ramp trajectories but worse on discontinuous trajectories, possibly attributable to the value function incentivizing the actor to learn to generate smoother trajectories which do not adapt as rapidly to setpoint changes.

The evaluated trajectories are shown in figure 6. SHAC and BPTT are noticeably better at tracking the commanded trajectories compared to PPO, which has high response time and error. Transient stabilization periods exist for all three algorithms in the first few steps. SHAC has slightly slower response time on the staircase steps compared to BPTT, consistent with the previous theory that the value function formulation may encourage transition smoothing with the selected hyperparameters.

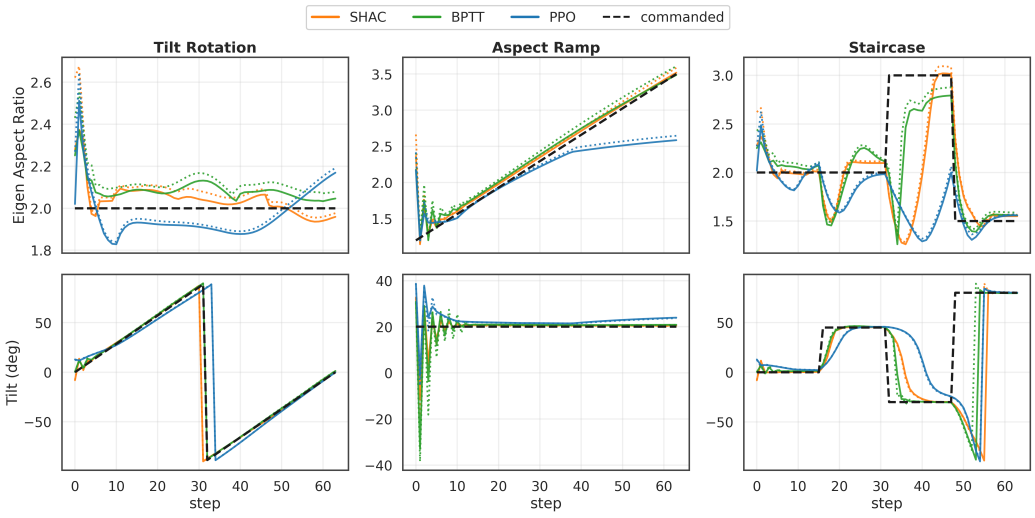

Figure 6: Commanded and performed trajectories of controllers trained on the normalizing flow surrogate, evaluated on IMPACT-T. Low-fidelity IMPACT-T runs are depicted as the solid lines, medium-fidelity runs are depicted as dotted lines.

6 Discussion

The results demonstrate the advantage of first-order MBRL methods used in conjunction with learned surrogate models for training particle accelerator beamline controllers in nonlinear regimes. Transverse emittance tuning was learned well by both the MBRL methods and the PPO model-free baseline. Both the simplified MLP surrogate and the flow surrogate demonstrated strong transferability in a sim2real scheme where the IMPACT-T numerical code was treated as the oracle in evaluation. However, the two surrogate architectures have significantly different performance and scaling characteristics. The MLP surrogate was significantly more lightweight than the normalizing flow, running about 40x faster on an A100 GPU (150 μs vs 6 ms), making it suitable for high-frequency inference for simple but constantly adjusted objectives. On the other hand, the flow surrogate predicts the phase-space position of every individual macroparticle, which is crucial for interfacing with other codes downstream, such as the differentiable beam simulation code Cheetah. (Stein et al., 2022) Additionally, the beam shape optimization demonstrates how multiple objectives can be used over the normalizing flow surrogate without the need for retraining.

On more challenging goal-conditioned objectives, the MBRL algorithms were able to demonstrate improved performance on equivalent episode counts and less walltime compared to PPO. Between BPTT and SHAC, it appears that BPTT was surprisingly more stable in training over the noisy flow surrogate, potentially due to challenges in stabilizing value function fitting in SHAC. Longer-horizon tasks may be necessary to differentiate between SHAC and BPTT, since the SHAC rollout horizon is a significant portion of the episode length. SHAC performed better in continuous setpoint tracking and BPTT performed better against sudden setpoint changes, again likely due to smoothing effects provided by the value function in training. Additional work is required to validate the reliability of these results, as only one seed was run for the shape control experiments due to time limitations. The scaling of trained controllers to higher resolution IMPACT-T runs is promising, though testing on other objectives is needed to verify robustness. Ellipsoid shape control is expected to be mostly driven by linear dynamics, so it may not be a strong bellwether of scaling performance. Testing at greater macroparticle counts will be necessary to ensure transferability to physical beamline conditions.

Beyond addressing the raised limitations, next steps for the project include deployment to more realistic beamline conditions. This includes lattice uncertainty in the photoinjector structure, realistic diagnostic outputs rather than ideal phase space measurements, and incorporation of more beamline elements within the flow model (through the end of the first linac segment L0A). The model gradients from MBRL become particularly powerful with large control dimensionality. This scaling motivates incorporating more of the beamline into the dynamics model, potentially via interfacing with Cheetah. An integrated beamline simulation would also enable in-the-loop system identification for model

error correction during operation. The optimization paradigm is also extensible to more challenging control problems such as two-bunch operation and attosecond pulses. (Yakimenko et al., 2019) These control problems are particularly challenging to solve with manual tuning, so MBRL may generate outsized performance gains.

7 Conclusion

Model-based reinforcement learning has the potential to accelerate RL controller convergence in high-dimensional environments governed by complex but known dynamics. In this project, surrogate models were created to capture electron bunch dynamics in the FACET-II photoinjector. MBRL methods were tested on these surrogates on emittance minimization and beam shape tracking tasks and achieved improved performance and training time compared to a PPO baseline. The success of these methods points towards new opportunities in leveraging physical simulations, surrogates, and world models to improve learned controller design.

8 Team Contributions

- **Ryan Wu:** All components, including literature review, experiment design, implementation, tests. The normalizing flow surrogate was developed as part of a prior research project by Ryan.

AI Use Disclosure Test harnesses, plotting utilities, and training orchestrators are built with the help of Claude Code. BPTT and SHAC were implemented by hand with reference to code from Xu et al. (2022). PPO used the StableBaselines3 implementation.

Changes from Proposal The majority of the stated goals from the proposal were completed, and additional ones were implemented (dynamic shape control instead of static centroid targets). The baseline algorithm was defined as PPO instead of TuRBO to more critically evaluate the contribution of the surrogate model. The surrogate retraining loop was dropped from the project as sim2real performance was already satisfactory on the initial trained surrogate.

References

Sara Ayoub Miskovich, Willie Neiswanger, William Colocho, Claudio Emma, Jacqueline Garrahan, Timothy Maxwell, Christopher Mayes, Stefano Ermon, Auralee Edelen, and Daniel Ratner. 2024. Multipoint-BAX: a new approach for efficiently tuning particle accelerator emittance via virtual objectives. *Machine Learning: Science and Technology* 5, 1 (Jan. 2024), 015004. doi:10.1088/2632-2153/ad169f

Tobias Boltz, Jose L. Martinez, Connie Xu, Kathryn R. L. Baker, Zihan Zhu, Jenny Morgan, Ryan Roussel, Daniel Ratner, Brahim Mustapha, and Auralee L. Edelen. 2025. Leveraging prior mean models for faster Bayesian optimization of particle accelerators. *Scientific Reports* 15, 1 (April 2025), 12232. doi:10.1038/s41598-025-95297-z

Frederick (Eric) Cropp V, Nathan Burger, Paul Denham, Auralee Edelen, Claudio Emma, Jorge Giner Navarro, Edmund Liu, Pietro Musumeci, and Leah Phillips. 2019. Maximizing 2-D Beam Brightness Using the Round to Flat Beam Transformation in the Ultralow Charge Regime. *Proceedings of the North American Particle Accelerator Conference* NAPAC2019 (2019), 4 pages, 0.649 MB. doi:10.18429/JACOW-NAPAC2019-FRXBA4 Artwork Size: 4 pages, 0.649 MB ISBN: 9783954502233 Medium: PDF.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. 2016. Density estimation using Real NVP. https://arxiv.org/abs/1605.08803v3

Auralee Edelen and Xiaobiao Huang. 2024. Machine Learning for Design and Control of Particle Accelerators: A Look Backward and Forward. *Annual Review of Nuclear and Particle Science* 74, 1 (Sept. 2024), 557–581. doi:10.1146/annurev-nucl-121423-100719

Auralee Edelen, Christopher Mayes, Daniel Bowring, Daniel Ratner, Andreas Adelmann, Rasmus Ischebeck, Jochem Snuverink, Ilya Agapov, Raimund Kammering, Jonathan Edelen, Ivan Bazarov, Gianluca Valentino, and Jorg Wenninger. 2018. Opportunities in Machine Learning for Particle Accelerators. doi:10.48550/arXiv.1811.03172 arXiv:1811.03172 [physics].

J Gonzalez-Aguilera, Y Kim, R Roussel, A Edelen, and C Mayes. 2023. Towards fully differentiable accelerator modeling. (2023).

Verena Kain, Simon Hirlander, Brennan Goddard, Francesco Maria Velotti, Giovanni Zevi Della Porta, Niky Bruchon, and Gianluca Valentino. 2020. Sample-efficient reinforcement learning for CERN accelerator control. *Physical Review Accelerators and Beams* 23, 12 (Dec. 2020), 124801. doi:10.1103/PhysRevAccelBeams.23.124801

Jan Kaiser, Chenran Xu, Annika Eichler, and Andrea Santamaria Garcia. 2024. Bridging the gap between machine learning and particle accelerator physics with high-speed, differentiable simulations. *Physical Review Accelerators and Beams* 27, 5 (May 2024), 054601. doi:10.1103/PhysRevAccelBeams.27.054601

Diederik P. Kingma and Jimmy Ba. 2017. Adam: A Method for Stochastic Optimization. doi:10.48550/arXiv.1412.6980 arXiv:1412.6980 [cs.LG].

Aashwin Mishra, Matt Seaberg, Ryan Roussel, Fred Poitevin, Jana Thayer, Daniel Ratner, Auralee Edelen, and Apurva Mehta. 2025. A Start to End Machine Learning Approach to Maximize Scientific Throughput from the LCLS-II-HE. *Synchrotron Radiation News* 38, 4 (July 2025), 10–17. doi:10.1080/08940886.2025.2538420

Aashwin Mishra, Matthew Seaberg, Ryan Roussel, Sanghoon Song, Auralee Edelen, Daniel Ratner, and Apurva Mehta. 2026. Data driven drift correction for complex optical systems. *Journal of Synchrotron Radiation* 33, 3 (May 2026), 596–603. doi:10.1107/S1600577526003395

Michael C Mozer. 2013. A focused backpropagation algorithm for temporal pattern recognition. In *Backpropagation*. Psychology Press, 137–169.

Ji Qiang, Steve Lidia, Robert D. Ryne, and Cecile Limborg-Deprey. 2006. Three-dimensional quasistatic model for high brightness beam dynamics simulation. *Physical Review Special Topics - Accelerators and Beams* 9, 4 (April 2006), 044204. doi:10.1103/PhysRevSTAB.9.044204

Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, Maximilian Ernestus, and Noah Dormann. 2021. Stable-Baselines3: Reliable Reinforcement Learning Implementations. *Journal of Machine Learning Research* 22, 268 (2021), 1–8. https://jmlr.org/papers/v22/20-1364.html

Mahindra Rautela, Alan Williams, and Alexander Scheinker. 2024. Towards latent space evolution of spatiotemporal dynamics of six-dimensional phase space of charged particle beams. (July 2024), 909–912 pages, 3.7 MB. doi:10.18429/JACoW-IPAC2024-MOPS75 arXiv:2406.01535 [physics].

Ryan Roussel, Gopika Bhardwaj, Dylan Kennedy, Chris Garnier, An Le, William Colocho, Michael Ehrlichman, Yuantao Ding, Feng Zhou, and Auralee Edelen. 2026. Autonomous operation of the DIAG0 diagnostic line for 6D phase-space monitoring at LCLS-II. doi:10.48550/arXiv.2604.20125 arXiv:2604.20125 [physics].

Ryan Roussel, Juan Pablo Gonzalez-Aguilera, Eric Wisniewski, Alexander Ody, Wanming Liu, John Power, Young-Kee Kim, and Auralee Edelen. 2024. Efficient six-dimensional phase space reconstructions from experimental measurements using generative machine learning. *Physical Review Accelerators and Beams* 27, 9 (Sept. 2024), 094601. doi:10.1103/PhysRevAccelBeams.27.094601

R Roussel, C Mayes, A Edelen, and A Bartnik. 2023. Xopt: A simplified framework for optimization of accelerator problems using advanced algorithms. (2023).

D. Sagan. 2006. Bmad: A relativistic charged particle simulation library. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 558, 1 (March 2006), 356–359. doi:10.1016/j.nima.2005.11.001

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal Policy Optimization Algorithms. doi:10.48550/arXiv.1707.06347 arXiv:1707.06347 [cs.LG].

M G Signorelli, J P Devlin, G H Hoffstaetter, and D Sagan. 2026. SCIBMAD: A DIFFERENTIABLE, GPU-PARALLELIZED SOFTWARE LIBRARY FOR PARTICLE ACCELERATOR DESIGN, NONLINEAR ANALYSIS, AND MACHINE LEARNING. (2026).

Oliver Stein, Ilya Agapov, Annika Eichler, and Jan Kaiser. 2022. Accelerating Linear Beam Dynamics Simulations for Machine Learning Applications. (2022), 4 pages, 0.139 MB. doi:10.18429/JACOW-IPAC2022-WEPOMS036 Artwork Size: 4 pages, 0.139 MB ISBN: 9783954502271 Medium: application/pdf.

D. Storey, C. Zhang, P. San Miguel Claveria, G.J. Cao, E. Adli, L. Alsberg, R. Ariniello, C. Clarke, S. Corde, T.N. Dalichaouch, C.E. Doss, H. Ekerfelt, C. Emma, E. Gerstmayr, S. Gessner, M. Gilljohann, C. Hast, A. Knetsch, V. Lee, M. Litos, R. Loney, K.A. Marsh, A. Matheron, W.B. Mori, Z. Nie, B. O'Shea, M. Parker, G. White, G. Yocky, V. Zakharova, M.J. Hogan, and C. Joshi. 2024. Wakefield generation in hydrogen and lithium plasmas at FACET-II: Diagnostics and first beam-plasma interaction results. *Physical Review Accelerators and Beams* 27, 5 (May 2024), 051302. doi:10.1103/PhysRevAccelBeams.27.051302

Helmut Wiedemann. 2015. *Particle Accelerator Physics*. Springer International Publishing, Cham. doi:10.1007/978-3-319-18317-6

Edward J. N. Wilson. 2001. *An introduction to particle accelerators*. Oxford University Press, Oxford. doi:10.1093/oso/9780198508298.001.0001

Jie Xu, Viktor Makoviychuk, Yashraj Narang, Fabio Ramos, Wojciech Matusik, Animesh Garg, and Miles Macklin. 2022. Accelerated Policy Learning with Parallel Differentiable Simulation. doi:10.48550/arXiv.2204.07137 arXiv:2204.07137 [cs.LG].

V. Yakimenko, L. Alsberg, E. Bong, G. Bouchard, C. Clarke, C. Emma, S. Green, C. Hast, M.J. Hogan, J. Seabury, N. Lipkowitz, B. O'Shea, D. Storey, G. White, and G. Yocky. 2019. FACET-II facility for advanced accelerator experimental tests. *Physical Review Accelerators and Beams* 22, 10 (Oct. 2019), 101301. doi:10.1103/PhysRevAccelBeams.22.101301

A Normalizing Flow Architecture

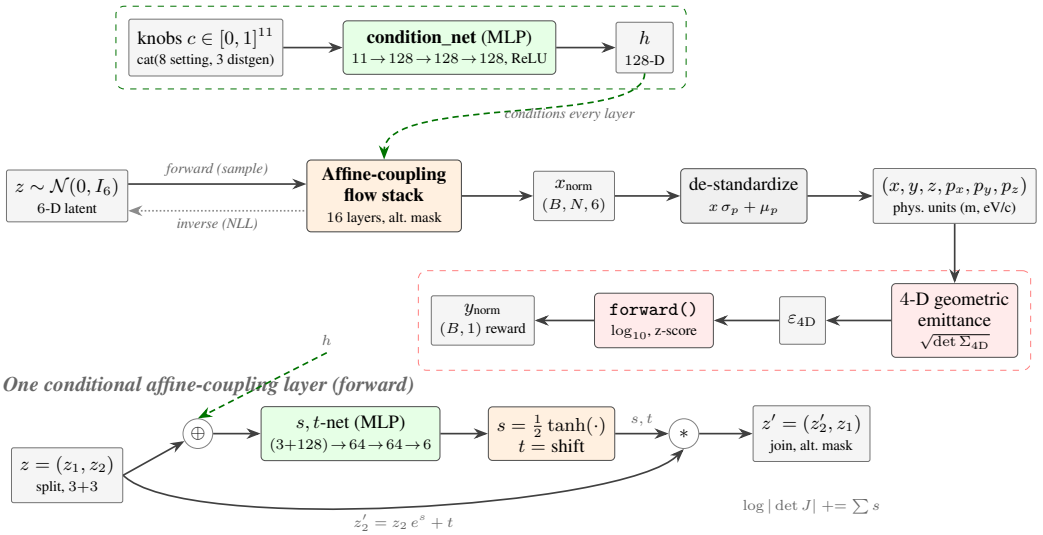

Figure 7: Outline of conditional normalizing flow architecture used in the particle dynamics model.

B Drift Stability

The MLP-trained surrogate models were also evaluated for control against drifting hidden state variables, similar to drifts in anticipated operation. Specifically, the 6 initial particle distribution parameters from the photocathode output were updated in a random-walk manner with different standard deviations. The trajectory-averaged ε_{4D} was tracked in MLP surrogate evaluation for controllers not trained on drifting parameters (figure 8). The three algorithms had comparable performance, with slightly more variability in PPO performance between seeds. As expected, performance degrades with increasing drift, but no sudden changes in tuning performance were observed.

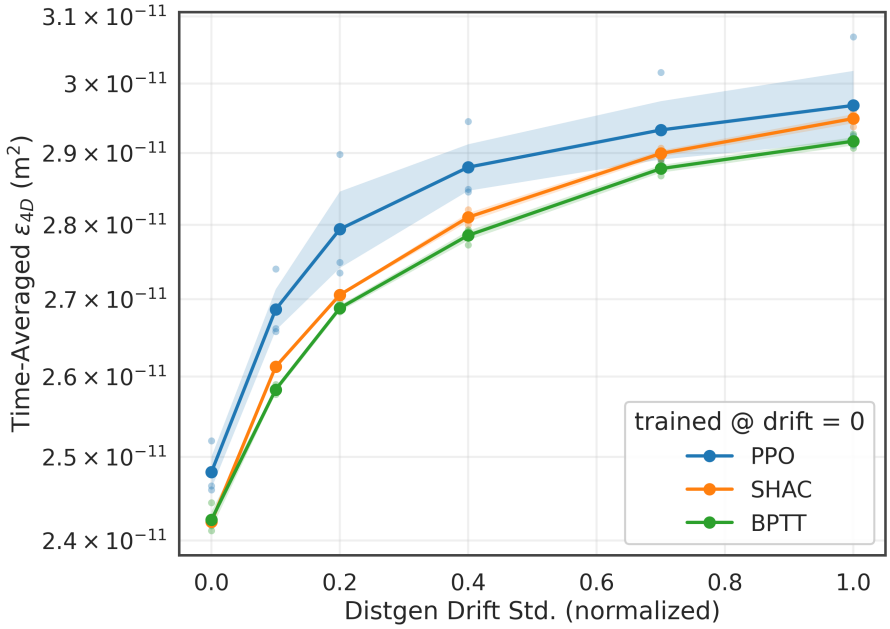

Figure 8: Time-averaged emittance over surrogate evaluation rollout trajectories for RL controllers trained without initial particle bunch drift. The drift in the initial distribution is determined as a random walk with standard deviation in normalized units on the x-axis. Performance is evaluated on three seeds of each algorithm.