Beyond PID Controllers: PPO with Neuralized PID Policy for Proton Beam Intensity Control in Mu2e

C. Xu*, J.YC. Hu*, J. Jiang, S. Memik, R. Shi, A.M. Shuping, M. Thieme, H. Liu,
Northwestern University†, Evanston, IL USA

M.R. Austin, J.M. Arnold, J.R. Berlioz, P. Hanlet, K.J. Hazelwood, M.A. Ibrahim, J.St. John, J. Mitrevski, V.P. Nagaslaev, D.J. Nicklaus, G. Pradhan, A.L. Saewert, B.A. Schupbach, K. Seiya, R.M. Thurman-Keup, N.V. Tran, A. Narayanan,
Fermi National Accelerator Laboratory‡, Batavia, IL USA

Abstract

We introduce a novel Proximal Policy Optimization (PPO) algorithm aimed at addressing the challenge of maintaining a uniform proton beam intensity delivery in the **Mu**on **to** **E**lectron Conversion Experiment (**Mu2e**) at Fermi National Accelerator Laboratory (Fermilab). Our primary objective is to regulate the spill process to ensure a consistent intensity profile, with the ultimate goal of creating an automated controller capable of providing real-time feedback and calibration of the **S**pill **R**egulation **S**ystem (**SRS**) parameters on a millisecond timescale. We treat the Mu2e accelerator system as a Markov Decision Process suitable for Reinforcement Learning (RL), utilizing PPO to reduce bias and enhance training stability. A key innovation in our approach is the integration of neuralized **P**roportional-**I**ntegral-**D**erivative (**PID**) controller into the policy fucntion, resulting in a significant improvement in the **S**pill **D**uty **F**actor (**SDF**) by 13.6%, surpassing the performance of the current PID controller baseline by an additional 1.6%. This paper presents the preliminary offline results based on a differentiable simulator of the Mu2e accelerator. It paves the ground works for real-time implementations and applications, representing a crucial step towards automated proton beam intensity control for the Mu2e experiment.

1 Introduction

We propose a novel RL-enhanced spill regularization system that incorporates a neuralized PID policy function to tackle the beam regularization challenge in the Mu2e experiments [Bartoszek et al., 2015] at Fermilab. Our objective is to create an automated controller that ensures consistent spill (proton beams) intensity during experiments meeting real-time control requirements [Narayanan et al., 2021b]. This objective falls under the broader scope of the Accelerator **R**eal-time **E**dge **AI** for **D**istributed **S**ystems (**READS**) project [Mitrevski, 2023, Seiya et al., 2021b, Hazelwood et al., 2021a]. To achieve this, we model the Mu2e accelerator system as a Markov Decision Process and employ the Proximal Policy Optimization (PPO) algorithm [Schulman et al., 2017] to cast the spill regulations as sequential decision-making problems. Our main contribution is the integration of a neuralized PID policy function [Zribi et al., 2018] for our RL framework, encompassing the inductive bias of the standard PID controller (i.e. the proportional, integral, and derivative information) to better capture states at different stages. Our experiments on the Mu2e simulator show that we observed an average improvement of 13.6% in the **S**pill **D**uty **F**actor (**SDF**). Additionally, our method outperforms the previous PID controller approach [Narayanan et al., 2021b].

*Equal contribution

†Performed at Northwestern with support from the Departments of Computer Science and Electrical and Computer Engineering

‡Operated by Fermi Research Alliance, LLC under Contract No.De-AC02-07CH11359 with the United States Department of Energy. Additional funding provided by Grant Award No. LAB 20-2261 [Department of Energy, Office of Science, 2020]

Machine Learning and the Physical Sciences Workshop, NeurIPS 2023.

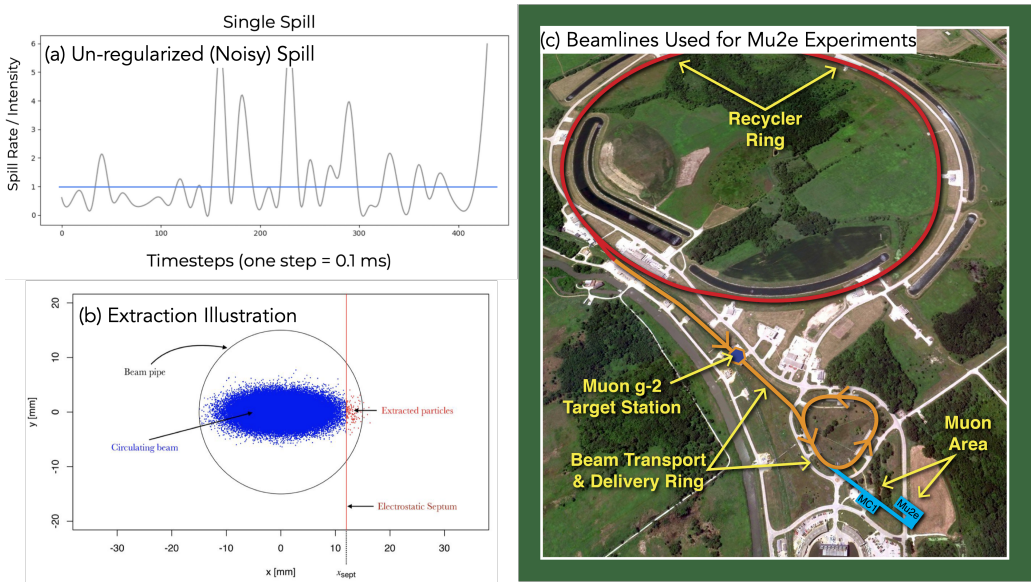

Figure 1: **(a)**: The extraction (or 'spill') of protons from the Delivery Ring is noisy (deviates from 1) without any regulation. **(b)**: A snapshot of the beam in physical space at the extraction location. As the horizontal beam size increases, a slice of circulating beam (that is past the position of the electrostatic septum) is extracted. **(c)**: To create the muons, proton pulses are made to hit a production target and muons are obtained from the secondaries. The proton pulses with the required time structure are created by extracting them from an accelerator ring called Delivery Ring at Fermilab and sending it to the Mu2e production target.

The goal of Mu2e experiment [Bartoszek et al., 2015, Bernstein, 2019, Narayanan et al., 2021a, 2022] at Fermilab is to search for new physics by studying the decay of muons into electrons. This intricate experiment places stringent demands on the quality of the proton beam directed at the muon production target, see Figure 1. These requirements are essential to minimize background particle physics processes that could obscure the discovery signal. One of the key prerequisites for the Mu2e experiment includes achieving a highly uniform extracted beam intensity during each spill of protons with 8 GeV kinetic energy. In order to achieve this, the **S**pill **R**egulation **S**ystem (**SRS**) [Thieme, 2022, Narayanan et al., 2021b, Ibrahim et al., 2019] is being developed to govern the extraction process of the beam and mitigate various sources of fluctuations in the spill profile.

The SRS adjusts the power supply currents of three dedicated fast quadrupoles (resulting in varying its magnetic fields) and this variation results in controlling the variations in the spill intensity (Figure 1 (c)). Ideally, the SRS aims for the spill intensity to be perfectly uniform.

Our approach involves utilizing a Proximal Policy Optimization model with a differentiable Mu2e simulator [Narayanan et al., 2021b] to regulate random generated spills. In each episode (or spill), the simulator generates a series of random spills, each with varying intensities (shown in Figure 1 (a)). Subsequently, the PPO model intervenes in each individual time step to correct these generated spills by adjusting the control signal of the Mu2e simulator. The primary goal of our model is to bring spill rate of all spills as close to a value of 1 as possible. To optimize this objective function, while mitigating the influence of excessively high or low intensity spills, we implement an exponential moving average (EMA) to measure the deviation of the sequence (spill rate) from the desired value of 1. Additionally, we incorporate neuralized PID controller, encompassing proportional, integral, and derivative components in the state representation. By employing different random seeds, our simulator can generate a diverse profile of noisy spills, reflecting the real-world scenarios.

We demonstrate superior performance by numerically benchmarking our methods with PID controller. Specifically, our experiments compare SDF (defined in (2.1)) performance on different seeds. Our results show that our methods consistently improve the SDF by 13.6% for 9 random generated spills and achieve 1.6% improvements compared to the PID controller.

The rest of this paper commences by a detailed description of our proposed method. Subsequently, we present numerical results to showcase the performance of our approach. Finally, we conclude this paper with a discussion of potential future directions and avenues for further research.

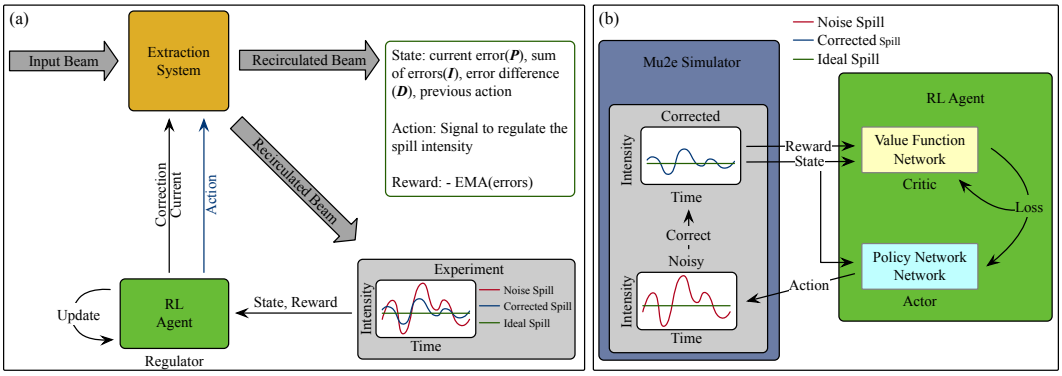

Figure 2: **(a)**: The Mu2e simulator initially generates the noised spill data. The agent proceeds to adjust the spill and the code employs this adjusted spill to compute relevant information such as the state and reward. These pieces of information are instrumental in training the RL agent, which in turn offers new actions for the subsequent time step to refine the spill. **(b)**: The simulator refines (corrects) the spill derived from noisy data. It conveys the state and reward, calculated using the corrected spill, to update the value network responsible for evaluating the quality of the correction. Subsequently, the state and loss generated by the value network contribute to the adaptation of the policy network. The policy network, in response, generates new actions for spill regulation.

2 Methodology

In this section, we first introduce the machine learning's role in optimizing Fermilab's accelerator parameters. Then, we provide a detailed design of our RL-enhanced regularization system.

2.1 Problem Setup

Large fluctuations in the proton spill rate result in large fluctuations in the intensity of produced muons. And this in turn causes background effects that hinder the signals of new physics. To combat these, we approach the beam intensity regulation challenge, specifically controlling the spill regulation system, as a tracking control problem. The objective is to keep certain signals (spill intensity) close to specific reference values ($\simeq 1$) by controlling the quadrupole currents (power supply of 3 dedicated fast-ramping quadrupoles) in the regulation system. By doing so, we adjust the magnetic field, which subsequently adjusts the beam intensity throughout the delivery ring, as depicted in Figure 1 (c).

Objective. To handle the constraint challenges posed by Mu2e experiment, we regulate the uniformity of the extracted spill by increasing its Spill Duty Factor (SDF),

$$\text{SDF} := 1/\left(1+\sigma^2_{\text{spill}}\right), \tag{2.1}$$

by regulating the extraction process (where σ_{spill} is the standard deviation in the spill rate). The ultimate goal for SRS in the Mu2e experiment is to achieve a SDF of 0.6 or higher, with an ideal spill having a constant spill rate value of 1 and an SDF of 1.

Mu2e Simulator. We employ a differentiable simulator proposed in [Narayanan et al., 2021b] to replicate the beam physics process in Mu2e experiments realistically. This physics simulator generates spill intensity and the associated data. It subsequently conveys this data to the RL agent, which aids in training the RL model. Once trained, the RL model transmits control signals back to the simulator, allowing it to regulate the *deviation* in the spill rate based on these signals and provide the modified data to the RL agent. This process is shown in Figure 2 (a).

2.2 Proximal Policy Optimization (PPO) Controller for Spill Regulation System

Reward Function. Let x_t be the observation of one single spill signal at time step t and $\sigma = 1$ be the corresponding target reference value. The reward function at t is defined as the exponential moving average

$$r_t = -\text{EMA}(t,\alpha), \alpha \in [0,1]\,, \text{ where } \text{EMA}(t,\alpha) = \alpha|x_t-\sigma| + (1-\alpha)\text{EMA}(t-1,\alpha). \tag{2.2}$$

EMA gives more weight to recent spills and less weight to older spills. This helps in reducing the impact of short-term fluctuations in the spills, making it easier to identify underlying trends.

PID Controller. A PID controller in discrete-time operation captures past details regarding tracking errors, their integrals, and derivatives within a linear control strategy. We denote the time series of spill signal at time t as $o_t = (x_0, x_1, \cdots, x_t)$. In formal terms, the discrete-time PID controller's policy, characterized by its parameters K_P, K_I, and K_D, is expressed as:

$$\pi^{\text{PID}}(o_t) = K_P\left(x_t-\sigma\right) + K_I\sum_{\tau=0}\left(x_\tau-\sigma\right) + K_D\frac{\left(x_t-\sigma\right)-\left(x_{t-1}-\sigma\right)}{\Delta t}, \tag{2.3}$$

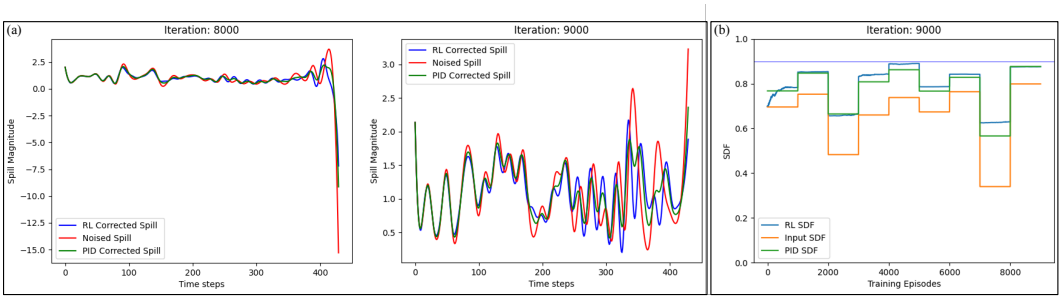

Figure 3: Spill intensity and SDF comparison in different seeds. **(LHS)**: Comparison of Spill Intensity: The spill intensity corrected by RL is closer to 1 when compared to the PID-corrected spill. **(RHS)**: Comparison of SDF: After 600 training iterations, the SDF achieved by RL outperforms or nears the SDF obtained through PID.

where K_P, K_I, K_D are tuneable scalar coefficients and Δt is the discrete time interval.

Model: PPO with Neuralized PID Policy. Our model leverages the inductive bias of PID (Proportional-Integral-Derivative) controller, and propose a neuralized PID policy function. Specifically, we incorporate tracking errors, integrals, and derivatives as components of the state vector s_t. Furthermore, we employ a linear network to parametrize the standard PID controller (2.3), and use it as a part of our policy function. Therefore, the policy function of our model not only enables the extraction of external information (based on previous actions) but also includes the PID control signals K_P, K_I, and K_D as its learnable parameters. Remarkably, when learned effectively, our policy network outperforms the standard PID controller (2.3), making it a highly adaptable solution.

Our approach incorporates three key components: the PPO algorithm, the EMA reward function, and the neuralized PID controller. The PPO algorithm [Schulman et al., 2017], a reinforcement learning technique, plays a central role in optimizing policy functions to enhance decision-making in sequential tasks and we specifically chose PPO to refine our reward function. Rather than relying on a single spill for reward computation, we employ EMA within the reward function. This approach allows us to both capture trends across a sequence of spills and mitigate fluctuations. Additionally, we integrate the neuralized PID bias into our policy network. In this setup, our policy network consists of a trainable PID controller and a linear projection of past actions. Let's denote the state at time step t as s_t, the action as a_t. We combine the PID policy π^{PID} and action policy π^{action} to formulate RL policies as π. The variable a_t represents the control signal used for regulating the spill. The state s_t encompasses both the previous action a_{t-1} and the time series of the spill signal o_t. As a result, the action at time t, a_t, can be represented as follows:

$$a_t = \pi(s_t) = \pi^{\text{PID}}(o_t) + \pi^{\text{action}}(a_{t-1}). \tag{2.4}$$

The learning process is shown in Figure 2 (b).

3 Experimental Studies

We validate our method by using the Mu2e differentialble simulator [Narayanan et al., 2021b] as the RL environment. Our experiments involve the utilization of both the Mu2e simulator and our RL-enhanced spill regularization system.

Settings. We configure various hyperparameters for both the simulator and the RL agent in our experiment. Specifically, we configure the simulator to generate 430 spills per iteration, equivalent to 10 data points per millisecond within a 43 ms spill duration, aligning closely with realistic settings as detailed in [Narayanan et al., 2021b]. In terms of reward calculation, we choose a value of $\alpha = 0.5$ for the EMA component. Regarding the RL agent, we employ the **`stable-baselines3`**[4] PPO model [Raffin et al., 2021]. The actor network is designed as a single linear layer, while the critic network took the form of a two-hidden layer (64×64) MLP network. The learning rate is set at 1×10^{-4}. We change the random seed every 1000 (spills) epochs.

Results. In Figure 3, we examine the spill intensity in scenarios involving unregularized, PID-regularized, and RL-regularized setups. Figure 3 (a) demonstrates that the spill corrected by RL brings the spill rate closer to the ideal value of 1 when compared to the PID-corrected spill, on the first random seed configuration. Furthermore, we provide a visual representation of the evolution of the SDF during the training process. Figure 3 (b) illustrates that the SDF achieved through RL uniformly surpasses (or approaches) that of PID regulation after 600 episodes.

[4] https://github.com/DLR-RM/stable-baselines3

Table 1: Evaluation of the impact of various parameters. "v.s. PID" represents the SDF improvement by comparing our methods to PID method. "v.s. Noise" is the SDF improvements of our methods compared to unregularized SDF. NN: neural network policy; PID: neuralized PID policy; EMA: exponential moving average of errors; α: EMA's smooth parameter; SUM: sum of errors; PPO: Proximal Policy Optimization; SAC: Soft-Actor Critic; P: current error; I: sum of errors; D: error difference; Act: previous action; CD: corrected spill difference; Over-1: number of noisy spill intensity ≥ 1.

v.s. PID	v.s. Noise	Policy Network	Reward Func.	RL Algorthim	State
-4.11	7.90	PID	-EMA (α=0.1)	PPO	P, I, D, Act
-11.12	0.89	NN	-EMA (α=0.5)	PPO	P, I, D, Act
1.42	13.55	PID	-EMA (α=0.9)	PPO	P, I, D, Act
1.95	-10.06	PID	-SUM	PPO	P, I, D, Act
1.46	13.47	PID	-EMA (α=0.5)	PPO	P, I, D
5.76	-6.25	PID	-EMA (α=0.5)	PPO	CD, Over-1, P, Act
-12.60	-24.61	PID	-EMA (α=0.5)	SAC	P, I, D, Act
1.65	**13.67**	PID	-EMA (α=0.5)	PPO	P, I, D, Act

Ablation Study. We systematically vary parameters and model architectural configurations to differentiate their impact in these four aspects: 1), we compare the usage of PID controller and a neural network in Policy network's; 2), we experiment with various values of the smoothing factor (α) in the Exponential Moving Average (EMA); 3), we explore the effect of employing different reward functions; 4), we select alternative states from the Mu2e (spills) environment to replace the usage of inductive biases from the PID; 5), we compare two RL algorthims: Soft-Actor Critic (SAC) [Haarnoja et al., 2018] and PPO. For all the experiments, we use the same random seed series. We report the average improvements of SDF of our proposed methods, compared to PID regularized SDF and unregularized SDF in Table 1.

4 Related Works in Real-time Edge AI for Distributed Systems (READS)

The READS project [Seiya et al., 2021a] includes two primary sub-projects: 1) Beam Loss Deblending for the Main Injector and Recycler, and 2) Mu2e Spill Regulation, which is the subject of this work.

Beam Loss Deblending for Main Injector and Recycler. In this sub-project, the challenge is distinguishing between beam losses from two adjacent accelerators (Recycler Ring and Main Injector) that share the same monitoring system. We summarize some related works in resolving this challenge bellow. The Deblending model (DBLN) [Hazelwood et al., 2021b] uses a dual-network MLP setup to classify beam losses and predict their probabilities for each accelerator and beam loss monitor (BLM). The Many Models [Hazelwood, 2023] employs individual MLP models for each BLM and aggregates their outputs to enhance local pattern recognition for beam losses. The semantic regression model [Thieme et al., 2022] uses the U-Net architecture to capture each accelerator's localized and extensive beam loss patterns. As for hardware, a series of work on Field Programmable Gate Array (FPGA)-based edge-AI systems [Berlioz et al., 2022, Ibrahim et al., 2023, Shi et al., 2023, Arnold et al., 2023] are developed and deployed for real-time identification of beam loss sources in accelerator complex, enhancing operational accuracy and efficiency.

Mu2e Spill Regulation. There are some related works in solving challenges (as described in Section 2.1) in Mu2e Spill Regulation project. The PID method [Narayanan et al., 2021b] employs a hybrid machine learning approach that integrates a neural network (NN) to optimize PID controller gains within an end-to-end machine learning (ML) differentiable simulator simulator, thereby enhancing the spill quality of a slow extraction system for proton delivery. The GRU network [Narayanan et al., 2022] ingests a history of spill intensity observations as input and makes predictions for the quadrupole adjustments based on these inputs.

5 Conclusion

We present an innovative RL-enhanced spill control system, utilizing a neural PID controller as the policy function, to tackle beam regulation issues in Mu2e experiments. To simulate real-world spill control scenarios, we utilize a differentiable Mu2e simulator to create spills, improve spill adjustments, and establish reward signals. Furthermore, we harness an RL-based controller for fine-tuning control signals during the regularization process. Our approach outperforms the PID-based regularization model, achieving a 1.6% higher SDF performance, with a 13.6% improvement in the SDF of unregularized spills, confirmed across nine different settings with random seeds.

Acknowledgments

JH and CW would like to thank Jiayi Wang for facilitating experimental deployments. The authors would like to thank the anonymous reviewers and program chairs for constructive comments. AN was previously affiliated with Northern Illinois University for the earlier part of the READS collaboration, DeKalb, IL, USA. Besides the aforementioned support, JH is partially supported by the Walter P. Murphy Fellowship. HL is partially supported by NIH R01LM1372201, NSF CAREER1841569, DOE DE-AC02-07CH11359 and a NSF TRIPODS1740735. This research was supported in part through the computational resources and staff contributions provided for the Quest high performance computing facility at Northwestern University which is jointly supported by the Office of the Provost, the Office for Research, and Northwestern University Information Technology. The content is solely the responsibility of the authors and does not necessarily represent the official views of the funding agencies.

References

Jeremy Arnold, Mark Austin, Jose Berlioz, Dave Bracey, Pierrick Hanlet, Kyle Hazelwood, Jerry Yao-Chieh Hu, Aisha Ibrahim, Jing Jang, Han Liu, et al. Edge ai for accelerator controls (reads): beam loss deblending. Technical report, Fermi National Accelerator Laboratory (FNAL), Batavia, IL (United States), 2023.

L. Bartoszek, E. Barnes, J. P. Miller, J. Mott, A. Palladino, J. Quirk, B. L. Roberts, J. Crnkovic, V. Polychronakos, V. Tishchenko, P. Yamin, C. h. Cheng, B. Echenard, K. Flood, D. G. Hitlin, J. H. Kim, T. S. Miyashita, F. C. Porter, M. Röhrken, J. Trevor, R. Y. Zhu, E. Heckmaier, T. I. Kang, G. Lim, W. Molzon, Z. You, A. M. Artikov, J. A. Budagov, Yu. I. Davydov, V. V. Glagolev, A. V. Simonenko, Z. U. Usubov, S. H. Oh, C. Wang, G. Ambrosio, N. Andreev, D. Arnold, M. Ball, R. H. Bernstein, A. Bianchi, K. Biery, R. Bossert, M. Bowden, J. Brandt, G. Brown, H. Brown, M. Buehler, M. Campbell, S. Cheban, M. Chen, J. Coghill, R. Coleman, C. Crowley, A. Deshpande, G. Deuerling, J. Dey, N. Dhanaraj, M. Dinnon, S. Dixon, B. Drendel, N. Eddy, R. Evans, D. Evbota, J. Fagan, S. Feher, B. Fellenz, H. Friedsam, G. Gallo, A. Gaponenko, M. Gardner, S. Gaugel, K. Genser, G. Ginther, H. Glass, D. Glenzinski, D. Hahn, S. Hansen, B. Hartsell, S. Hays, J. A. Hocker, E. Huedem, D. Huffman, A. Ibrahim, C. Johnstone, V. Kashikhin, V. V. Kashikhin, P. Kasper, T. Kiper, D. Knapp, K. Knoepfel, L. Kokoska, M. Kozlovsky, G. Krafczyk, M. Kramp, S. Krave, K. Krempetz, R. K. Kutschke, R. Kwarciany, T. Lackowski, M. J. Lamm, M. Larwill, F. Leavell, D. Leeb, A. Leveling, D. Lincoln, V. Logashenko, V. Lombardo, M. L. Lopes, A. Makulski, A. Martinez, D. McArthur, F. McConologue, L. Michelotti, N. Mokhov, J. Morgan, A. Mukherjee, P. Murat, V. Nagaslaev, D. V. Neuffer, T. Nicol, J. Niehoff, J. Nogiec, M. Olson, D. Orris, R. Ostojic, T. Page, C. Park, T. Peterson, R. Pilipenko, A. Pla-Dalmau, V. Poloubotko, M. Popovic, E. Prebys, P. Prieto, V. Pronskikh, D. Pushka, R. Rabehl, R. E. Ray, R. Rechenmacher, R. Rivera, W. Robotham, P. Rubinov, V. L. Rusu, V. Scarpine, W. Schappert, D. Schoo, A. Stefanik, D. Still, Z. Tang, N. Tanovic, M. Tartaglia, G. Tassotto, D. Tinsley, R. S. Tschirhart, G. Vogel, R. Wagner, R. Wands, M. Wang, S. Werkema, H. B. White Jr. au2, J. Whitmore, R. Wielgos, R. Woods, C. Worel, R. Zifko, P. Ciambrone, F. Colao, M. Cordelli, G. Corradi, E. Dane, S. Giovannella, F. Happacher, A. Luca, S. Miscetti, B. Ponzio, G. Pileggi, A. Saputi, I. Sarra, R. S. Soleti, V. Stomaci, M. Martini, P. Fabbricatore, S. Farinon, R. Musenich, D. Alexander, A. Daniel, A. Empl, E. V. Hungerford, K. Lau, G. D. Gollin, C. Huang, D. Roderick, B. Trundy, D. Na. Brown, D. Ding, Yu. G. Kolomensky, M. J. Lee, M. Cascella, F. Grancagnolo, F. Ignatov, A. Innocente, A. L'Erario, A. Miccoli, A. Maffezzoli, P. Mazzotta, G. Onorato, G. M. Piacentino, S. Rella, F. Rossetti, M. Spedicato, G. Tassielli, A. Taurino, G. Zavarise, R. Hooper, D. No. Brown, R. Djilkibaev, V. Matushko, C. Ankenbrandt, S. Boi, A. Dychkant, D. Hedin, Z. Hodge, V. Khalatian, R. Majewski, L. Martin, U. Okafor, N. Pohlman, R. S. Riddel, A. Shellito, A. L. de Gouvea, F. Cervelli, R. Carosi, S. Di Falco, S. Donati, T. Lomtadze, G. Pezzullo, L. Ristori, F. Spinella, M. Jones, M. D. Corcoran, J. Orduna, D. Rivera, R. Bennett, O. Caretta, T. Davenne, C. Densham, P. Loveridge, J. Odell, R. Bomgardner, E. C. Dukes, R. Ehrlich, M. Frank, S. Goadhouse, R. Group, E. Ho, H. Ma, Y. Oksuzian, J. Purvis, Y. Wu, D. W. Hertzog, P. Kammel, K. R. Lynch, and J. L. Popp. Mu2e technical design report, 2015.

J.R. Berlioz, K.J. Hazelwood, M.A. Ibrahim, M.R. Austin, J.M. Arnold, B.A. Schupbach, K. Seiya, and R.M. Thurman-Keup. Synchronous High-Frequency Distributed Readout for Edge Processing at the Fermilab Main Injector and Recycler. JACoW Publishing, Geneva, Switzerland, 2022.

URL https://napac2022.vrws.de/papers/mopa15.pdf. presented at NAPAC'22, Albuquerque, New Mexico, USA, Aug. 2022, paper MOPA15, unpublished.

Robert H Bernstein. The mu2e experiment. *Frontiers in Physics*, 7:1, 2019.

Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Misha Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. Decision transformer: Reinforcement learning via sequence modeling. *Advances in neural information processing systems*, 34:15084–15097, 2021.

Department of Energy, Office of Science. Data, Artificial Intelligence, and Machine Learning at DOE Scientific User Facilities, 2020. URL https://science.osti.gov/-/media/grants/pdf/lab-announcements/2020/LAB_20-2261.pdf.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. *CoRR*, abs/1801.01290, 2018. URL http://arxiv.org/abs/1801.01290.

K.J. Hazelwood. Disentangling beam losses in the fermilab main injector enclosure using real-time edge ai, 10 2023. presented at The 19th Biennial International Conference on Accelerator and Large Experimental Physics Control Systems (ICALEPCS) 2023, Cape Town, South Africa.

K.J. Hazelwood et al. Real-Time Edge AI for Distributed Systems (READS): Progress on Beam Loss De-Blending for the Fermilab Main Injector and Recycler. In *Proc. IPAC'21*, number 12 in International Particle Accelerator Conference, pages 912–915. JACoW Publishing, Geneva, Switzerland, 08 2021a. ISBN 978-3-95450-214-1. doi: 10.18429/JACoW-IPAC2021-MOPAB288. URL https://jacow.org/ipac2021/papers/mopab288.pdf.

Kyle Hazelwood, Mark Austin, Michelle Ibrahim, Han Liu, Seda Memik, Vladimir Nagaslaev, Aakaash Narayanan, Dennis Nicklaus, Andrea Saewert, Brian Schupbach, et al. Real-time edge ai for distributed systems (reads): Progress on beam loss de-blending for the fermilab main injector and recycler. Technical report, Fermi National Accelerator Lab.(FNAL), Batavia, IL (United States), 2021b.

Jerry Yao-Chieh Hu, Donglin Yang, Dennis Wu, Chenwei Xu, Bo-Yu Chen, and Han Liu. On sparse modern hopfield model. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2309.12673.

MA Ibrahim, Ed Cullerton, JS Diamond, KS Martin, PS Prieto, E Scarpine, and Philip Varghese. Preliminary design of mu2e spill regulation system (srs). Technical report, Fermi National Accelerator Lab.(FNAL), Batavia, IL (United States), 2019.

MA Ibrahim, MR Austin, JM Arnold, JR Berlioz, P Hanlet, KJ Hazelwood, J Mitrevski, VP Nagaslaev, DJ Nicklaus, G Pradhan, et al. Fpga architectures for distributed ml systems for real-time beam loss de-blending. Technical report, Fermi National Accelerator Laboratory (FNAL), Batavia, IL (United States), 2023.

Michael Janner, Qiyang Li, and Sergey Levine. Offline reinforcement learning as one big sequence modeling problem. *Advances in neural information processing systems*, 34:1273–1286, 2021.

Yanrong Ji, Zhihan Zhou, Han Liu, and Ramana V Davuluri. Dnabert: pre-trained bidirectional encoder representations from transformers model for dna-language in genome. *Bioinformatics*, 37 (15):2112–2120, 2021.

J. Mitrevski. Edge ai for accelerator controls (reads): Beam loss deblending, 09 2023. URL https://indico.cern.ch/event/1283970/contributions/5550643/attachments/2721973/4729145/READS%20FastML%20v3.pdf. presented at Fast Machine Learning For Science Workshop 2023, London UK.

A. Narayanan et al. Optimizing Mu2e Spill Regulation System Algorithms. In *Proc. IPAC'21*, pages 4281–4284. JACoW Publishing, Geneva, Switzerland, 2021a. doi: 10.18429/JACoW-IPAC2021-THPAB243. URL https://jacow.org/ipac2021/papers/THPAB243.pdf.

A. Narayanan et al. Machine Learning for Slow Spill Regulation in the Fermilab Delivery Ring for Mu2e. JACoW Publishing, Geneva, Switzerland, 2022. URL https://napac2022.vrws.de/papers/mopa75.pdf. presented at NAPAC'22, Albuquerque, New Mexico, USA, Aug. 2022, paper MOPA28, unpublished.

Aakaash Narayanan, KJ Hazelwood, Michelle Ibrahim, Han Liu, Seda Memik, Vladimir Nagaslaev, Dennis Nicklaus, Peter Prieto, Brian Schupbach, Kiyomi Seiya, et al. Optimizing mu2e spill regulation system algorithms. Technical report, Fermi National Accelerator Lab.(FNAL), Batavia, IL (United States), 2021b.

Fabian Paischer, Thomas Adler, Vihang Patil, Angela Bitto-Nemling, Markus Holzleitner, Sebastian Lehner, Hamid Eghbal-Zadeh, and Sepp Hochreiter. History compression via language models in reinforcement learning. In *International Conference on Machine Learning*, pages 17156–17185. PMLR, 2022.

Antonin Raffin, Ashley Hill, Adam Gleave, Anssi Kanervisto, Maximilian Ernestus, and Noah Dormann. Stable-baselines3: Reliable reinforcement learning implementations. *The Journal of Machine Learning Research*, 22(1):12348–12355, 2021.

Hubert Ramsauer, Bernhard Schäfl, Johannes Lehner, Philipp Seidl, Michael Widrich, Thomas Adler, Lukas Gruber, Markus Holzleitner, Milena Pavlović, Geir Kjetil Sandve, et al. Hopfield networks is all you need. *arXiv preprint arXiv:2008.02217*, 2020.

Alex Reneau, Jerry Yao-Chieh Hu, Chenwei Xu, Weijian Li, Ammar Gilani, and Han Liu. Feature programming for multivariate time series prediction, 2023.

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

K. Seiya, K. J. Hazelwood, M. A. Ibrahim, V. P. Nagaslaev, D. J. Nicklaus, B. A. Schupbach, R. M. Thurman-Keup, N. V. Tran, H. Liu, and S. Memik. Accelerator real-time edge ai for distributed systems (reads) proposal, 2021a.

K. Seiya, K. J. Hazelwood, M. A. Ibrahim, V. P. Nagaslaev, D. J. Nicklaus, B. A. Schupbach, R. M. Thurman-Keup, N. V. Tran, H. Liu, and S. Memik. Accelerator real-time edge ai for distributed systems (reads) proposal, 2021b. URL https://arxiv.org/abs/2103.03928.

R Shi, S Ogrenci, JM Arnold, JR Berlioz, P Hanlet, KJ Hazelwood, MA Ibrahim, H Liu, VP Nagaslaev, A Narayanan, et al. Ml-based real-time control at the edge: An approach using hls4ml. *arXiv preprint arXiv:2311.05716*, 2023.

M. Thieme. Machine learning for slow spill regulation in the fermilab delivery ring, 11 2022. URL https://indico.bnl.gov/event/16158/contributions/69563/attachments/44212/74590/ICFA%20SRS%20Presentation.pdf. presented at third ICFA Beam Dynamics Mini-Workshop on Machine Learning Applications for Particle Accelerators, Chicago, Illinois, USA, Nov. 2022.

Mattson Thieme, Jeremy Arnold, Mark Austin, Pierrick Hanlet, Kyle Hazelwood, Michelle Ibrahim, Han Liu, Seda Memik, Vladimir Nagaslaev, Aakaash Narayanan, et al. Semantic regression for disentangling beam losses in the fermilab main injector and recycler. Technical report, Fermi National Accelerator Lab.(FNAL), Batavia, IL (United States), 2022.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*, 2021.

Shuangfei Zhai, Tatiana Likhomanenko, Etai Littwin, Dan Busbridge, Jason Ramapuram, Yizhe Zhang, Jiatao Gu, and Joshua M Susskind. Stabilizing transformer training by preventing attention entropy collapse. In *International Conference on Machine Learning*, pages 40770–40803. PMLR, 2023.

Zhihan Zhou, Yanrong Ji, Weijian Li, Pratik Dutta, Ramana Davuluri, and Han Liu. Dnabert-2: Efficient foundation model and benchmark for multi-species genome. *arXiv preprint arXiv:2306.15006*, 2023.

Ali Zribi, Mohamed Chtourou, and Mohamed Djemel. A new pid neural network controller design for nonlinear processes. *Journal of Circuits, Systems and Computers*, 27(04):1850065, 2018.

A Future Directions

We outline two potential directions for future investigations toward a *model-free* RL controller for the Mu2e experiments.

Pretrained Transformers. A main motivation for our method is the sequential modeling of noisy spill rate hard to converge for naive RL algorithms. This challenge persists even with the simplified simulator environment [Narayanan et al., 2021b]. While our approach incorporates model-based inductive bias (such as PID), another popular method in the literature involves using transformers [Vaswani et al., 2017] for generative trajectory modeling in RL [Paischer et al., 2022, Janner et al., 2021, Chen et al., 2021]. This approach is beneficial for our problem: it scales up easily using existing transformer-based language and vision models like BERT [Zhou et al., 2023, Ji et al., 2021, Devlin et al., 2018] and GPT [Wei et al., 2021], enables stable training [Zhai et al., 2023], and addresses the challenges of short-sighted behavior and long-term credit assignment in RL [Chen et al., 2021].

However, a major issue with training transformers solely on observed samples from the environment is their tendency to overfit, especially given the typically limited data generated by the current policy [Chen et al., 2021]. To circumvent this, we propose using a Transformer pre-trained on extensive observations/samples without any fine-tuning or weight adjustments [Paischer et al., 2022]. Modern Hopfield networks [Hu et al., 2023, Paischer et al., 2022, Ramsauer et al., 2020] provide a natural approach for utilizing frozen transformers via their associative memory interpretation. We notice that there are works in (i) employing "frozen" modern Hopfield networks with pre-trained transformers for RL [Paischer et al., 2022], and (ii) addressing noisy sequence modeling using sparse modern Hopfield models [Hu et al., 2023]. We would like to explore the integration of (i) & (ii) in the future.

Feature Programming. A key innovation in our approach is the integration of *model-based* inductive bias from PID controllers into our policy network's architecture. This involves manually crafting features (PID parameters) from sequences of spills. However, this method risks missing other crucial features and may introduce instability in the training process, as evidenced in Table 1. To more effectively identify and leverage informative features for RL model training, we propose employing Feature Programming [Reneau et al., 2023], an automated feature engineering framework for noisy multivariate time series. This framework will be used to (i) generate informative features from noisy spill sequences and (ii) incorporate model-based inductive bias from PID controllers. In future work, we aim to combine both (i) and (ii) to develop a *model-free* RL controller for the Mu2e experiments.