

# ADVANCED ML METHODS FOR BEAM TUNING AT FRIB\*

K. Hwang<sup>†</sup>, Jinyu Wan, Tong Zhang, Tomofumi Maruta, Qiang Zhao, Alexander Plastun, Kei Fukushima, Peter Ostroumov, Facility for Rare Isotope Beams, East Lansing, MI, USA

## Abstract

Experiments with rare isotope beams at FRIB are highly time-constrained, making rapid setup and delivery of high-quality ion beams critical to maximizing scientific output. The Bayesian framework is particularly well-suited for this challenge, offering sample-efficient optimization, principled incorporation of prior knowledge, and uncertainty-aware inference. In particular, Bayesian Optimization (BO) has proven to be an efficient and general approach for the non-sequential, static nature of beam-tuning tasks. To further accelerate convergence, Prior-Mean-Assisted Bayesian Optimization (pmBO) was developed, enabling rapid adaptation from prior belief to real-time machine conditions with minimal computational overhead. In parallel, a virtual diagnostic for the beam's transverse quadrupolar moment (BPM-Q) has been developed to provide non-invasive, fast measurements of beam envelope information. To optimize the reconstruction of Courant-Snyder parameters from BPM-Q data, Bayesian Active Learning (BAL), employing a differentiable beam envelope simulator as a surrogate model, has been implemented. Together, these developments illustrate the power of Bayesian methods in achieving faster, more accurate beam-tuning.

## INTRODUCTION

Experiments at Facility for Rare Isotope Beams (FRIB) require frequent switching of diverse primary ion beams, imposing strict time constraints that necessitate rapid setup and delivery of high-quality beams to maximize scientific productivity. Beam tuning within a limited time is a complex task. The high dimensionality, coupled with nonlinear and noisy system responses, renders traditional tuning methods inefficient, particularly under tight time constraints. Advanced machine learning (ML) techniques offer a powerful solution.

This paper presents several advanced ML methods developed or adapted for beam tuning at FRIB. We begin by explaining why the Bayesian Framework, including Bayesian Optimization (BO), is particularly well-suited for this task. We then discuss enhancements to BO. To address high-dimensional parameter spaces, we introduce the Surrogate-gradient Assisted optimizer. Additionally, we present a virtual diagnostic (VD) for the beam's transverse quadrupolar moment (BPM-Q) and demonstrate how Bayesian Active Learning (BAL) optimizes the reconstruction of Courant-Snyder parameters despite the VD inaccuracy.

\* Work supported by the U.S. Department of Energy, Office of Science, Office of Nuclear Physics, under Award Number DE-SC0024707 and used resources of the FRIB Operations, which is a DOE Office of Science User Facility under Award Number DE-SC0023633

<sup>†</sup> hwang@frib.msu.edu

## BO FOR BEAM TUNING

### Why BO is Well-Suited

BO leverages a probabilistic model, such as the Gaussian Process (GP), to explicitly assess uncertainty quantification (UQ) and related metrics. This approach offers several advantages for beam tuning.

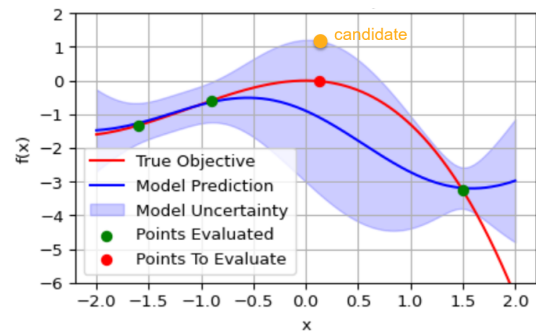


Figure 1: Illustration of sample-efficient optimal search for an objective  $f(x)$ , over  $x$  control parameters considering model uncertainty towards the optimum [1]. Green dots: training data, Yellow dot: the candidate queried BO.

**Sample Efficiency** Probabilistic modeling enables a principled balance between exploration and exploitation for active learning of the objective toward optimal control (Fig. 1). This allows BO to efficiently converge using as few data samples as possible. Such sample efficiency is especially beneficial in beam tuning tasks, where each evaluation involves adjusting magnet strengths—requiring power supply ramping over a few seconds—followed by measurement averaging over a few seconds to reduce noise. These time-consuming steps limit the number of feasible evaluations, making efficient data use critical. Furthermore, the surrogate model training in BO typically takes a few seconds allowing it to run in parallel with the objective evaluation process.

**Noise Robustness** Beam parameter measurements are inherently noisy. While temporal averaging reduces variance, residual noise persists. Probabilistic modeling in BO naturally incorporates this noise, typically by adding a learnable noise term to the GP kernel function, which corresponds to a Gaussian likelihood with additive noise.

**Comparison with RL** Unlike Reinforcement Learning (RL), which is designed for sequential, dynamic tasks (e.g., robotics) where a Markov Decision Process is assumed, BO is designed for static optimization problems that are common in beam development tuning. BO can also naturally incorporate prior models trained on offline data, such as those learned from simulation data, similar to approaches in

pre-trained RL [2,3]. This enables near-instant optimization when the offline data is well aligned with the online data of the real machine. Furthermore, BO is inherently equipped to sample-efficiently handle distribution shifts from simulation or archived offline data to real machine online data.

### Customization of BO for Beam Tuning.

We customized BO to meet our needs in focus on minimizing beam tuning time. In this section, we review our previous work [1].

**Asynchronous BO** To minimize machine idle time during BO, we adopt an asynchronous evaluation. While BO performs computations to select the next candidate, the machine concurrently evaluates the current candidate control setting (Fig. 2). Since the outcome of the current evaluation is not yet available, the selection of the next candidate must avoid the point currently being evaluated.

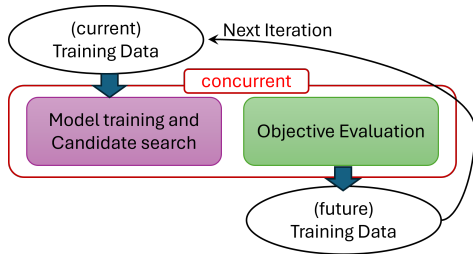


Figure 2: Flow chart for asynchronous BO.

Our approach follows a penalization strategy similar to Ref. [4], where the acquisition function is modified to discourage selection near the currently evaluating point.

An alternative technique is to exploit *fantasization* [5], in which the unknown objective value at the currently evaluating point is temporarily “fantasized” using the model’s predictive distribution.

Additionally, we designed the candidate selection process to terminate as soon as the machine completes its current evaluation, ensuring that new candidates can be evaluated without delay.

**Cost-Aware BO** To mitigate time-cost associated with power supply ramping, we augment the acquisition function opposite way of the penalization and polarity change suppression. This is critical, as polarity changes in the FRIB corrector magnet power supply incur a delay over 15 seconds.

Additionally, when evaluating batches of candidate points—such as during the initialization phase with random candidates—the evaluation sequence is optimized to minimize ramping costs as depicted in Fig. 3.

**Objective Function Regularization** Constructing a smooth and well-behaved objective function is just as important as developing a sophisticated optimizer. We design the objective such that the optimal value is normalized to 1 and apply a bilog transformation, defined as

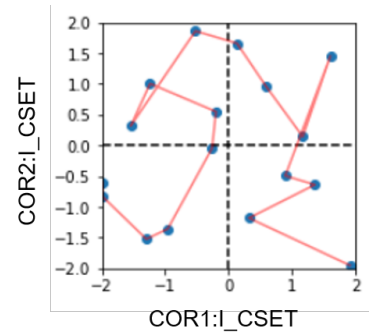


Figure 3: Optimized evaluation path for corrector settings, designed to minimize ramping time costs [1].

$y \rightarrow \text{sign}(y) \log(1 + y)$ , to regularize the function and better resolve the region near the optimum.

Figure 4 illustrates the effect of the bilog transformation [6]. Note that in the original objective function, large variations away from the optimum can obscure the relatively small variations near the peak. This increases the number of samples needed to resolve the optimum region accurately. The bilog transformation mitigates this imbalance by compressing the scale of large values while preserving detail near the optimum.

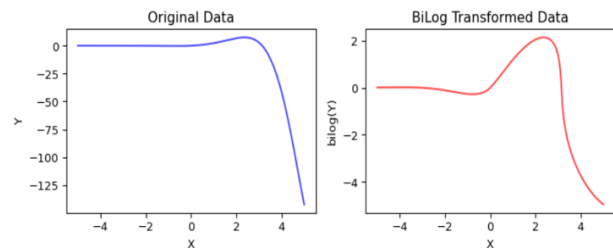


Figure 4: Illustration of bilog transformation.

**Local Optimization** While BO offers superior sample efficiency, performing a global search in control spaces of moderate to high dimensionality ( $D > 5$ ) under strict time constraints can be impractical. To address this, we employ localmBO in which, the search is confined to a hypercube centered on the current best-found point. At each iteration, this hypercube is re-centered on the updated optimum as shown in Fig. 5. This localization confers three primary benefits:

1. **Enhanced model accuracy:** By restricting the search to a smaller local region, we effectively increase data density, enabling more accurate surrogate modeling within the relevant subspace.
2. **Hysteresis mitigation:** Smaller control-knob adjustments limit magnet-power-supply excursions reducing magnetic hysteresis errors.
3. **Drift minimization:** Accelerating convergence shortens the overall tuning duration, mitigating the effects

of machine drifts and eliminating the need for explicit time-context aware modeling.

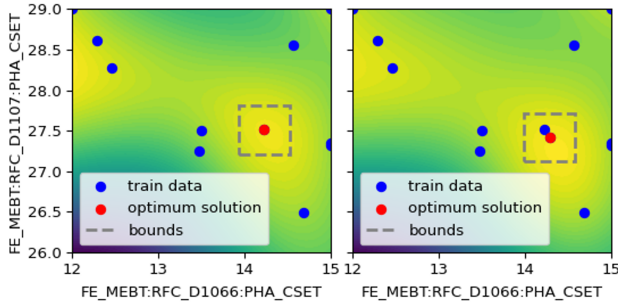


Figure 5: Evolution of the local BO search domain over the model's predicted-objective contours [1]. *Left*: transition from global to local search, centered on the current best point. *Right*: re-centered local search region in the subsequent iteration. Brighter colors indicate higher predicted objective values.

**Prior Mean Assisted BO** When an offline model, trained on historical data or through beam dynamics simulations, is available, it can be integrated into BO in various ways. Among them, incorporation of the offline model into prior mean of the GP is conceptually simple, computationally lightweight, and requires no modification to the standard GP framework beyond specifying the prior mean. Prior mean assisted BO (pmBO) is also robust to the distribution shift between the prior mean model and real machine as the online data collected during optimization can be used to update the model sample efficiently [3, 7, 8] as illustrated in Fig 6.

## HIGH-DIMENSIONAL ONLINE BEAM TUNING

High-dimensional beam tuning presents significant challenges due to the curse of dimensionality. However, machine learning models—often involving millions of parameters—are routinely trained. Beyond advancements in GPU computing, a key enabler of this progress is auto-differentiation (AD) for gradient descent optimization, which scales gracefully with the dimensionality of the parameter space.

However, this is not directly applicable to real-world beam tuning, where gradient information is not available. To address this, we propose a hybrid approach that combines Extremum Seeking (ES) [9, 10], Surrogate Gradient (SG) modeling and the Adam optimizer [11].

### Extremum Seeking (ES)

ES is powerful for high-dimensional control parameter spaces without requiring explicit gradient information. It introduces sinusoidal perturbations (dithers) to the control

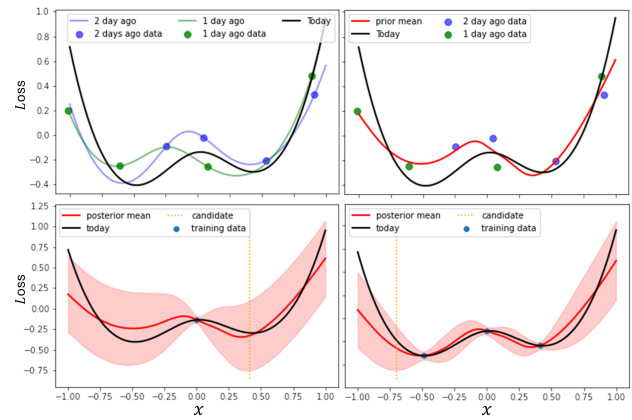


Figure 6: Conceptual illustration of pmBO under machine drift [12]. **Top-left**: A hypothetical 1D loss function (to be minimized) across three days, illustrating daily machine variation. Colored lines show loss functions for different days; dots represent archived data used for offline model training. **Top-right**: Prior mean model (e.g., neural network) trained on archived data. **Bottom-left**: GP posterior after one online evaluation on the current machine. **Bottom-right**: GP posterior after two iterations of BO using the prior mean and updated observations. Orange vertical line corresponds to the candidate solution queried by BO.

parameters, effectively performing gradient descent (or ascent for maximization) on average, regardless of the dimensionality. The dynamics of control parameters by ES for maximizing an objective function  $f(x; t)$  are described by:

$$\frac{dx}{dt} = \sqrt{2\alpha\omega} \cos(\omega t - kf(x; t)) \xrightarrow{\omega \rightarrow 0} \frac{k\alpha}{2} \frac{\partial f(x; t)}{\partial x} \quad (1)$$

Here,  $\alpha$  is the dithering amplitude,  $\omega$  is the dithering frequency, and  $k$  is the ES gain. This mechanism mimics numerical AD and gradient descent in deep learning, enabling ES to scale gracefully with dimensionality. For example, in a 100-dimensional optimization problem, the traditional Nelder-Mead algorithm requires 100 iterations just to construct the initial simplex. In contrast, ES can start optimizing within the first few iterations. Additionally, ES is robust to noise and machine drift over time.

### Surrogate Gradient (SG) Assisted Optimization

We incorporate a SG approach using a surrogate model trained on online data collected during the ES run. Unlike surrogate modeling of the objective function across the entire high-dimensional space—which demands large datasets—estimating the gradient at a single point (i.e., the current setpoint) is feasible with much fewer data points. We found that Gaussian Processes (GP) outperform Neural Networks (NN) in both gradient accuracy and computation time, primarily due to the sparse data typical in high-dimensional settings. The resulting SG step,  $\Delta x^{\text{SG}} = \mu_g$  where  $\mu_g$  is the gradient of the GP's predictive mean at the setpoint, is then

integrated with the ES update as:

$$x \leftarrow x + \Delta x^{\text{SG}} \cdot \text{lr} + \Delta x^{\text{ES}} \cdot \text{lr}^{\text{ES}}$$

where  $\text{lr}$  is the learning rate for the SG, and we choose  $\text{lr}^{\text{ES}} = 1$ . This hybrid approach leverages ES's exploration to collect data and SG's exploitation to optimize, significantly improving convergence speed. The combination of ES and SG is critical, as SG alone lack reliability due to limited data. ES's perturbative exploration ensures robust gradient estimation through exploration, while SG accelerates convergence.

### Adaptive Momentum with Adam Optimizer

To further enhance convergence speed, we incorporate the Adam optimizer that is commonly used in deep learning [11]. The Adam optimizer accumulate momentum gain of the gradient. This momentum not only accelerates progress toward the optimum but can also help escape small local optima. But it can also overshoot the objective past the optimum crest. In our implementation, we assume the optimal objective value to be 1 and mitigate overshooting by decaying the learning rate based on the objective function value, ensuring stability near the optimum.

### Uncertainty of SG

Although we estimate highly localized gradients around the current setpoint, the SG can still be unreliable due to high-dimensionality. The uncertainty in the SG—quantified as the covariance matrix of the gradient—is given by

$$\Sigma_g = \nabla K(x, x) \nabla^T - \nabla K(x, X) (K(X, X) + \sigma^2 \mathbf{I})^{-1} K(X, x) \nabla^T \quad (2)$$

where  $x$  is the evaluation point,  $X$  is the training inputs,  $\sigma$  is the learned noise parameter, and  $K(\cdot, \cdot)$  is the GP kernel. The gradient operator  $\nabla$  acts on the first argument of  $K$  when placed on the left, and on the second argument when placed on the right.

We suppress the uncertain gradient directions by multiplying the inverse of the uncertainty  $\Sigma_g$  on the gradient of the GP mean  $\mu_g$  as

$$v^* = \Sigma_g^{-1} \mu_g. \quad (3)$$

It is shown that  $v^*$  is also the direction of the most probable ascent, and the probability of ascent is [13]

$$P_g = \Phi \left( \sqrt{\mu_g^T \Sigma_g^{-1} \mu_g} \right) \quad (4)$$

where  $\Phi(\cdot)$  is the cumulative distribution function of the normal distribution. In our implementation, we modify  $\Delta x^{\text{SG}}$  by

$$\Delta x^{\text{SG}} = P_g \left( \mu_g \cdot \frac{v^*}{\|v^*\|_2} \right) \frac{v^*}{\|v^*\|_2} \quad (5)$$

Here  $\|\cdot\|_2$  represents L2-norm of a vector. This update can be interpreted as projecting the mean gradient onto the unit vector of the most probable ascent direction, and then scaling it by the probability of improvement in that direction.

### Benchmark Studies

We conducted benchmark studies comparing standard ES, ES combined with SG (ES+SG), SG without ES, ES with Adam on SG (ES+AdamSG), AdamSG without ES, ES+SG with awareness of the uncertainty of SG (ES+uSG), and ES+uAdamSG. The result is illustrated in Fig. 7. The benchmark was performed on a synthetic objective function defined by a randomly initialized neural network over a 50-dimensional input space. For all cases, ES was run for up to 50 epochs. After collecting 50 data points through ES, different optimizers were applied, each demonstrating distinct performance. Significant improvement from ES is noticeable. SG and AdamSG without ES suffer from wrong gradient estimation due to the lack of exploration, which is handled by ES. We are conducting more robust and statistical benchmark on various test functions with random initialization.

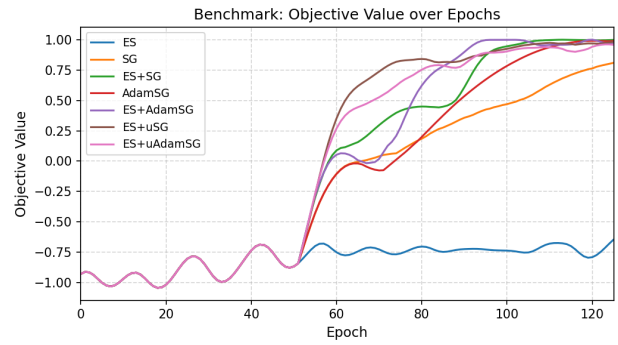


Figure 7: Comparison of optimization performance.

## BPMQ VIRTUAL DIAGNOSTICS

In this section, we briefly review Ref. [14].

### BPMQ Modeling

Beam Position Monitor (BPM) with a circular cross-section and four pickups positioned at the top, left, right, and bottom can measure the beam quadrupole moment (BPMQ)  $BPMQ = \sigma_x^2 - \sigma_y^2$  by [15]:

$$BPMQ = G \frac{U_R + U_L - (U_T + U_B)}{U_R + U_L + U_T + U_B} - (x^2 - y^2) \approx \sigma_x^2 - \sigma_y^2 \quad (6)$$

where  $G$  is a geometric factor that depends on the BPM geometry,  $U_T$ ,  $U_L$ ,  $U_R$  and  $U_B$  are the induced signal at the top, left, right, and bottom pickups, and  $x$  and  $y$  represent the horizontal and vertical beam centroids which are also a function of the 4 pickup signals. The BPMQ signal is significantly weaker compared to the beam centroid signal. Consequently, small errors, such as calibration inaccuracies, can lead to substantial errors in BPMQ measurements.

To address these issues, we utilize Virtual Diagnostics (VD) by supervised learning with NN and GP as in Fig 8.

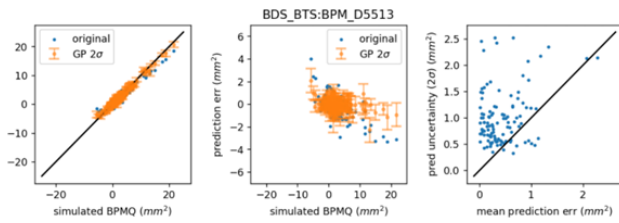


Figure 8: GP prediction of BPMQ of a BPM on validation data. Original represent the calculated BPMQ using the formula Eq. (6). Middle: prediction error with error bar. Right: prediction error versus model uncertainty. It shows that model uncertainty generally is higher than prediction error as intended [14].

### Bayesian Active Learning (BAL)

Although the BPMQ models were carefully trained, their accuracy is limited by the quality of the training data labels. We also observed that even small BPMQ errors can lead to significant errors in the reconstructed Courant–Snyder (CS) parameters. Overfitting CS parameters to BPMQ errors can be mitigated by acquiring a large volume of BPMQ VD data through scans of quadrupole magnets (q-scan). Accurate reconstruction of CS parameters from beam quadrupole moments at a limited number of BPMs requires an efficient q-scan that maximizes information gain. To achieve this, we implement BAL for optimal q-scan design.

We use an ensemble of beam envelope simulations implemented in PyTorch as a surrogate model [16]. The simulation is vectorized [17, 18] to simultaneously process many CS parameter sets at once to boost computational speed. Each CS parameter set and its corresponding simulated beam envelope represent a sample from a probabilistic distribution. Unlike black-box probabilistic models like GP, this approach directly incorporates physics constraints, making it highly sample-efficient and interpretable enabling the implementation of various physics-informed considerations. For example, when the quadrupole magnet setting is queried we can impose penalty on large beam size at quadrupole magnet locations to avoid beam loss.

In each iteration of BAL, the ensemble of CS parameters is updated based on VD BPMQ data. A new quadrupole magnet setting is then queried by selecting the one that maximizes the variation in the predicted BPMQ over ensemble model. It can be shown that maximum variation in the predicted BPMQ corresponds to the maximum information gain under variational approximation approach [19].

### Beam Test

Recent beam tests have shown promising results. To reduce computation time, the surrogate model for BAL was initialized after conducting 8 random quadrupole scans (q-scans). In each q-scan, BPMQ values were recorded from 16 BPMs using BPMQ VD. VD predicted uncertainties over all measurement are also leveraged when selecting q-scans to query.

Using BAL, we queried 2 additional q-scans for BPMQ measurements and 1 additional q-scan for a Profile Monitor (PM) measurement.

Figure 9 shows the final CS solutions, where all ensemble members have converged to a unique solution. The reference solution was obtained using a separate q-scan procedure with PM measurements, which is the standard method traditionally used for CS reconstruction.

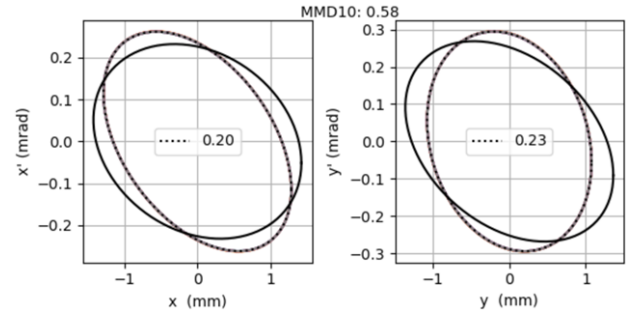


Figure 9: Final ensemble of BPMQ based CS solution. The black ellipse correspond to the reference solution. MMD:Maximum Mean Discrepancy for 2D mismatch quantification [20]

We also performed transverse envelope matching downstream of the beamline using the BPMQ-based CS solution. It resulted in significantly lower readings from the beam loss monitors compared to the reference that is envelope matching based on the PM-derived CS solution, as shown in Fig. 10.

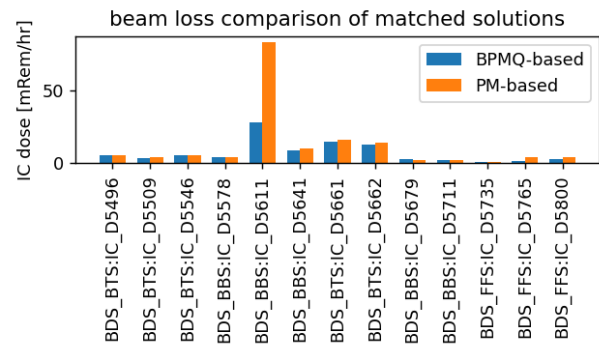


Figure 10: Comparison of Ion Chamber readings for beam loss detection using two optics solutions.

## CONCLUSION

We presented a suite of advanced ML methods for rapid, robust beam tuning at FRIB, including pmBO, SG-assisted optimization combined with ES and Adam, and BAL for BPMQ based CS reconstruction. Simulation and beam tests demonstrate faster convergence and improved robustness: the BPM-Q ensemble converged to a unique CS solution and produced reduced beam-loss monitor readings compared to the standard procedure.

## REFERENCES

- [1] K. Hwang *et al.*, “Machine-learning-assisted beam tuning at FRIB”, in *Proc. LINAC’24*, Chicago, IL, USA, Aug. 2024, pp. 573–576. doi:10.18429/JACoW-LINAC2024-THXA004
- [2] J. Kaiser *et al.*, “Reinforcement learning-trained optimisers and Bayesian optimisation for online particle accelerator tuning”, *Sci. Rep.*, vol. 14, p. 15733, 2024. doi:10.1038/s41598-024-66263-y
- [3] K. Hwang *et al.*, “Prior-mean-assisted Bayesian optimization application on FRIB Front-End tuning”, *arXiv*, 2022. doi:10.48550/arXiv.2211.06400
- [4] Javier Gonzalez, Zhenwen Dai, Philipp Hennig and Neil Lawrence, “Batch Bayesian Optimization via Local Penalization”, in *Proc. 19th Int. Conf. on Artificial Intelligence and Statistics*, vol. 51, pp. 648–657, May 2016.
- [5] D. Ginsbourger, R. Le Riche, and L. Carraro, “Kriging is well-suited to parallelize optimization”, *Comput. Intell. Expens. Optim. Probl. Adapt. Learn. Optim.*, vol. 2. Berlin, Heidelberg: Springer, 2010, pp. 131–162. doi:10.1007/978-3-642-10701-6\_6
- [6] D. Eriksson, M. Pearce, J. R. Gardner, R. D. Turner, and M. Poloczek, “Scalable Global Optimization via Local Bayesian Optimization”, in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, Vancouver, Canada, Dec. 2019.
- [7] K. Hwang *et al.*, “Beam Tuning at the FRIB Front End Using Machine Learning”, in *Proc. IPAC’22*, Bangkok, Thailand, Jun. 2022, pp. 983–986. doi:10.18429/JACoW-IPAC2022-TUPOST053
- [8] T. Boltz *et al.*, “Leveraging prior mean models for faster Bayesian optimization of particle accelerators”, *Sci. Rep.*, vol. 15, p. 12232, 2025. doi:10.1038/s41598-025-95297-z
- [9] A. Scheinker and D. Scheinker, “Bounded extremum seeking with discontinuous dithers”, *Automatica*, vol. 69, pp. 250–257, 2016. doi:10.1016/j.automatica.2016.02.023
- [10] A. Scheinker, E. Huang, and C. Taylor, “Extremum Seeking-Based Control System for Particle Accelerator Beam Loss Minimization”, *IEEE Trans. Control Syst. Technol.*, vol. 30, no. 5, 2022. doi:10.1109/TCST.2021.3136133
- [11] D.P. Kingma and J. Ba, “Adam: A Method for Stochastic Optimization”, *arXiv*, 2014. doi:10.48550/arXiv.1412.6980
- [12] P. Ostroumov, K. Hwang, A. Sheinker *et al.*, “Online Autonomous Tuning of the FRIB Accelerator Using Machine Learning”, project proposal narrative, 2023, unpublished.
- [13] Q. Nguyen, K. Wu, J. R. Gardner, and R. Garnett, “Local Bayesian optimization via maximizing probability of descent”, *Adv. Neural Inf. Process. Syst.*, vol. 35, 2022, pp. 13190–13202.
- [14] K. Hwang *et al.*, “Application of ML tools for extraction of BPM-Q and transverse beam matching”, in *Proc. HIAT2025*, Michigan State University, East Lansing, USA, Jun. 2025, to be published. doi:10.18429/JACoW-HIAT2025-TUP11
- [15] B. E. Carlsten and S. J. Russell, “Measuring Emittance Using Beam Position Monitors”, in *Proc. PAC’93*, Washington D.C., USA, Mar. 1993, pp. 2537–2540.
- [16] A. Paszke *et al.*, “Automatic differentiation in PyTorch”, presented at NIPS’17, Long Beach, CA, USA, Dec. 2017. <https://openreview.net/forum?id=BJJsrnfCZ>
- [17] Wang, B., Rosales-Fernandez, C., Tang, W., “Improve Performance by Vectorizing Particle-in-Cell Codes: A Guide”, Intel, 2019. <https://www.intel.com/content/www/us/en/developer/articles/technical/improve-performance-vectorizing-particle-in-cell.html>
- [18] J.-L. Vay *et al.*, “Novel methods in the Particle-In-Cell accelerator Code-Framework Warp,” *Comput. Sci. Discov.*, vol. 5, no. 1, 2012.
- [19] K. Hwang, “Bayesian Active Learning for Converging Posteriors in Latent Variable Inference for Control Systems,” to be presented at Advanced AI/ML and Generative Models for the Control of Large Complex Systems Workshop of ICALEPCS’25, Chicago, IL, USA, Sep. 2025.
- [20] C. E. Mitchell, R. D. Ryne, K. Hwang, “Using kernel-based statistical distance to study the dynamics of charged particle beams in particle-based simulation codes”, *Phys. Rev. E*, 106, 065302, 2022.