ISBN: 978-3-95450-248-6 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2025-THPM116

EFFICIENT DATA-DRIVEN MODEL PREDICTIVE CONTROL FOR ONLINE ACCELERATOR TUNING

C. Xu*,[1], J. Kaiser[2], A. Santamaria Garcia[3], A. Eichler[2,4],
C. Hespe[2], B. Rodriguez Mateos[5], S. Hirlaender†,[6]
[1]Karlsruhe Institute of Technology, Karlsruhe, Germany
[2]Deutsches Elektronen-Synchrotron DESY, Hamburg, Germany
[3]University of Liverpool, Liverpool, United Kingdom
[4]Hamburg University of Technology, Hamburg, Germany
[5]CERN, Geneva, Switzerland
[6]University of Salzburg, Salzburg, Austria

Abstract

Reinforcement learning (RL) is a promising approach for the online control of complex, real-world systems, with recent success demonstrated in applications such as particle accelerator control. However, model-free RL algorithms often suffer from sample inefficiency, making training infeasible without access to high-fidelity simulations or extensive measurement data. This limitation poses a significant challenge for efficient real-world deployment. In this work, we explore data-driven model-predictive control (MPC) as a solution. Specifically, we employ Gaussian processes (GPs) to model the unknown transition functions in the real-world system, enabling safe exploration in the training process. We apply the GP-MPC framework to the transverse beam tuning task at the ARES accelerator, demonstrating its potential for efficient online training. This study showcases the feasibility of data-driven control strategies for accelerator applications, paving the way for more efficient and effective solutions in real-world scenarios.

INTRODUCTION

Modern particle accelerators are complex, high-dimensional systems that require precise, continuous tuning to maintain optimal performance. Traditional tuning approaches often rely on expert knowledge and manual adjustments, which can be time-consuming and suboptimal. As accelerators become increasingly sophisticated, there is a growing demand for intelligent control and tuning algorithms that can automate this process efficiently.

Reinforcement learning (RL) has emerged as a promising tool in this context, offering the ability to learn effective control strategies directly from data. Recent studies show that RL optimizers, once pre-trained in simulation models, can efficiently control the accelerator tasks [1]. However, a key limitation of standard RL methods is their poor sample efficiency. Typically, thousands to millions of interactions with the system are required, posing a significant challenge in accelerator applications, where experiment is costly and time is limited. Model-based reinforcement learning (MBRL) addresses this issue by learning a predictive model of the system dynamics and using it to simulate future outcomes, thus reducing the need for direct interaction [2]. One successful modeling approach within MBRL is the use of Gaussian processs (GPs) [3], which not only provide accurate predictions but also uncertainty estimates. In recent years, Bayesian optimization (BO) algorithms based on GP models have already demonstrated considerable success in accelerator tuning tasks due to their ability to make informed decisions under uncertainty [4]. Building on this foundation, the GP-model predictive control (MPC) [5, 6] offers a powerful new approach that combines GP for the system modeling and MPC for action selection. By explicitly incorporating model uncertainty into a predictive control scheme, GP-MPC enables sample-efficient, robust decision-making tailored to the nonlinear and unknown dynamics of particle accelerators.

This paper presents the application of GP-MPC in the context of accelerator control, illustrating its potential to significantly advance the state of automated tuning and control in accelerator operations. The GP-MPC algorithm has previously been successfully applied at the AWAKE accelerator for beam trajectory tuning with corrector magnets [7], as well as at SIS-18 for a beam injection task [8]. In this contribution, we extend the GP-MPC method to a nonlinear transverse beam tuning task at the ARES accelerator. We demonstrate that this method can be deployed turn-key and allows sample-efficient online tuning. Here, we adopt the formulation as described in Ref. [6] and outline the algorithm in Algorithm 1. It is worth noting that although GP model training and action planning are performed sequentially in our implementation, these steps can be executed asynchronously to accommodate the demands of real-time control systems. A more comprehensive overview of the GP-MPC algorithms can be found in the review articles [9, 10].

THE ARES EXPERIMENTAL AREA TUNING TASK

The Accelerator Research Experiment at SINBAD (ARES) accelerator [11] at DESY is a test facility for sub-femtosecond electron bunches studies and accelerator component research. This paper focuses on the lattice section

* chenran.xu@kit.edu
† simon.hilaender@plut.at.edu

ISBN: 978-3-95450-248-6 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2025-THPM116

Algorithm 1 GP-MPC control loop

Define the prior for the GP models.
Interact with the environment on n_0 initial points to get initial data set of transitions $D_0 = \{((s,a) \times (s'))_i, i = 1, \ldots, n_0\}$.
for $t = 1, 2, \ldots$ **do** ▷ Control loop
    Build GP models $GP : (s,a) \longrightarrow s'$ using available data D_{t-1}.
    Optimize the GP models hyperparameters.
    Predict the system dynamics with GP-models
    Find the action sequence that minimizes the cost between predicted and the target states (s^{Target})
    Apply the first action a_t, observe new transition $(s_t, a_t) \longrightarrow (s_{t+1})$.
    Augment the dataset.
end for

known as the Experimental Area (EA), which is responsible for tuning the beam parameters to meet the specific requirements of the downstream experimental chamber.

The EA section consists of three quadrupole magnets $Q_{1,2,3}$, one vertical C_v, and one horizontal steering magnet C_h. The magnets are arranged in the order of $[Q_1, Q_2, C_v, Q_3, C_h]$. The section has a diagnostic screen downstream of the magnets, where the electron beam can be imaged. From the screen image, the statistical parameters of the beam $\boldsymbol{b} = (\mu_x, \sigma_x, \mu_y, \sigma_y)$ can be determined, where μ denotes the beam size, σ denotes the one standard deviation beam size, and (x, y) denote the horizontal and vertical plane respectively. The actuators $\boldsymbol{u}$, i.e. the input parameters controlled by the algorithm, are the normalized quadrupole strengths and the angles of the steerers $\boldsymbol{u} = (k_{Q_1}, k_{Q_2}, \alpha_{C_v}, k_{Q_3}, \alpha_{C_h})$. The ARES-EA tuning task is visualized in Fig. 1. The goal is to achieve a desired target beam $\boldsymbol{b}'$ by iteratively adjusting the magnet settings $\boldsymbol{u}$.

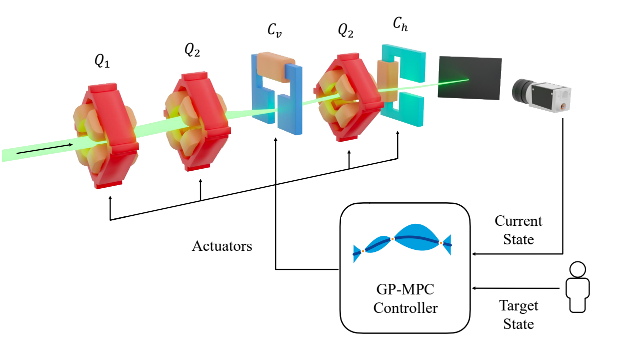

Figure 1: Overview of the EA section beam tuning task. The EA section consists of three quadrupoles, one vertical, and one horizontal steering magnet. Figure adapted from Ref. [1].

The tuning task can be rephrased as minimizing a given objective function

$$\min_{\boldsymbol{u}} O\left(\boldsymbol{u} \mid M, I\right) = \min D\left(\boldsymbol{b}\left(\boldsymbol{u} \mid M, I\right), \boldsymbol{b}'\right), \quad (1)$$

i.e., minimizing the distance metric D between the target beam $\boldsymbol{b}'$ and the observed beam $\boldsymbol{b}$. The task is only partially observable, as the measured beam $\boldsymbol{b}$ not only depends on the actuator settings $\boldsymbol{u}$, but also on variables that can not be easily determined in operation, including the quadrupole misalignments M and the beam parameters I upstream of the EA section. Therefore, the simulation model cannot be directly used in operation to perform model-based optimization or control. The GP-MPC is a promising candidate for such problem as it can efficiently learn the model using data gathered online during operation.

Problem Formulation for GP-MPC

To apply the GP-MPC algorithm, we formulate the EA beam tuning task as follows. The states s are chosen to be the measured beam parameters $\boldsymbol{b}$ and the actions are the direct magnet settings $\boldsymbol{u}$. The transition $(\boldsymbol{b}_t, \boldsymbol{u}_t) \longrightarrow \boldsymbol{b}_{t+1}$ is modeled by independent GPs. To improve the GP modeling stability, the beam parameters are rescaled so that the beam positions between ±2 mm lie within $[0, 1]$, and the magnet settings are normalized to $[0, 1]$ respectively. We use the quadratic cost for the MPC part,

$$\begin{aligned} D((\boldsymbol{b}, \boldsymbol{u}), (\boldsymbol{b}', \boldsymbol{u}')) &= (\boldsymbol{b} - \boldsymbol{b}')^T W^{(b)} (\boldsymbol{b} - \boldsymbol{b}') \\ &\quad + (\boldsymbol{u} - \boldsymbol{u}') W^{(u)} (\boldsymbol{u} - \boldsymbol{u}'), \end{aligned} \quad (2)$$

with $W^{(b)}$ and $W^{(u)}$ containing the weighting factors in the diagonal entries. For the the ARES EA task, we only care about the beam parameters and do not weight the actuators $W^{(u)} = 0$. While here an equal weighting is used $W^{(b)} = \mathbb{1}$, it is possible to further fine-tune the controller's behavior by focusing on specific beam parameters. In the action optimization, we used a horizon $h = 5$, meaning that at each step t, the GP-MPC predicts five steps ahead and minimizes the average of costs throughout the trajectory

$$\min_{\boldsymbol{u}_{t+1}, \ldots, \boldsymbol{u}_{t+5}} \frac{1}{5} \sum_{i=1}^{5} D((\tilde{\boldsymbol{b}}_{t+i}, \boldsymbol{u}_{t+i}), (\boldsymbol{b}', \boldsymbol{u}')), \quad (3)$$

where the future states $\tilde{\boldsymbol{b}}_{t+i}$ are estimated using the GP models. The action changes are limited to 10% of the entire range at each step.

RESULTS

We evaluated the GP-MPC algorithm on a simulated ARES EA model using the *Cheetah* [12] library, averaging over 50 different trials. Each trial is defined by a tuple $(I, M, \boldsymbol{b}')$, i.e., randomized incoming beam, magnet misalignments, and target beam. The GP-MPC was started from scratch, with first three interactions being randomly selected to construct the initial GP models. In addition, the standard BO with a upper confidence bound (UCB) acquisition function and the Nelder-Mead simplex algorithms were evaluated as benchmarks. The actions selected by the BO are also limited to 10% of the total range. Both methods were implemented using the *Xopt* [13] library.

ISBN: 978-3-95450-248-6 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2025-THPM116

The obtained results from each algorithm are quantified using the mean absolute error (MAE) between the observed and the target beam parameters. Figure 2 shows the MAE values averaged over the 50 trials. Both GP-MPC and BO successfully converged within 100 steps, achieving comparable final MAE values. In contrast, the simplex method failed to reach satisfactory settings.

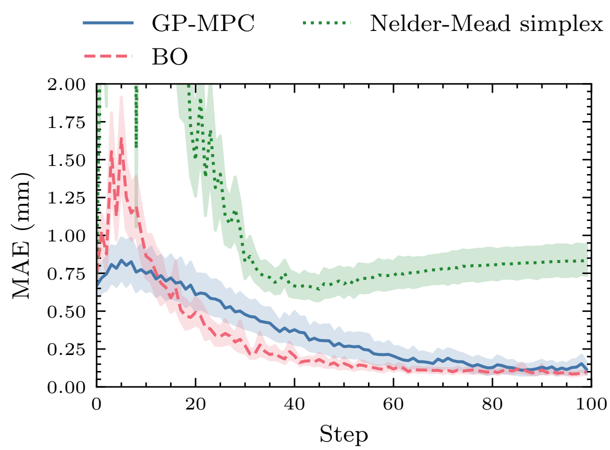

Figure 2: Convergence plot for different algorithms on the beam tuning task. The MAE between the observed and target beam parameters are shown. Each algorithm is evaluated on 50 trials in simulation with different configurations of target beam, incoming beam, and magnet misalignment settings, representing the realistic scenario at the ARES accelerator.

The progress of one sample trial is shown in Fig. 3. The upper plot shows the evolution of the normalized beam parameters, corresponding to a centered and focused beam. Overall, GP-MPC demonstrated a very smooth convergence behavior, thanks to its ability to look ahead and plan the action.

The final evaluation results are listed in Table 1. The beam differences are the median values of the best achieved MAE over the 50 test trials. GP-MPC reached 24 µm, better than BO with 33 µm. Both GP-based methods are clearly outperforming the Nelder-Mead simplex with 276 µm. As GP-MPC is deployed entirely from scratch like BO, the performance is expected to not degrade when transferred to real-world operation [1]. Additionally, we report the cumulative magnet changes, normalized over the allowed action range. Ideally, this value should be low to minimize unnecessary adjustments and reduce stress on real-world power supplies. GP-MPC achieved a total normalized change of 8, representing an 82% reduction compared to BO, and 93% reduction compared to the simplex method. The similar final beam parameters achieved by GP-MPC and BO are expected, as both approaches rely on the same underlying GP modeling technique and the beam tuning task is inherently static. However, the significantly smaller magnet adjustments observed with GP-MPC can be attributed to its ability to plan over a finite horizon. In contrast, BO relies on exploration of the parameter space to locate a global optimum, which often results in more frequent and larger parameter changes.

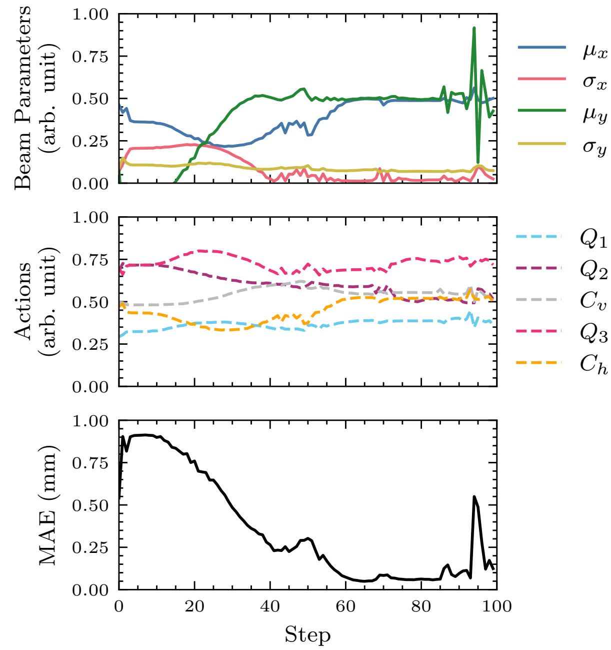

Figure 3: Progress of the GP-MPC algorithm on one example tuning task. The beam parameters and magnet settings are normalized to $[0, 1]$.

Table 1: Results of the beam tuning task. Each algorithm was tested on 50 simulated trials, and the median values of the respective metrics were reported. The beam differences are calculated using MAE. The magnet changes are normalized over the allowed action range.

Algorithm	**Best beam differences (µm)**	**Sum of magnet changes**
GP-MPC	24	8
BO	33	46
Nelder-Mead	276	122

CONCLUSION

This paper presented the application of GP-MPC to a transverse beam tuning task at the ARES experimental area section. The method demonstrates high sample efficiency, making it suitable for direct online deployment in accelerator operations without requiring prior offline training. On this static tuning problem, GP-MPC achieves performance comparable to that of BO. Future work will focus on extending the approach to incorporate safety constraints and prior knowledge of the accelerator lattice, leveraging the fast executing and differentiable Cheetah simulation model. We believe that integrating GP-MPC into accelerator control workflows offers a promising step toward more autonomous and intelligent tuning algorithms.

REFERENCES

[1] J. Kaiser *et al.*, “Reinforcement learning-trained optimisers and Bayesian optimisation for online particle accelerator

ISBN: 978-3-95450-248-6 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2025-THPM116

tuning", *Sci. Rep.*, vol. 14, no. 1, p. 15 733, 2024. doi:10.1038/s41598-024-66263-y

[2] S. Hirlaender and N. Bruchon, "Model-free and Bayesian ensembling model-based deep reinforcement learning for particle accelerator control demonstrated on the FERMI FEL", *arXiv:2012.09737*, 2022. doi:10.48550/arXiv.2012.09737

[3] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. The MIT Press, 2005. doi:10.7551/mitpress/3206.001.0001

[4] R. Roussel *et al.*, "Bayesian optimization algorithms for accelerator physics", *Phys. Rev. Accel. Beams*, vol. 27, no. 8, p. 084 801, 2024. doi:10.1103/PhysRevAccelBeams.27.084801

[5] M. P. Deisenroth and C. E. Rasmussen, "PILCO: A model-based and data-efficient approach to policy search", in *Proc. ICML'11*, pp. 465–472, 2011.

[6] S. Kamthe and M. P. Deisenroth, "Data-efficient reinforcement learning with probabilistic model predictive control", *arXiv:1706.06491 [cs]*, 2018. doi:10.48550/arXiv.1706.06491

[7] S. Hirlaender, L. Lamminger, G. Z. D. Porta, and V. Kain, "Ultra fast reinforcement learning demonstrated at CERN AWAKE", in *Proc. IPAC'23*, Venice, Italy, pp. 4510–4513, 2023. doi:10.18429/JACoW-IPAC2023-THPL038

[8] S. Appel, S. Hirlaender, and N. Madysa, "Data-driven model predictive control for automated optimization of injection into the SIS18 synchrotron", in *Proc. IPAC'24*, Nashville, TN, USA, pp. 1800–1803, 2024. doi:10.18429/JACoW-IPAC2024-TUPS59

[9] L. Hewing, K. P. Wabersich, M. Menner, and M. N. Zeilinger, "Learning-based model predictive control: Toward safe learning in control", *Annu. Rev. Control Rob. Auton. Syst.*, vol. 3, pp. 269–296, 2020. doi:10.1146/annurev-control-090419-075625

[10] J. Wang and Y. Zhang, "A tutorial on Gaussian process learning-based model predictive control", *arXiv:2404.03689 [cs]*, 2024. doi:10.48550/arXiv.2404.03689

[11] F. Burkart *et al.*, "The ARES Linac at DESY", in *Proc. LINAC'22*, Liverpool, UK, pp. 691–694, 2022. doi:10.18429/JACoW-LINAC2022-THPOJO01

[12] J. Kaiser, C. Xu, A. Eichler, and A. Santamaria Garcia, "Bridging the gap between machine learning and particle accelerator physics with high-speed, differentiable simulations", *Phys. Rev. Accel. Beams*, vol. 27, no. 5, p. 054 601, 2024. doi:10.1103/PhysRevAccelBeams.27.054601

[13] R. Roussel, C. Mayes, A. Edelen, and A. Bartnik, "Xopt: A simplified framework for optimization of accelerator problems using advanced algorithms", in *Proc. IPAC'23*, Venice, Italy, pp. 4847–4850, 2023. doi:10.18429/JACoW-IPAC2023-THPL164