ISBN: 978-3-95450-247-9 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2024-TUPS64

ESTIMATION AND CONTROL OF ACCELERATOR BEAMS BY LATENT SPACE TUNING OF GENERATIVE MODELS*

A. Williams[1, †], A. Scheinker, Los Alamos National Laboratory, Los Alamos, NM, USA
[1]also at University of California - San Diego, San Diego, CA, USA

Content from this work may be used under the terms of the CC BY 4.0 licence (© 2024). Any distribution of this work must maintain attribution to the author(s), title of the work, publisher, and DOI.

Abstract

In this work we explore the estimation and control of a particle accelerator simulation of the 800 MeV linac at Los Alamos National Lab. We use a convolution neural network model with a low dimensional latent space to predict the phase space projections of the beam and beam loss, which are mapped from accelerator settings. In deploying the model, we assume phase space predictions cannot be measured but beam loss can, and we apply a feedback using the error in beam loss prediction to tune the latent space. With beam loss and phase space predictions well correlated, we apply constrained optimization techniques, simultaneous with phase space prediction, to control the beam phase space while keeping beam loss from reaching unsafe levels.

INTRODUCTION

In the field of control engineering, recent work has explored *safe* extremum seeking. Extremum seeking (ES) in its classical form is designed to solve optimization problems, using a perturbation to dither control parameters while taking measurements of the objective function in order to compute gradients. Work in [1, 2] proposed designs which are guaranteed to be "practically" safe – meaning that safety violations can be made arbitrarily small with a choice of design parameters. This means that it approximately solves the constrained optimization problem

$$\min_{\theta(t)} J(\theta(t)) \text{ s.t. } h(\theta(t)) \geq 0 \text{ for all } t \in [0, \infty). \quad (1)$$

We achieve $h(\theta(t)) \geq O(\epsilon)$ for any small ϵ for all time, and $J(\theta(t))$ tending to an arbitrarily small neighborhood of the optimum. Here the function h is a measured metric of safety, and J is the measured objective. System safety is a priority for any large accelerator facility. Spilled beam is an important metric of safety as a misguided beam can cause damage to the accelerator and irradiate components, causing operational downtime, financial losses, and unsafe working conditions for personnel. Recently, safe extremum seeking has shown usefulness at LANSCE [3] demonstrating, both in simulation and on an experiment of the actual beamline, that it can tune various accelerator settings while maintaining low levels of beam loss.

In [4] an early approach to adaptive machine learning (ML) was demonstrated combining a deep neural network (NN) with extremum seeking [5] for automatic shaping of the longitudinal phase space of intense electron beams in the LCLS FEL. Recently an adaptive latent space tuning-based approach has been developed which utilizes a bounded adaptive feedback extremum seeking algorithm [6] directly within the low-dimensional latent space of an autoencoder convolutional NN [7] in order to track in real-time the time-varying 6D phase space of charged particle beams at the HiRES accelerator at LBNL and also at FACET-II at SLAC. It has been shown that this adaptive latent space tuning method makes generative NNs for predicting a beam's 6D phase space more robust and physically consistent for extrapolation further beyond that span of their training data than what is possible with traditional ML approaches that rely on brute-force re-training [8]. The work of [9] also demonstrated that a latent space encoding is also useful for prediction of phase space projections at future times, given a sufficient set of prior projections.

In this work we seek to blend these two tools 1) safe extremum seeking for safe tuning of the beam 2) latent space tuning for estimation of phase space projections. The work in latent space tuning has been shown to be an effective diagnostic tool, which is able to generate an estimation of a hard-to-measure quantity like 6D phase space. We adapt the tools used in latent space tuning approaches, and incorporate them into the work of [1, 2] which guarantees safe behavior when performing control. The control objective will be to minimize J, defined by the closeness of the estimated phase space with a desired phase space. The safety metric h is defined such that the measured beam loss is below a threshold.

Methods using ML and other techniques have been used in the past to both tune the beam and estimate various quantities about the beam or the accelerator itself (sometimes referred to as "diagnostics"). Safety was addressed in [10], which presented a Bayesian optimization based method to tune beamlines at PSI, incorporating the safety metric in a statistical sense using the underlying Bayesian framework. Other accelerator tuning algorithms span reinforcement learning (RL) based sample efficient methods [11], a modification of the Nelder-Mead simplex method [12], and a FPGA implemented RL based controller for fast implementation on the booster magnet power supply at Fermilab [13].

METHODS

Latent Space Estimation

A diagram of the proposed scheme is presented in Fig. 1. The estimator block runs in realtime on a pretrained NN, and generates an estimate of both the phase space projections $\hat{\mathbf{y}}_{ps}$ and the anticipated beam losses $\hat{\mathbf{y}}_{bl}$. The 54 dimensional

* Work supported by the Los Alamos National Laboratory, Los Alamos, NM, USA, through the Laboratory Directed Research and Development Project, under Grant 20220074DR
† awilliams@lanl.gov

ISBN: 978-3-95450-247-9 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2024-TUPS64

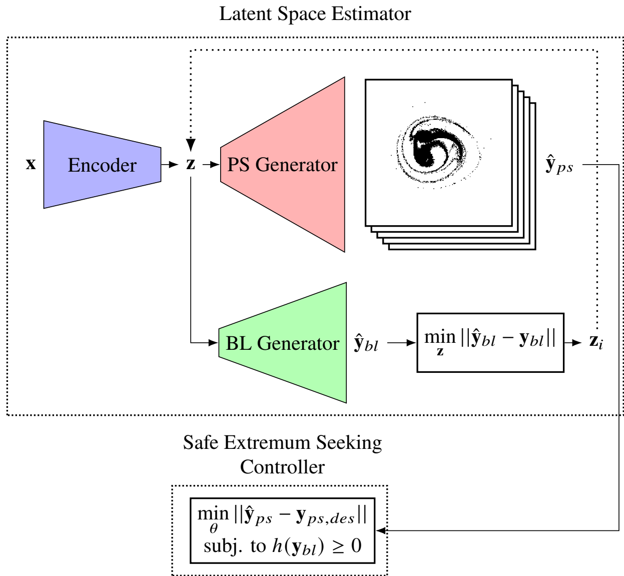

Figure 1: Diagram of the estimation and safe ES scheme. The phase space (PS) generator and the beam loss (BL) generator are decoder style networks, mapping from a latent variable $\mathbf{z}$ to estimates $\hat{\mathbf{y}}_{ps}$ and $\hat{\mathbf{y}}_{bl}$. The value $\mathbf{y}_{bl}$ is the measured beam loss which is used in both generate an estimate $\hat{\mathbf{y}}_{ps}$ and in performing the safe control actions.

vector $\mathbf{x}$ are various accelerator setting data consisting of magnet strengths, RF phases, and RF amplitudes. The 7 dimensional latent vector $\mathbf{z}$ encodes the vector $\mathbf{x}$ in order to generate the outputs $\hat{\mathbf{y}}_{bl}$ and $\hat{\mathbf{y}}_{ps}$. We gather a data set of $\mathbf{x}$ values and labels $\mathbf{y}_{bl}, \mathbf{y}_{ps}$ in order to train this NN.

Under the assumption that the outputs are well correlated and using the *known* beam loss measurement $\mathbf{y}_{bl}$, we solve the problem $\min_{\mathbf{z}} ||\mathbf{y}_{bl} - \hat{\mathbf{y}}_{bl}||$ to generate $\hat{\mathbf{y}}_{ps}$ without access to $\mathbf{x}$ on deployment. Here the norm defined is actually a weighted norm based on data sensitivity of vector elements from the training data set. The estimation scheme is based on work [4, 7]. Once an estimate is found, the safe ES controller seeks to drive the estimate $\hat{\mathbf{y}}_{ps}$ to a desired value $\mathbf{y}_{ps,des}$ by tuning accelerator settings θ (selected from the settings used to gather $\mathbf{x}$ to pretrain the NN), while maintaining beam loss below a threshold b_{th} [%]. After a step of safe ES is taken, the new beam loss is measured and the estimator solves to find a best $\hat{\mathbf{y}}_{ps}$ – this happens iteratively. Note that the estimation problem happens quickly as the inference times of the NN are on the order of milliseconds for single data points, and efficient on GPUs for a larger number of data points.

The vector $\mathbf{y}_{bl}$, given as a percentage, is a 48 dimensional vector of beam loss [%] at various locations on the beamline and the last downstream monitor will measure a total loss of $\mathbf{y}_{bl,48}$, so we define the safety constraint measurement as

$$h = b_{th} - \mathbf{y}_{bl,48}, \tag{2}$$

for some positive fraction (threshold) b_{th}. We use $b_{th} = 0.20$ and therefore $h = 1$ corresponds to 19 % beam loss.

Safe Extremum Seeking Control

The version of safe ES we consider is the algorithm in [2, 3], discretizing time from continuous to discrete steps. The vector θ is a set accelerator settings one wishes to tune, a subset of the accelerator settings used to generate $\mathbf{x}$. The quantities $\nabla J(\theta_n), \nabla h(\theta_n)$ are estimates of gradients of functions of the constrained optimization problem in (1), now discretized. The function J is an error defined by the closeness of two images (these are the desired as estimated longitudinal phase space projection for this study), shown as $||\hat{\mathbf{y}}_{ps} - \mathbf{y}_{ps,des}||$ in the safe ES block in Fig. 1. This norm which is actually performed involves element wise scaling by a log transformation before an MSE calculation between $\hat{\mathbf{y}}_{ps}$ and $\mathbf{y}_{ps,des}$.

The safe ES dynamics are essentially described as

$$\theta_{n+1} = \theta_n + k\big(\underbrace{-\nabla J(\theta_n)}_{\text{Gradient *Descent* of the Objective}} + \overbrace{\alpha \nabla h(\theta_n)}^{\text{Gradient *Ascent* of the Safety}}\big). \tag{3}$$

So the vector, θ, is imparted with gradient descent dynamics acting on the objective function *and* gradient ascent dynamics acting on the safety function. The quantity α is a non-negativ scalar function defined as

$$\begin{aligned} \alpha := & \min\{||\nabla h(\theta_n)||^{-2}, M^+\}\times \\ & \max\{\nabla J(\theta_n)^T \nabla h(\theta_n) - ch(\theta_n), 0\} \geq 0, \\ \approx & \frac{\max\{\nabla J(\theta_n)^T \nabla h(\theta_n) - ch(\theta_n), 0\}}{||\nabla h(\theta_n)||^2}, \end{aligned} \tag{4}$$

for $M^+ \gg 0$. It turns "on/off" to determine whether to consider safety and how much to consider it. The gradient of a function describes the direction steepest increase, and the term α therefore provides the corrective action which enforces safety such that $h(\theta_n) \geq 0$ approximately for all n.

At every step n a series of the forward difference steps (equal to the dimensions θ) are conducted to compute gradient estimates. For any given step the latent space estimate searches 3000 points for the best $\hat{\mathbf{y}}_{ps}$ based on the concurrent beam loss measurement (approximately 3 seconds inference time on a RTX A6000 GPU using a modified grid search).

RESULTS

We consider HPSim [14] as a model of the accelerator at LANSCE, running 131072 macroparticles along the beamline up to approximately 800 MeV. The set of accelerator settings are a vector of 54 values which are comprised of: 22 quadrupole magnet strengths in the LEBT, 12 quadrupole magnet strengths in the so called "ramp" section between modules 5 and 10, and modules 1-10 phase and amplitude values. The beam loss values are measured as a fraction of the total particles to lost particles after each of the 48 total modules in the linac. The phase space images are a histogram of the particles on a 256x256 grid in each of the 15 unique projections of 6D phase space for x, y, x_p, y_p, ϕ, and E. The boundaries of the grid are ± 3 cm in x and y, ±

© Content from this work may be used under the terms of the CC BY 4.0 licence (© 2024). Any distribution of this work must maintain attribution to the author(s), title of the work, publisher, and DOI.

ISBN: 978-3-95450-247-9 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2024-TUPS64

Content from this work may be used under the terms of the CC BY 4.0 licence (© 2024). Any distribution of this work must maintain attribution to the author(s), title of the work, publisher, and DOI.

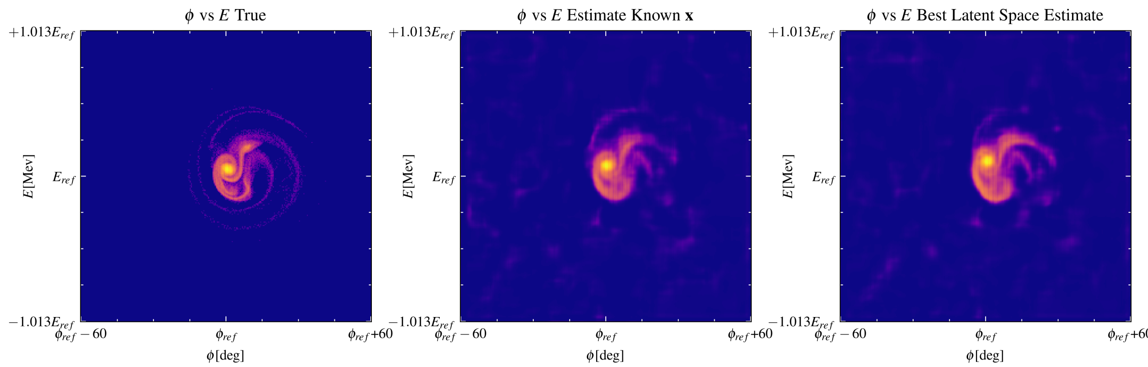

Figure 2: Estimates of the longitudinal phase space projection, showing the NN output with the accelerator settings known, and the estimate when performing latent space tuning using 3000 samples of $\mathbf{z}$ to solve $\min_{\mathbf{z}} ||\mathbf{y}_{bl} - \hat{\mathbf{y}}_{bl}||$. The data point is selected from the test set. Plots are given in high contrast color, with 90 %+ particles close to the yellow centroid.

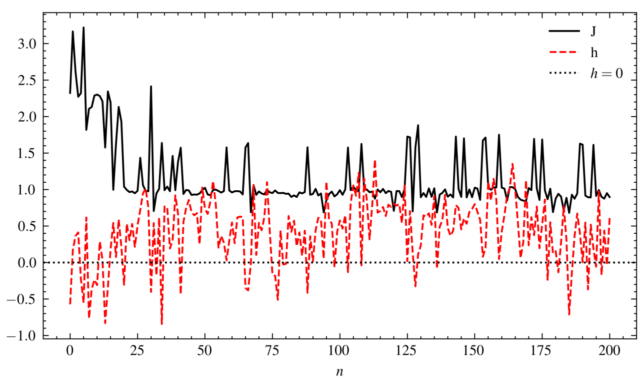

Figure 3: The metric J, the closeness between the desired and estimated longitudinal phase space image, and h, the difference between 20 % and the total beam loss.

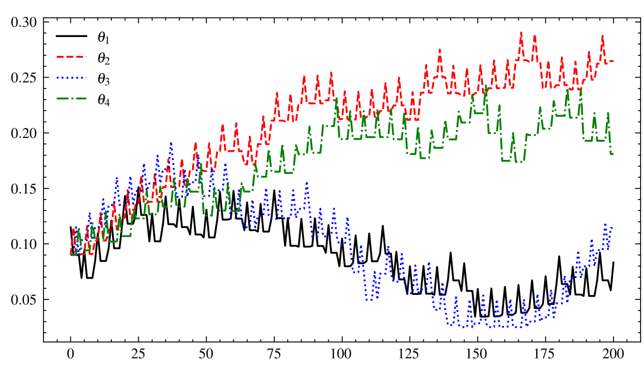

Figure 4: The parameter vector, corresponding to Module 1/2 amplitude and Module 1/2 phase, normalized to values between 0 and 1.

10 mrad in x_p and y_p, ± 60 deg + ϕ_{ref} in ϕ, $\pm 1.013 E_{ref}$ in E. The reference particle coordinates (for the average accelerator setting vector) in E and ϕ are E_{ref} and ϕ_{ref}. All data is normalized between 0 and 1 based on the range of values encountered in 1200 runs (20 % validation) of HPSim sampled at different accelerator settings, which are uniformly sampled in range of accelerator settings.

In Fig. 1 depicted are three neural networks. The encoder is made up of 6 dense layers, with between 128 and 32 neurons, with an output of 7 – the dimension of $\mathbf{z}$. The phase space generator consists of 15 identical architecture branches, consisting of 4 dense layers, and 4 convolutional layers and a sigmoid output activation. The beam loss generator consists of 7 dense layers, each having 48 neurons and the final layer with a sigmoid output activation.

From the test set $\mathbf{x}$, we choose a longitudinal phase space projection as the desired output $\mathbf{y}_{ps,des}$. We then perturb 2 RF phase and 2 RF amplitude setting and apply the framework in Fig. 1. We generate forward difference estimates of the gradients in the safe ES algorithm (3), eliminating the need for perturbation frequencies shown in the original work of [2]. Fig. 2 shows the phase space estimation performance for a data point collected from the test set.

In Fig. 3 and Fig. 4 a trajectory is shown for a marginally unsafe initial condition, close to $h = 0$. In 200 steps an optimum value of J is found constrained to the safe set with approximately less that 20 % beam loss. During tuning, no knowledge of the other 50 accelerator parameters were known to the latent space estimator.

CONCLUSION

We hope to expand this work in the future to consider all 15 phase space projection predictions as part of the control objective and refine the decoder generating phase space estimates. We show that safe ES can be utilized, working together with a ML based diagnostic tool, to track an unknown quantity while also maintaining system safety.

REFERENCES

[1] A. Williams, M. Krstić, and A. Scheinker, "Practically safe extremum seeking", in *IEEE 61st Conf. Decis. Control*, IEEE, 2022, pp. 1993–1998. doi:10.1109/CDC51059.2022.9993069

[2] A. Williams, M. Krstić, and A. Scheinker, "Semi-global practical extremum seeking with practical safety", in *IEEE 62st Conf. Decis. Control*, IEEE, 2023, pp. 6774–6779. doi:10.1109/cdc49753.2023.10383630

ISBN: 978-3-95450-247-9 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2024-TUPS64

[3] A. Williams, A. Scheinker, E.-C. Huang, C. Taylor, and M. Krstic, “Experimental safe extremum seeking for accelerators”, *IEEE Trans. Control Syst. Technol.*, 2024.

[4] A. Scheinker, A. Edelen, D. Bohler, C. Emma, and A. Lutman, “Demonstration of model-independent control of the longitudinal phase space of electron beams in the linac-coherent light source with femtosecond resolution”, *Phys. Rev. Lett.*, vol. 121, no. 4, p. 044 801, 2018.

[5] A. Scheinker and M. Krstić, “Minimum-seeking for clfs: Universal semiglobally stabilizing feedback under unknown control directions”, *IEEE Trans. Autom. Control*, vol. 58, no. 5, pp. 1107–1122, 2012.

[6] A. Scheinker and D. Scheinker, “Extremum seeking for optimal control problems with unknown time-varying systems and unknown objective functions”, *Int. J. Adapt. Control Signal Process*, vol. 35, no. 7, pp. 1143–1161, 2021.

[7] A. Scheinker, “Adaptive machine learning for time-varying systems: Low dimensional latent space tuning”, *J. Instrum.*, vol. 16, no. 10, p. P10008, 2021.

[8] A. Scheinker, F. Cropp, and D. Filippetto, “Adaptive autoencoder latent space tuning for more robust machine learning beyond the training set for six-dimensional phase space diagnostics of a time-varying ultrafast electron-diffraction compact accelerator”, *Phys. Rev. E*, vol. 107, no. 4, p. 045 302, 2023.

[9] M. Rautela, A. Williams, and A. Scheinker, “A conditional latent autoregressive recurrent model for generation and forecasting of beam dynamics in particle accelerators”, 2024.

[10] J. Kirschner, M. Mutnỳ, A. Krause, J.C. de Portugal, N. Hiller, and J. Snuverink, “Tuning particle accelerators with safety constraints using bayesian optimization”, *Phys. Rev. Accel. Beams*, vol. 25, no. 6, p. 062 802, 2022.

[11] V. Kain *et al.*, “Sample-efficient reinforcement learning for cern accelerator control”, *Phys. Rev. Accel. Beams*, vol. 23, no. 12, p. 124 801, 2020.

[12] X. Huang, “Robust simplex algorithm for online optimization”, *Phys. Rev. Accel. Beams*, vol. 21, no. 10, p. 104 601, 2018.

[13] J. S. John *et al.*, “Real-time artificial intelligence for accelerator control: A study at the fermilab booster”, *Phys. Rev. Accel. Beams*, vol. 24, no. 10, p. 104 601, 2021. `doi:10.1103/physrevaccelbeams.24.104601`

[14] X. Pang, “Advances in Proton Linac Online Modeling”, in *Proc. IPAC’15*, Richmond, VA, USA, May 2015, pp. 2423–2427. `doi:10.18429/JACoW-IPAC2015-WEXC2`

© Content from this work may be used under the terms of the CC BY 4.0 licence (© 2024). Any distribution of this work must maintain attribution to the author(s), title of the work, publisher, and DOI.