ISBN: 978-3-95450-231-8 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2023-THPL039

SURROGATE MODEL FOR LINEAR ACCELERATOR: A FAST NEURAL NETWORK APPROXIMATION OF THOMX'S SIMULATOR

E. Goutierre*[1,2], C. Bruni[1], J. Cohen[2], H. Guler[1], M. Sebag[2]
[1]Université Paris-Saclay, CNRS, IJCLab, Orsay, France
[2]Université Paris-Saclay, CNRS, LISN, Gif-sur-Yvette, France

Abstract

Accelerator physics simulators accurately predict the propagation of a beam in a particle accelerator, taking into account the particle interactions (*e.g.* space charge) inside the beam. A precise estimation of the space charge is required to understand the errors causing the difference between simulations and reality.

Unfortunately, the space charge is computationally expensive, needing the simulation of a few dozen thousand particles to obtain an accurate prediction. This work presents a Machine Learning-based approximation of the simulator output, *a.k.a.* surrogate model. Such an inexpensive surrogate model can support multiple experiments in parallel, allowing the exhaustive exploration of the simulator control parameters.

While the state of the art of surrogate models considers only a few parameters, our proposed approach LinacNet, scales up to one hundred parameters with broad domains. LinacNet uses a large-size particle cloud to represent the beam and estimates the particle behavior using a dedicated neural network architecture reflecting the architecture of a linear accelerator (Linac) and its different physical regimes.

INTRODUCTION

Accelerator physics simulators are widely used when designing a new machine as well as approaching simulation to the real accelerator (e.g. during the commissioning phase). To consider the interactions between particles, they solve the underlying electromagnetic equations using Particle-in-cell and Euler methods [1, 2]. These simulations are often time-consuming, limiting their usage in real-time use.

Machine-learning-based surrogate models are a tool to create a fast-executing replacement of such simulators by generating random simulations and learning to reproduce them accurately. Once trained, their generalization capability makes them suitable for searching for optimal control settings (surrogate optimization). Edelen *et al.* [3] stated that this process is faster than a direct optimization by running simulations. They can also be used as an assisting tool for operators to compare theoretical and experimental data without running expensive simulations. Well tuned, they can also provide virtual diagnostics [4] for the human operator, showing beam properties where there is no diagnostic station.

Notably, most surrogate models of particle accelerator simulators like [5] and [6] are themselves limited to only a few variables on a tiny domain. Even if this choice is adapted to the description of a well-known machine in operation, it is unsuitable for a machine under commissioning, potentially misaligned with unknown values of some physical parameters (like the Schottky effect and the emission delay at the cathode).

In this work, we explore a new physics-aware surrogate model that scales to a hundred parameters sampled on a large scale called LinacNet. This model also benefits from its modular particle-based architecture to give precise and understandable results.

This model is applied to ThomX [7], a compact backscattering Compton source, currently under commissioning at the IJCLab, Orsay, France. Its Linac is simulated with Astra [8].

LINACNET

The building block of a particle accelerator simulator like Astra is the macro-particle: a computational tool representing many real particles used to make time-efficient predictions. In Astra, each macro-particle is an 8D scalar vector $\mathbf{x}$ containing position, momentum, clock, and charge information. Taken together, they form a beam $D = \{\mathbf{x}_1, \dots \mathbf{x}_n\}$, whose evolution is governed by the electromagnetic field produced by the element of the accelerator and by the self-interaction of the particles (collective effects).

The simulator takes as input the description of the ThomX's machine and its state. This state can be represented as a scalar vector $\mathbf{a}$ whose values describe the position, the alignment, and the characteristics of the different elements of the machine.[1] Formally, a simulation is a tuple $(\mathbf{a}, (D_0, D_1, \dots, D_N))$ where D_0 is the initial beam entering the accelerator and $(D_i)_{i\in[\![1;N]\!]}$ are the beam recordings at N different positions corresponding to real or virtual diagnostic stations.

The surrogate model aims to accurately predict the particle coordinates for a control setting vector $\mathbf{a}$ sampled in the domain of interest. In our case, this domain corresponds to the commissioning of ThomX, potentially misaligned, whose behavior has to be studied extensively. While most of the existing surrogate models for particle accelerators limit themselves to a dozen variables sampled on a tiny domain, we chose to add up to 100 variables sampled on a broad domain to represent the potential states of the machine accurately. Our dataset comprises 16 000 simulations with the particle coordinates recorded at 25 positions. Among

* emmanuel.goutierre@universite-paris-saclay.fr

[1] We are only considering the variables that vary among the simulations in the vector $\mathbf{a}$.

14th International Particle Accelerator Conference,Venice, Italy JACoW Publishing
ISBN: 978-3-95450-231-8 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2023-THPL039

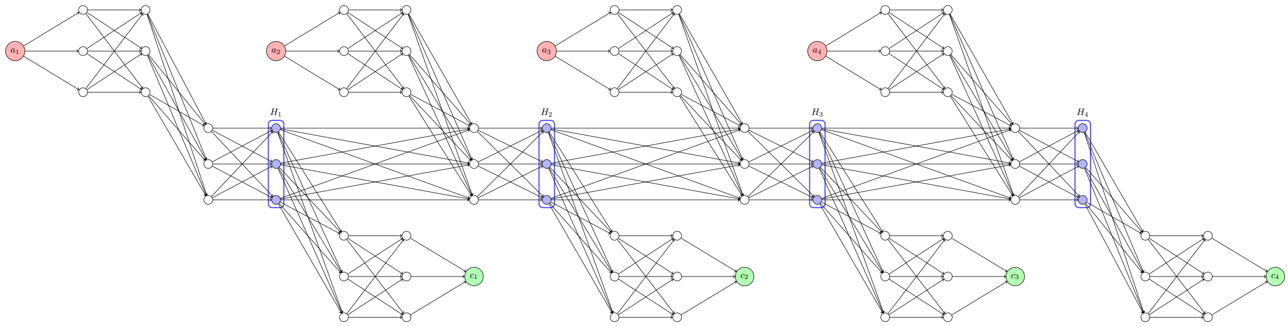

Figure 1: LinacNet architecture with 4 modules. The control setting vector **a** is split among the modules according to the position of its effect. The beam is represented by a vector H. In our case, H is the coordinate matrix of the macro-particles set.

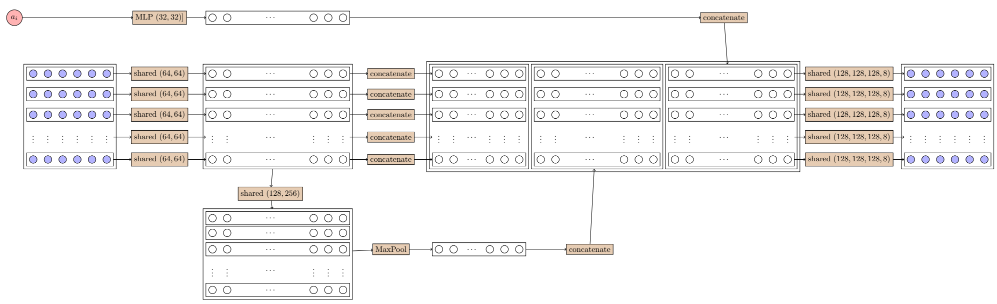

Figure 2: PointNet-like architecture to handle the particle-based nature of a beam. Each particle embedding is learned with a MLP. The features of the beam are extracted with a symmetric function over these features. Control settings embedding is learned with another MLP. Each particle's coordinates are predicted by concatenating all these embeddings as an input to the final MLP.

these simulations, only 25% of them reached the end of the machine. This last figure reflects the large domain used during the generation of the dataset.

LinacNet Architecture

We present the performance of LinacNet [9], a new surrogate model for particle accelerators based on particle-based and modular principles.

Particle-Based Surrogate Model At low energy, the interaction between particles is crucial in the evolution of the beam. It has been shown in Ref. [9] that classic feed-forward neural networks driven only by a few state indicators of the beam are not successfully reproducing the simulator's behavior. On the other hand, LinacNet takes as input the complete set of macro particles. This is made possible by using a PointNet-like neural network architecture [10]. This architecture aggregates the features of numerous particles with a single symmetric function (see Fig. 2). The resulting vector is a latent representation of the entire beam. This vector is then concatenated with each particle representation to predict their evolution.

Modularity The control setting vector **a** represents the state of the different elements in the accelerator. These elements are spread along the accelerator. By design, their state cannot modify the properties of the beam at positions preceding this element. LinacNet enforces this impossibility by spreading the values of **a** among multiple modules. Each module m_i approximate the behavior of the simulator between the $(i-1)$ -th and the i -th diagnostic station. It receives as input from **a** only the values that have a physical impact at this stage of the accelerator.

Finally, each particle cloud D_i is approximated by $m_j\left(\dots\left(m_1\left(D_0, \mathbf{a}_1; \theta_1\right)\dots\right), \mathbf{a}_j; \theta_j\right)$ where $(\theta_1, \dots, \theta_N)$ are the weight of the PointNet-like architecture and $(\mathbf{a}_1, \dots, \mathbf{a}_N)$ is the split version of the vector **a**. This mechanism is illustrated in Fig. 1.

Training Process The neural network parameters are learnt by minimizing the training loss with an Adam optimizer [11]. The training loss is defined as the sum of errors done on the prediction of each particle. This error combines a binary cross entropy targeting the classification problem of knowing if a particle has reached the given diagnostic and a mean squared error between its predicted and real coordinates.

RESULTS

The performance of our model lies in its capacity to recover the entire distribution of particles, even in significantly deteriorated situations. In addition to the convergence of the validation loss, we propose to visually compare the beam

ISBN: 978-3-95450-231-8 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2023-THPL039

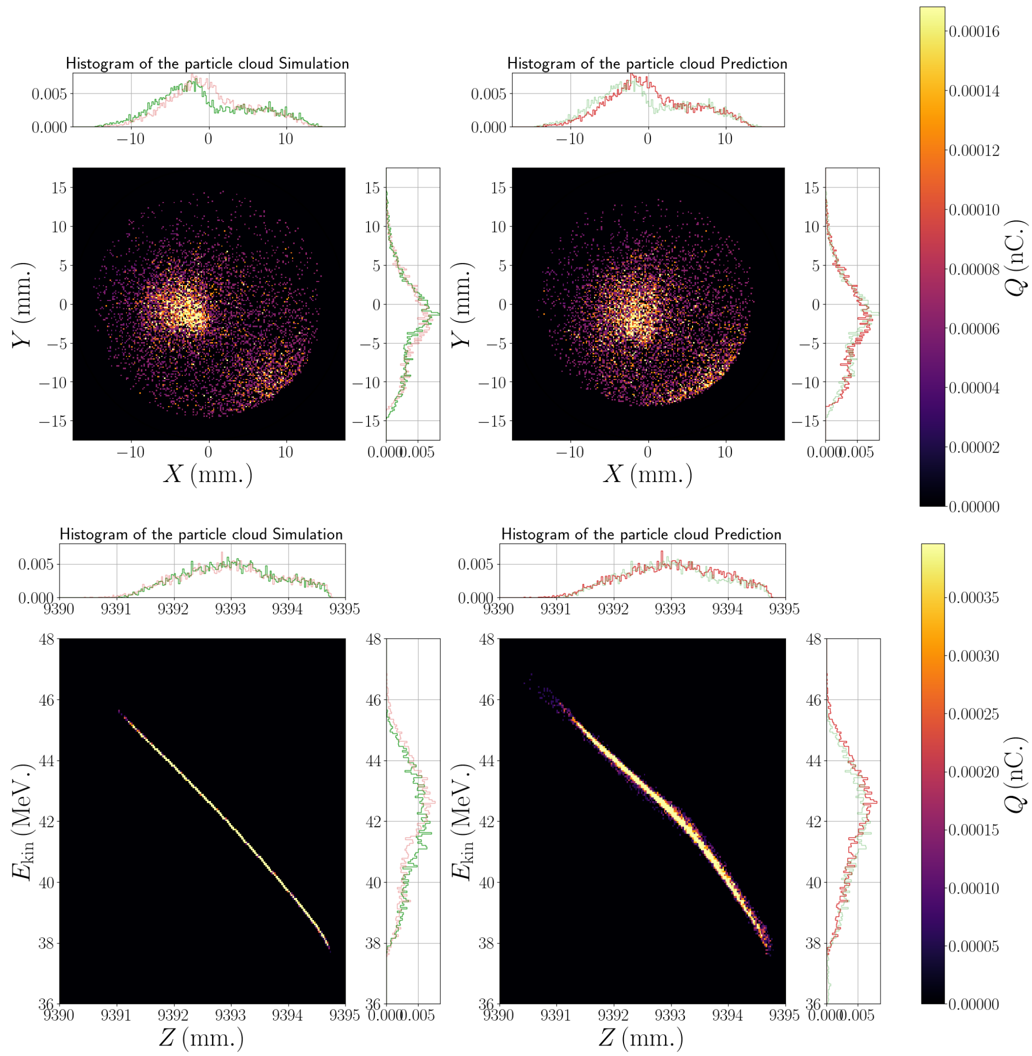

Figure 3: Comparison between the projection of the simulated beam (left) and predicted beam (right) on the transverse and longitudinal space. For comparison, histograms of the simulation are reported in green with low opacity on the right figures, and conversely histograms of the prediction are reported in red with low opacity on the left figures.

predicted by LinacNet to the simulated beam. In Fig. 3, we present two projections of a simulated beam at the end of the Linac and the prediction performed by LinacNet.

The simulation results from a random sampling of the 108 control parameters and results in a beam with unusual properties. In the transverse plane, we can identify two modes in the distribution, one in the center of the beam and another at the edge. LinacNet successfully recovers these aspects and the beam's overall size and position.

LinacNet can accurately predict the marginal distributions in the longitudinal space. However, the joint distribution could be improved, as the thickness of the predicted beam needs to be more accurate.

PERSPECTIVES

Our model successfully reproduces the full beam distribution at the end of the Linac, even in deteriorated situations.

Reality Gap Our model successfully reproduces the simulator's behaviour on a broad domain of over 100 variables. Such a wide sampling is performed to encompass the potential behavior of an in-commissioning machine. By comparing real-world data to the surrogate model predictions, one could recover the underlying hidden state of a machine.

Modularity The particle-based approach taken in our model enables the modularity of its architecture. The input and output of each module are the same as the ones of the simulator, suggesting potential interoperability with the simulator or with other surrogate models. Two use cases have been identified. Firstly, the simulator could handle the most sensitive part of the accelerator, leaving the rest to the surrogate model for more precise predictions. Secondly, an evolution of the machine could be reflected in the surrogate model by only adding or modifying a few modules.

ISBN: 978-3-95450-231-8 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2023-THPL039

REFERENCES

[1] R. A. Fonseca *et al.*, "OSIRIS: A three-dimensional, fully relativistic particle in cell code for modeling plasma based accelerators," in *Computational Science — ICCS 2002*, 2002, pp. 342–351.

[2] J. Qiang, R. D. Ryne, S. Habib, and V. Decyk, "An object-oriented parallel particle-in-cell code for beam dynamics simulation in linear accelerators," in *Proc. of the 1999 ACM/IEEE conference on Supercomputing*, 1999, p. 55.

[3] A. Edelen, N. Neveu, M. Frey, Y. Huber, C. Mayes, and A. Adelmann, "Machine learning for orders of magnitude speedup in multiobjective optimization of particle accelerator systems," *Phys. Rev. Accel. Beams*, vol. 23, p. 044 601, 2020. doi:10.1103/PhysRevAccelBeams.23.044601

[4] A. Hanuka *et al.*, "Accurate and confident prediction of electron beam longitudinal properties using spectral virtual diagnostics," *Sci. Rep.*, vol. 11, no. 1, p. 2945, 2021. doi:10.1038/s41598-021-82473-0

[5] A. Edelen, N. Neveu, C. Mayes, C. Emma, and D. Ratner, "Machine learning models for optimization and control of x-ray free electron lasers," in *NeurIPS Machine Learning for the Physical Sciences Workshop*, 2019.

[6] A. Scheinker, F. Cropp, S. Paiagua, and D. Filippetto, "An adaptive approach to machine learning for compact particle accelerators," *Scientific Reports*, vol. 11, no. 1, p. 19 187, 2021. doi:10.1038/s41598-021-98785-0

[7] K. Dupraz *et al.*, "The thomx ics source," *Physics Open*, vol. 5, p. 100 051, 2020. doi:10.1016/j.physo.2020.100051

[8] K. Flöttmann, "ASTRA: A space charge tracking algorithm, manual, 2017," Tech. Rep., 2017.

[9] E. Goutierre, H. Guler, C. Bruni, J. Cohen, and M. Sebag, "Physics-aware modelling of an accelerated particle cloud," 2023.

[10] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "Pointnet: Deep learning on point sets for 3d classification and segmentation," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[11] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.