ISBN: 978-3-95450-261-5 ISSN: 2673-7000 doi: 10.18429/JACoW-NAPAC2025-MOP102

IMPEDANCE MODELING OF IN-VACUUM UNDULATOR WITH GAUSSIAN PROCESS

Minghao Song*, Aamna Khan, Victor Smaluk, Guimei Wang, Michael Seegitz
Brookhaven National Laboratory, Upton, NY, USA

Abstract

The impedance of in-vacuum undulators (IVUs) significantly affect the broadband impedance and, consequently, the beam dynamics in storage rings. During the IVU design phase, numerous iterative discussions between physicists and engineers are required, often involving extensive simulations of the complete 3D geometry, a few meters long, using limited computational resources. In this paper, we propose training a Gaussian process model with limited simulation data to emulate the physical model. We compare the predictions of the trained model to the simulation data and explore its application in optimizing the IVU design.

INTRODUCTION

In-vacuum undulator (IVU) [1], in which magnet arrays are placed inside the vacuum chamber, is a key component for producing high-brightness photon beams in storage ring light sources [2–4]. By placing the undulator inside the vacuum, it allows a smaller vertical gap and more periods to generate shorter wavelength radiation and higher photon flux. However, a smaller gap will significantly influence the total coupling impedance of the storage ring [5], resulting in instabilities in the motion of the electron beam. Therefore, the design of an IVU needs to be optimized to minimize the impedance effects.

The IVU exhibits a complex geometry including tapers and transitions between cross sections. Due to its structural complexity, there are generally no accurate analytical theories to calculate its contribution to total impedance. As a result, it usually computes impedance through a comprehensive 3D simulation, which requires very expensive computation resources, including many number of working nodes, long processor time, and extensive computation memories. In particular, more computation is required when simulating the wakefields induced by short (e.g., a few millimeters) electron bunches.

The conventional way to design an IVU involves many back-and-forth conversations between engineers and accelerator physicists. For example, accelerator physicists calculate the impedance and provide feedback to engineers to make necessary changes to IVU geometry. Consequently, due to the expensive simulated computation and file exchange between engineers and accelerator physicists, it usually takes a long time before the structure of IVU is finalized. In addition, because the impedance of the IVU is affected by multiple parameters, therefore, it is difficult to find the best structure by manually changing the size of the IVU.

* msong1@bnl.gov

In this work, we propose a machine learning-based surrogate model to replace the computationally expensive IVU model, enabling significantly faster impedance calculations under varying IVU geometries. In this study, Gaussian process regression is used for model training due to the limited availability of data.

IN-VACUUM UNDULATOR

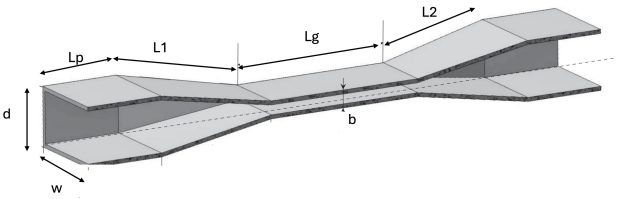

Figure 1: Simplified geometry of closed gap in-vacuum undulator.

Figure 1 shows the simplified geometry of the closed-gap in-vacuum undulator. The simplified IVU features a taper transition from the center to the ends, introducing additional complexity to the impedance computation. Table 1 summarizes the parameters that determine the simplified IVU and their valid ranges. To train the surrogate model, a total of 100 data points are randomly sampled from the parameter space, each parameter being constrained to integer values within its specified range due to engineering considerations. These data points serve as input features for the model.

Consequently, the output used for the model training is the kick factor, which quantifies the magnitude of the transverse kick experienced by the beam. Given the transverse wake function, the kick factor is defined as follows [5]:

$$k_y = \frac{1}{\sqrt{2\pi}\sigma_s} \int_{-\infty}^{\infty} W_y(s) e^{-s^2/2\sigma_s^2} ds, \qquad (1)$$

where σ_s is the rms length of a Gaussian electron bunch, and $W_y(s)$ is the vertical wake potential.

Table 1: Parameters of the Simplified IVU and Their Valid Ranges

Parameters	Range (mm)
b	5-25
d	20-30
w	50-100
L_p	50-100
L1, L2	150-300
L_g	500-800

ISBN: 978-3-95450-261-5 ISSN: 2673-7000 doi: 10.18429/JACoW-NAPAC2025-MOP102

GAUSSIAN PROCESS REGRESSION

Gaussian process (GP) is defined as multi-variate normal distribution of variables distributed over space (or time) [6]. Depending on the type of data input and output, it can be divided into regression and classification. Gaussian process regression uses the prior GP model and measured data to determine a posterior GP model to approximate the underlying physics model. As a surrogate model, it has been widely used in the accelerator community for optimization [7–10]. In the function space framework, the Gaussian process is specified by the mean function and the kernel function [6]. Without learning any knowledge, the mean function is usually defined as zero. The kernel function describes the covariance between variables at two points in the space.

Suppose that the input of the training data is $\mathbf{x} = [\mathbf{x_1}, \mathbf{x_2}, \ldots, \mathbf{x}_n]$ with corresponding outputs $\mathbf{y} = [y_1, y_2, \ldots, y_n]^\top$, where n represents the number of data points and the index of points $i = 1, 2, \cdots, n$. In our study, Automatic Relevance Determination (ARD) squared exponential kernel function is used and defined as [6]:

$$k(\mathbf{x}_i, \mathbf{x}_j) = \sigma_f^2 \exp\left(-\frac{1}{2}\sum_{d=1}^{D}\frac{(x_{i,d} - x_{j,d})^2}{\ell_d^2}\right), \quad (2)$$

where σ_f^2 is the signal variance, ℓ_d represents the length scale associated with the d-th input dimension. The terms $x_{i,d}$ and $x_{j,d}$ correspond to the d-th components of the input vectors $\mathbf{x}_i$ and $\mathbf{x}_j$, respectively. The input dimensionality is denoted by D, which is $D = 7$ in our case. Unlike the standard squared exponential kernel function, this type of kernel function uses a distinct length scale for each input dimension. The optimal hyper-parameters of σ_f^2 and ℓ_d are found by maximizing the marginal likelihood [6].

According to the kernel function , the kernel matrix is calculated as:

$$K = \begin{bmatrix} k(\mathbf{x}_1, \mathbf{x}_1) & k(\mathbf{x}_1, \mathbf{x}_2) & \cdots & k(\mathbf{x}_1, \mathbf{x}_n) \\ k(\mathbf{x}_2, \mathbf{x}_1) & k(\mathbf{x}_2, \mathbf{x}_2) & \cdots & k(\mathbf{x}_2, \mathbf{x}_n) \\ \vdots & \vdots & \ddots & \vdots \\ k(\mathbf{x}_n, \mathbf{x}_1) & k(\mathbf{x}_n, \mathbf{x}_2) & \cdots & k(\mathbf{x}_n, \mathbf{x}_n) \end{bmatrix}. \quad (3)$$

In scenarios where the output measurements are noisy, the noise variance σ_n^2 is incorporated by adding it to the diagonal of the kernel matrix. This results in a modified kernel matrix given by:

$$K_{\text{noise}} = K + \sigma_n^2 I, \quad (4)$$

where I is the $n \times n$ identity matrix.

Provided with a new input data point $\mathbf{x}_*$, the kernel vector is given by:

$$\mathbf{k}_* = \begin{bmatrix} k(\mathbf{x}_*, \mathbf{x}_1) \\ k(\mathbf{x}_*, \mathbf{x}_2) \\ \vdots \\ k(\mathbf{x}_*, \mathbf{x}_n) \end{bmatrix}. \quad (5)$$

Given the training data and the prior joint distribution, the Gaussian process predictive distribution for the output y_* at $\mathbf{x}_*$ is given by [6]:

$$\mu_* = \mathbf{k}_*^\top K_{\text{noise}}^{-1} \mathbf{y}, \quad (6)$$

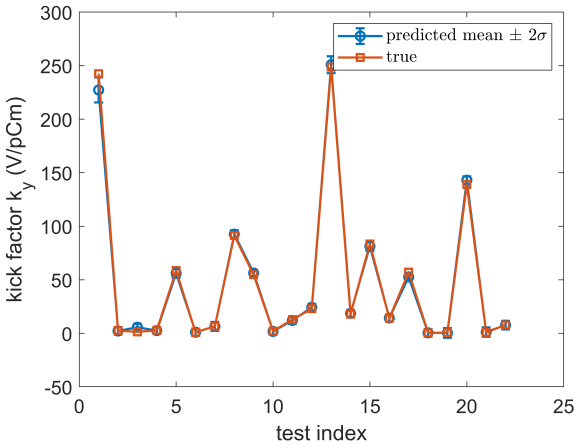

Figure 2: Comparison between the predicted kick factor and the true calculated kick factor on the test dataset generated by random sampling method.

and

$$\sigma_*^2 = k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^\top K_{\text{noise}}^{-1} \mathbf{k}_*, \quad (7)$$

where μ_* and σ_*^2 represent the predicted mean and variance for y_*, respectively. Therefore, the posterior Gaussian process yields both the predictive mean and the associated uncertainty.

TRAINING RESULTS

The Gaussian process model was trained on 75 samples randomly selected from a dataset of 97 valid calculations, each containing seven input variables and one response variable (i.e., the kick factor). The remaining 22 samples were used as a test dataset to evaluate the performance of the model.

Figure 2 shows the comparison between the predicted and true kick factors on the test dataset. The prediction includes the predicted mean and the 95% confidence intervals (i.e., ± 2 standard deviations). The predictions agree very well with the true values, with a coefficient of determination $R^2 = 0.9973$, indicating a highly accurate and reliable model prediction on new data.

However, the use of random sampling may lead to clustering of data points, which will limit the model's ability to represent the global parameter space effectively. To further evaluate the robustness of the trained GP model, an alternative sampling strategy was employed. Specifically, the Latin hypercube sampling (LHS) [11] method was used to generate a separate dataset consisting of 100 valid samples. Figure 3 presents a comparison of the heatmap of the correlation matrix resulting from the random sampling and LHS methods. The results show that the LHS method produces lower correlations among the input variables, indicating a more uniform coverage of the parameter space and, consequently, better representation of the global parameter space.

Figure 4 presents the comparison between the predicted kick factor and the true calculated kick factor for the dataset

ISBN: 978-3-95450-261-5 ISSN: 2673-7000 doi: 10.18429/JACoW-NAPAC2025-MOP102

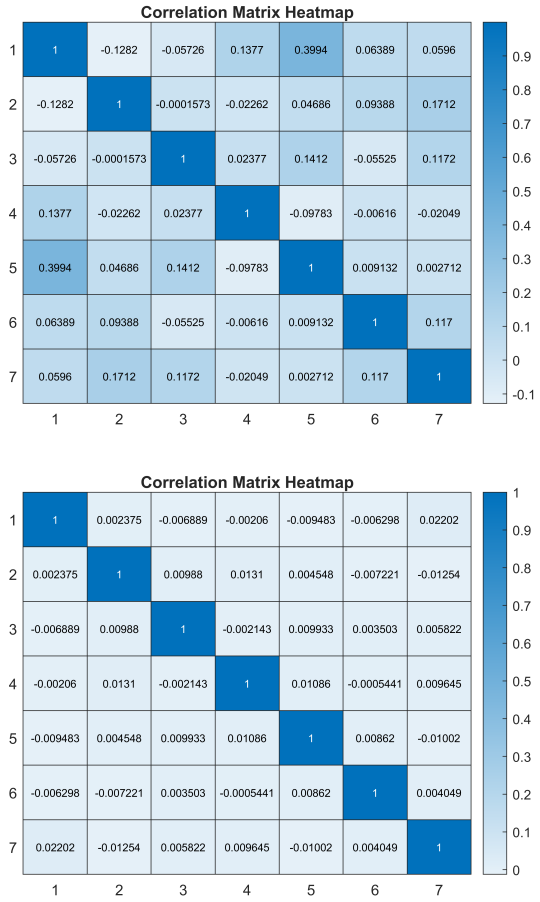

Figure 3: Comparison of the correlation analysis between the random sampling method (top) and the Latin hypercube sampling method (bottom).

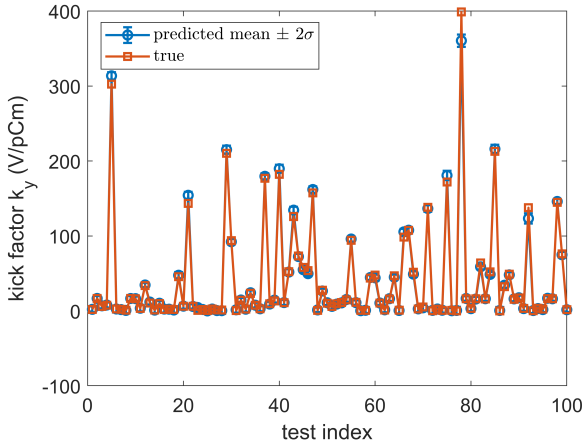

Figure 4: Comparison between the predicted kick factor and the true calculated kick factor on the dataset generated by LHS method.

generated using the LHS method. The results again demonstrate strong agreement between the predictions and the true values, with a coefficient of determination $R^2 = 0.9950$, indicating the high accuracy and robustness of the trained GP model. This excellent agreement suggests that the trained GP model is suitable for use as an emulator in subsequent IVU optimization.

CONCLUSION

In this work, a detailed study was conducted to model the impedance of the in-vacuum undulator using a machine learning approach based on Gaussian process regression. The predictive performance of the trained GP model was evaluated on two datasets generated using different sampling methods, demonstrating its accuracy and robustness. This study not only paves the way for future IVU optimization using the GP-based surrogate model, but also provides a general methodology for building surrogate models for other types of geometric components, such as bellows and flanges.

ACKNOWLEDGEMENTS

This work has been supported by the U.S. Department of Energy (DOE) under Contract No. DE-SC0012704. This information is controlled by the U.S. Department of Commerce Export Administration Regulations 15 CFR Parts 730-774 as EAR99 Technology.

REFERENCES

[1] S. Yamamoto *et al.*, "Construction of an in-vacuum type undulator for production of undulator x rays in the 5–25 keV region", *Rev. Sci. Instrum.*, vol. 63, no. 1, pp. 400–403, Jan. 1992. doi:10.1063/1.1142768

[2] T. Tanabe *et al.*, "Insertion devices at the National Synchrotron Light Source-II", *Synchrotron Radiat. News*, vol. 28, no. 3, pp. 39–44, 2015. doi:10.1080/08940886.2015.1037682

[3] H. Kitamura, "Insertion devices for third-generation light sources", *Rev. Sci. Instrum.*, vol. 66, no. 2, pp. 2007–2010, Feb. 1995. doi:10.1063/1.1145784

[4] J. Chavanne, C. Penel, B. Plan, and F. Revol, "In-vacuum undulators at ESRF", in *Proc. PAC'03*, Portland, OR, USA, May 2003, paper TOPA013, pp. 253–255.

[5] A. Khan, M. Seegitz, V. Smaluk, and X. Yang, "Comparison of simulation and measurement of an in-vacuum undulator coupling impedance at NSLS-II", in *Proc. IPAC'24*, Nashville, TN, May 2024, pp. 3098–3100. doi:10.18429/JACoW-IPAC2024-THPC46

[6] C. K. Williams and C. E. Rasmussen, *Gaussian processes for machine learning*. Cambridge, MA: MIT Press, 2006.

[7] X. Huang, M. Song, and Z. Zhang, "Multi-objective multi-generation gaussian process optimizer for design optimization", *arXiv*, 2019. doi:10.48550/arXiv.1907.00250

[8] M. Song, X. Huang, L. Spentzouris, and Z. Zhang, "Storage ring nonlinear dynamics optimization with multi-objective multi-generation gaussian process optimizer", *Nucl. Instrum. Methods Phys. Res. A: Accel. Spectrom. Detect. Assoc. Equip.*, vol. 976, p. 164273, 2020. doi:10.1016/j.nima.2020.164273

[9] Z. Zhang, M. Song, and X. Huang, "Online accelerator optimization with a machine learning-based stochastic algo-

ISBN: 978-3-95450-261-5 ISSN: 2673-7000 doi: 10.18429/JACoW-NAPAC2025-MOP102

rithm", *Mach. Learn.: Sci. Technol.*, vol. 2, no. 1, p. 015014, Dec. 2020. doi:10.1088/2632-2153/abc81e

[10] R. Roussel *et al.*, "Bayesian optimization algorithms for accelerator physics", *Phys. Rev. Accel. Beams*, vol. 27, no. 8, p. 084801, Aug. 2024. doi:10.1103/PhysRevAccelBeams.27.084801

[11] M. D. McKay, R. J. Beckman, and W. J. Conover, "A comparison of three methods for selecting values of input variables in the analysis of output from a computer code", *Technometrics*, vol. 21, no. 2, pp. 239–245, 1979. doi:10.2307/1268522