ISBN: 978-3-95450-221-9 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2021-WEPV011

RESEARCH ON CORRECTION OF BEAM BETA FUNCTION OF HLS-II STORAGE RING BASED ON DEEP LEARNING

Yong-Bo Yu*, Ke Xuan, Gong-Fa Liu, Wei Xu,Chuan Li, Wei-Min Li
National Synchrotron Radiation Laboratory, University of Science and Technology of China, Hefei Anhui, China

Abstract

In recent years, artificial intelligence (AI) has experienced a renaissance in many fields. AI-based concepts are nature-inspired and can also be used in the area of accelerator controls. At HLS-II, there are not many studies on these procedures. We focused on HLS-II beam stability in order to get better performance. We have created a deep learning-based approach for correcting beta function. Simulation studies reveal that the method presented in this work performs well in terms of correction outcomes and efficiency, resulting in a new way to adjust the accelerator beam function.

INTRODUCTION

Since the 1980s, artificial intelligence (AI) approaches in accelerator control have been studied [1]. In the light of recent theoretical and practical advances in machine learning and the use of deep neural network-based modeling and controlling techniques, new approaches for the control and monitoring of particle accelerators are emerging. Furthermore, the availability of powerful deep learning programmings frameworks like TensorFlow [2], PyTorch [3], Keras [4], and Matlab allow rapid and optimized implementations of complex algorithms and network architectures. Therefore, we propose a method based on a deep neural network to correct the beta function.

FEEDBACK THEORY

Corrective Theory for the Beta Function

The beta function is the lateral dynamic function of the particle, and it is one of the most important optical parameters of a beam. The focus intensity K of the quadrupole and the change $\Delta Q_{x,y}$ of the storage ring tune at this time are recorded so as to calculate the beta function of the quadrupole. When changing ΔK, the theoretical formula for maintaining the measured value of the ring beta function is

$$\beta_{x,y} = \pm \frac{2}{\Delta Kl}\Big(\cot\left(2\pi Q_{x,y}\right)\left[1-\cos\left(2\pi\Delta Q_{x,y}\right)\right] + \sin\left(2\pi Q_{x,y}\right)\Big) \quad (1)$$

The theoretical formula for the measured value of the function can be simplified as follows when the tune is far from the integer or half-integer resonance line, and the change value is small.

$$\beta_{x,y} \approx \pm 4\pi \frac{\Delta Q_{x,y}}{\Delta Kl} \quad (2)$$

According to the formula,the beta function at the location a quadrupole magnet is calculated using the variation of the quadrupole strength and the measured tune shift.

Using a Deep Learning Model to Conduct Beta Function Correction

The beta function of the storage ring receives feedback correction based on the generated storage ring beam function model. The feedback correction diagram of the HLS beam beta function is shown in Fig. 1. The steps involved in beta function feedback correction are followed: The focus intensity change of the storage ring quadrupole is obtained by feeding the beta function error value into the storage ring beam model. The focus intensity change is sent back to the storage ring to obtain the amended beta function value. This goes back and forth until the beta function is wholly corrected.

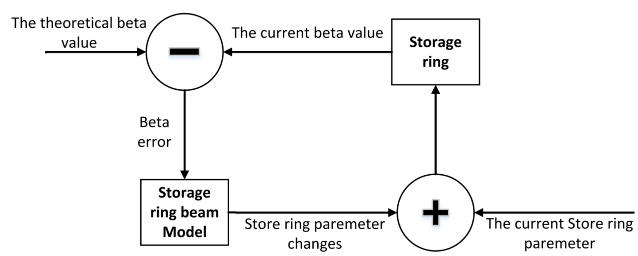

Figure 1: Schematic of the beta function correction system using a Neural network method.

MACHINE LEARNING DESIGN STEPS

Five significant steps are involved in developing an ML-based beat function application (see Fig. 2). Data acquisition and cleaning are the first steps. The topology of the neuron network is then defined and optimized. Finally, the beat function correction application must be tested after multiple training sessions and continuous performance tests. In the following sections, we will go over the most critical development steps.

Data Generation

In order to perform supervised neural network learning, A large number of data pairs must be provided. As a result, we took the Lattice as an object and created the HLS-II virtual storage ring model. The final data is finished with MATLAB's AT toolbox and a python program created with pyAT. After that, 10,000 data pairs were created. Each data pair has 96 data points, with Δbetax (32), Δbetay (32), and Δk (32).

* yybnsrl@mail.ustc.edu.cn

ISBN: 978-3-95450-221-9 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2021-WEPV011

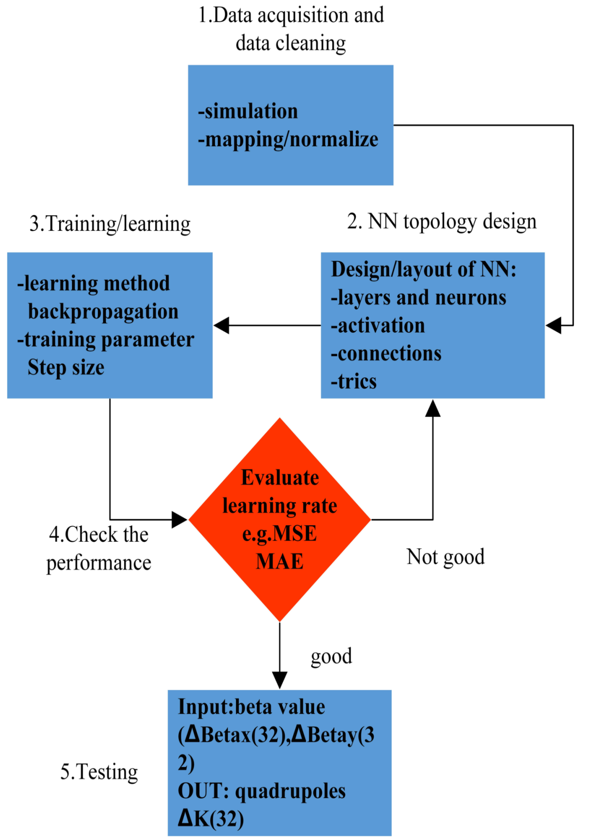

Figure 2: Development stages for an ML-based beta function correction.

Definition of Neural Network Structure

The topology diagram of the neural network after screening is illustrated in Fig. 3, which includes four hidden layers, batchnorm layers, and dropout layers. It has six layers of neurons, with 64 input neurons (related to the number of beta functions) and 32 output layer neurons (for the number of quadrupole values). The neurons in the buried layer are chosen based on previous experience [5]. Deep neural networks can approximate continuous functions of any complexity with arbitrary precision, according to research. For simulation, we use a deep feedforward neural network. The TensorFlow2.0 framework is used to create the deep neural network model.

Some Processing

Normalization is applied to both the input and output, and they are on a single uniform standard. Full connection is used in the neural network connection, and Dropout is used in this application. Dropout is a type of learning strategy that's used to improve neural networks. Dropout randomly disables a small number of neurons throughout each training phase, drastically reducing the complexity of the connection. It can also help to reduce overfitting caused by data while also improving the generalization of neural networks.

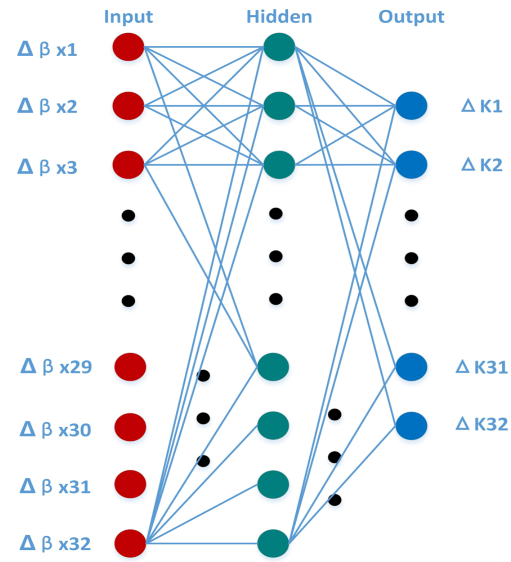

Figure 3: Feed Forward Neural Network (FFNN) topology used for ML-based beta function correction.

Model Train

The data comes from the previous virtual storage ring. A total of 1,000 groups are generated. These training data sets are divided into two parts in proportion.

1. data just for pure training, adjusting weights and bias values to minimize the MSE (80%)
2. Testing data is used to measure how accurately the network was trained (20%)

Adjusting the hyperparameters involves the first change, which is changing the learning rate. Let us use Adam, a gradient-based optimization approach rather than stochastic gradient descent. Because of his excellent computational efficiency and minimal memory footprint, Adam enjoys the advantages. It is pretty simple to use the hyperparameters, and just a few parameter tweaks result in the desired outcome. The iteration count is also critical. This is the number of times the entire training set is input to the neural network. Iterations can be considered adequate when the difference between test and training is minimal. It is over-fitting if the loss value initially decreases and then increases. The training times must be lowered. After testing, it was found that the number of epochs is 100 iterations, resulting in superior outcomes. The losses on the training set are known as "train loss" and on the test set as "value loss."MSE is a quantitative representation of model performance. So after around 80 iterations, the training set and test set stabilized. The MSE is less than 10-e5. Currently, it is confirmed that the neural network fitting algorithm's accuracy is acceptable, and the measured results are also rather dependable.

SIMULATION ANALYSIS

Therefore, we have corrected the beta function using a fairly complete neural network model so far. Through simulation, we generate beta function that have errors at random.

ISBN: 978-3-95450-221-9 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2021-WEPV011

The results are depicted in Fig. 4 using this Storage ring beam model. We see that after many corrections, the beta function close in on their theoretical values quickly. The method has been proven to be feasible. The beta functions of the storage ring was measured before and after the lattice correction for comparison [6]. A before-and-after comparison shows that the average beatx beating was 22.82% and 3.3%, respectively. However, due to the tiny fluctuation in vertical, the correction has not changed significantly.

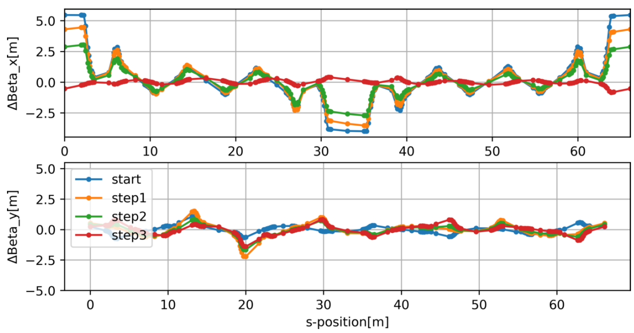

Figure 4: Beta parameters after iterations through the model.

CONCLUSION

The neural network approach is introduced to address the current beta function correction problem. Through the neural network model, the beta function is effectively corrected and has a good effect, providing fresh ideas and enhancing beam stability. Make a drawing. Other beam current properties on the storage ring can be studied using the neural network approach. It can be used to solve various optimization issues, including multi-parameter, multi-objective, so on and so forth.

ACKNOWLEDGMENT

The authors would like to thank all our colleagues at NSRL who provided us technical supports.

REFERENCES

[1] D. P. Weygland, "Artificial Intelligence and Accelerator Control", in *Proc. PAC'87*, Washington, DC, USA, pp. 564–566.

[2] TensorFlow, `https://www.tensorflow.org`

[3] PyTorch, `http://pytorch.org/`

[4] Keras, `https://keras.io/`

[5] D. Schirmer, "Orbit Correction With Machine Learning Techniques at The Synchrotron Light Source Delta", in *Proc. ICALEPCS'19*, New York, NY, USA, doi:10.18429/JACoW-ICALEPCS2019-WEPHA138

[6] K. Xuan, G. Liu, C. Li, J. Wang, L. Wang, W. Li, and J. Y. Li, "High Level Applications For HLS-II", in *Proc. ICALEPCS'15*, Melbourne, Australia, Oct. 2015, pp. 974–976. doi:10.18429/JACoW-ICALEPCS2015-WEPGF117