ISBN: 978-3-95450-209-7 ISSN: 2226-0358 doi:10.18429/JACoW-ICALEPCS2019-TUCPL07

OPTIMAL CONTROL FOR RAPID SWITCHING OF BEAM ENERGIES FOR THE ATR LINE AT BNL[*]

J. P. Edelen[†], N. M. Cook, RadiaSoft LLC, Boulder, USA
K. A. Brown, P. Dyer, Brookhaven National Laboratory, Upton, USA

Abstract

The Relativistic Heavy Ion Collider (RHIC) at Brookhaven National Laboratory will undergo a beam energy scan over the next several years. To execute this scan, the transfer line between the Alternating Gradient Synchrotron (AGS) and RHIC or the so-called the ATR line, must be re-tuned for each energy. Control of the ATR line has four primary constraints: match the beam trajectory into RHIC, match the transverse focusing, match the dispersion, and minimize losses. Some of these can be handled independently, for example orbit matching. However, offsets in the beam can affect the transverse beam optics, thereby coupling the dynamics. Furthermore, the introduction of vertical optics increases the possibilities for coupling between transverse planes, and the desire to make the line spin transparent further complicates matters. During this talk, we will explore three promising avenues for controlling the ATR line: model based control, on-line optimization methods, and hybrid model based and optimization methods. We will provide an overview of each method, discuss the tradeoffs between these methods, and summarize our conclusions.

INTRODUCTION

The Relativistic Heavy Ion Collider (RHIC) at Brookhaven National Laboratory will undergo a beam energy scan [1] over the next several years. To execute this scan, the transfer line between the Alternating Gradient Synchrotron (AGS) and RHIC or the so-called the ATR line [2, 3], must be retuned for each energy. This transfer line controls the orbit matching, optics matching, and dispersion matching, of the beam into RHIC. The optics are further complicated by a 1.7 m vertical drop in order to get the beam from the AGS to RHIC. In order to ensure optimum performance of RHIC during the energy scan the magnets and correctors will need to be properly set on demand. A high-level diagram of the transfer line is shown in Figure 1.

The first part of the ATR (referred to as the U-line) line starts with the fast extraction from the AGS and stops before the vertical drop from the AGS to RHIC. The U-line consists of two bends. The first bend is 4.25° consisting of two A-type dipole magnets. The second bend is an 8° bend consisting of four C-type combined function magnets (placed in a FDDF arrangement), and thirteen quadrupoles.

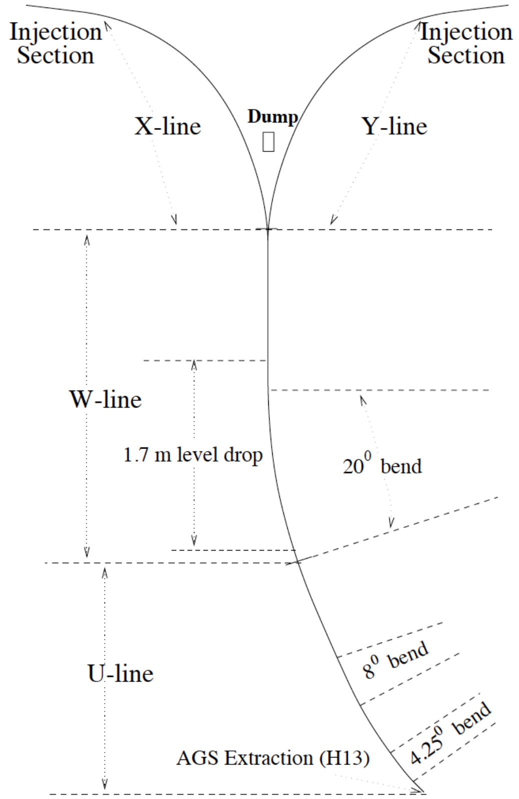

Figure 1: High level schematic of the RHIC transfer line [4].

The primary purpose of the U-line is to: 1) Match the Twiss parameters at the AGS extraction point and provide achromatic transport of the beam to the exit of the 8° bend, 2) Create a beam waist with low beta function values at the location of a thin gold foil which is placed just upstream of the quadrupole Q6 of the U-line, 3) Match the Twiss parameters of the line to the ones at the origin of the W-line, and 4) Keep the beam size small to minimize losses.

The second part of the ATR line (referred to as the W-line) introduces the vertical drop for injection into RHIC and the matching sections for the injection lines. It contains eight C-type combined function magnets that each make a of 2.5° bend, followed by six quadrupoles. The eight combined function magnets form a 20° achromatic horizontal bend placed in a (F-D) configuration. The W-Line is also responsible for lowering the beam elevation by 1.7 m. This is accomplished by two vertical dipoles referred to as pitching magnets. The first bends the beam down, and is located between the first and second combined function dipoles of the W-line. The second, which restores the beam to the horizontal level (bend-up), is located between the second and third

[*] This material is based upon work supported by the U.S. Department of Energy, Office of Science, Office of Nuclear Physics under Award Number DE-SC0019682.
[†] jedelen@radiasot.net

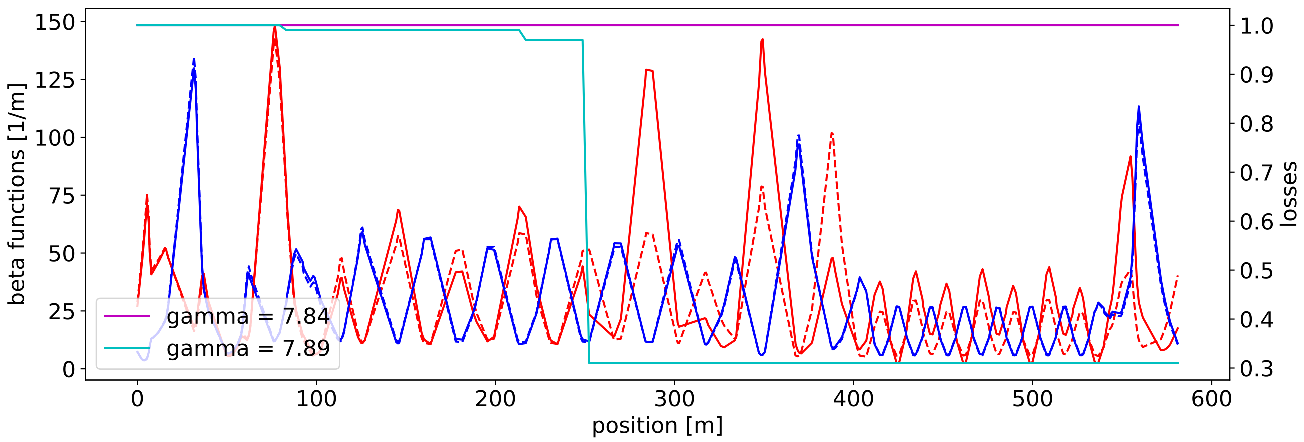

Figure 2: Left axis: Beam envelope functions as a function of position along the ATR line for two different energies. Right axis: beam losses along the ATR line for the different energies.

quadrupoles of the W-line. The beam section between the two pitching magnets is designed to be non-dispersive in the vertical direction, introducing linear beam-coupling which is not significant as far as the first-order beam transport optics are concerned. However, this simultaneous vertical and horizontal bend of the beam turns out to be a concern when polarized protons are to be transported. Along the line there are also a number of BPMs and correctors that are required to match the orbit of the beam into RHIC. Figure 2 shows the beam envelope for two different beam energies through the ATR line with the loses plotted on the right axis. Here we can see that even for small energy deviations the losses can be significant, highlighting the need for automated tuning when transitioning between operating conditions.

In this paper we will study the use of optimization and machine learning for rapid reconfiguration of the ATR line.

A PYTHON MIDDLE-LAYER FOR MAD-X

To facilitate the development of automated tuning programs we have constructed a Python based middle layer that can interact with MAD-X simulations as well as dynamically modify the model based on external input from the users. This allows one to not only take advantage of the suite of tools available in MAD-X but also the tools available in Python. For example Python optimization packages, machine learning tools, and advanced visualization tools. Figure 3 shows a block diagram of our middle layer and the different avenues for interfacing with MAD-X.

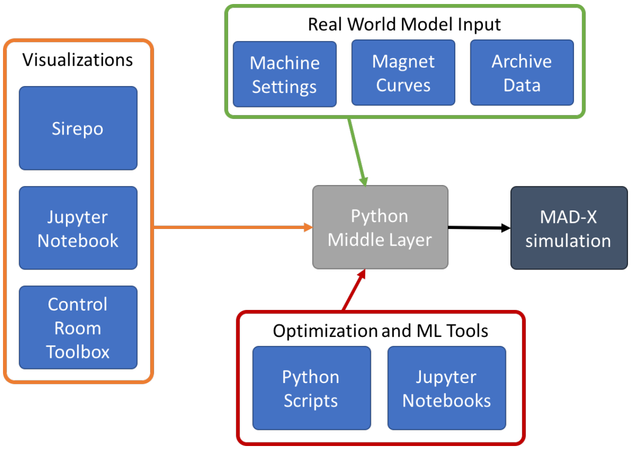

Figure 3: Schematic of the MAD-X middle layer used for optimization simulations and machine learning.

OVERVIEW OF CONTROL METHODS

Here we provide an overview of the control methods used for this study. We are considering model based, model independent, and hybrid methods to tune the transfer line. Model independent methods have an advantage over model dependent methods as they do not require accurate modeling of the machine and can be applied independent of machine sate. For example, machine drift will ultimately lead to new operating conditions that require reconfiguration of the machine. However, model independent methods can accommodate this drift without re-training. Some common methods for model independent techniques include online optimization. Classical PID is also a form of model independent control.

Model dependent methods have an advantage over model independent methods due to their ability quickly reconfigure the machine as they do not rely on a feedback mechanism. These feed-forward methods however are subject to steady state offsets due to machine drift or incomplete models of the machine. Furthermore these models may require updating over time to account for drift. Some examples of model dependent methods are model predictive control, which is useful for systems with long time delays and time dynamics, linear or nonlinear model based tuning, or online model based tuning of a machine. In the latter case the models can be either learned from machine data, accurate simulations of the machine, or hybrid techniques using artificial neural networks where the model is trained on both simulation data and measured data.

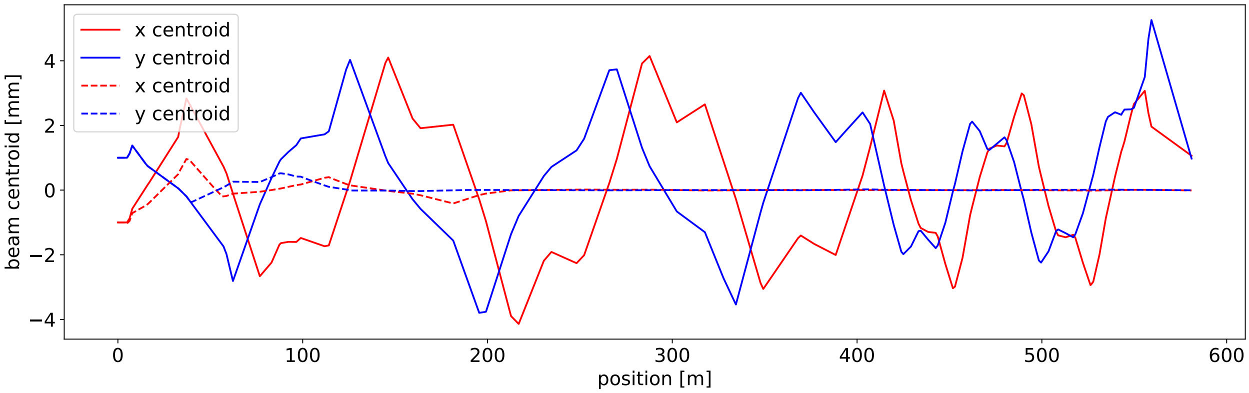

Figure 4: Beam trajectory before and after application of Nelder-Mead optimization for tuning the trajectory through the beam-line.

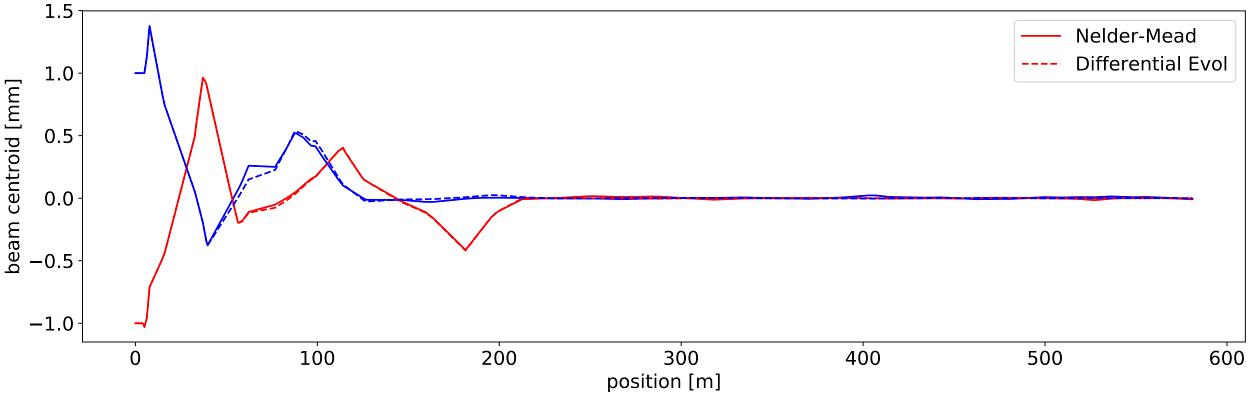

Figure 5: Comparison of the final trajectory using Nelder-Mead optimization and Differential Evolution.

There are also hybrid methods where you use a model to get close to a desired operating point and then apply optimization to fine tune the settings and optimize the machine state. This is akin to using a feed-forward term in a PID controller. For tuning the ATR at different energies, hybrid methods are the optimal solution due to the difficulties in accurately modeling such a complicated transfer line.

For a broader overview of these methods and how they can be applied to accelerators see [5, 6]

OPTIMIZATION BASED TRAJECTORY CONTROL

Our initial studies are focused on trajectory control using optimization methods. We have examined the use of Nelder-Mead optimization and Differential Evolution to tune the beam trajectory using the correctors. This study lays the groundwork for the developing the necessary tools for automatic reconfiguration of the ATR line. For each of the optimization runs the beam-line is initialized with zero excitation in the correctors. We tuned 23 corrector magnets to minimize the sum of the squares of the beam position along the beam-line. The cost function was defined by Equation 1.

$$F(\mathbf{c}) = \sum_i x(\mathbf{c})^2_{0,i} + \sum_i y(\mathbf{c})^2_{0,i} \tag{1}$$

Here **c** are the corrector settings $x_{0,i}$ and $y_{0,i}$ are the beam offsets in the two transverse planes at the i^{th} BPM reading. Figure 4 shows the beam trajectory as a function of position along the ATR line with an initial offset in the x and y plane of 1 and -1 mm respectively and the result of the Nelder-Mead optimization.

The approach is able to correct an initial trajectory mismatch, producing a beam at the exit of the ATR with no net offset nor centroid divergence. In general the Nelder-Mead optimizer converged in approximately 2500 - 2800 iterations. We also explored the use of Differential Evolution to tune the beam-line. This has the advantage of being a more global optimization technique and may be better suited for tuning the beam-line for different energies. Figure 5 shows the final trajectory using the settings from the Differential Evolution optimizer and the Nelder-Mead optimizer.

When comparing the beam trajectory resulting from the Nelder-Mead optimization and the Differential Evolution optimization the curves are quire similar. However, the Differential Evolution algorithm required a factor of 5 more iterations to converge. This could be related to hyper parameter tuning of the evolutionary algorithm or simply be a result of the global nature of the optimizer. In general the optimizer should be chosen based on the needs of the problem. Our framework provides a tool for seamlessly interfacing MADX to state-of-the-art Python optimization tools that can also integrate non-ideal machine characteristics.

MODEL BASED TRAJECTORY CONTROL

We also explored the use of model based tuning of the trajectory through the ATR line. Some form of model tuning will be necessary to move the machine in large parameter spaces before using optimization for fine tuning. Artificial neural networks are a compelling tool for this application due to their ability to capture complex relationships in large nonlinear parameter spaces. While trajectory control is not in principle a nonlinear problem, the tools developed for trajectory control will be applied to more complicated nonlinear problems. Additionally, using a nonlinear model allows for us to account for magnet excitation curves and for other nonlinearities in the machine that are not captured by traditional methods.

We trained our neural network on 5000 MADX simulations with random initial beam offsets and random corrector settings. The inputs to the network are the beam positions along the beam-line and the outputs are the corrector settings. In effect we are using a neural network to learn an inverse model of the MADX simulations. To apply this method the user would specify a desired trajectory and the network would return a set of corrector settings to achieve this trajectory. Figure 6 and 7 show the inputs and outputs to the neural network respectively. The training data and validation data are shown in blue and orange respectively.

In general the network does a pretty good job of predicting the corrector settings from a given beam trajectory across the dataset. However, there are three parameters that do not perform well and furthermore the relationship between the second corrector and the beam trajectory is not captured at all by the neural network. This indicates that more work is needed to fine tune our model. Figure 8 shows the loss function as a function of epoch on both the training and validation set.

The training and validation data were split using an 80/20 ratio and the mini-batch size was 100 samples. The network architecture contained 5 fully connected layers with three hidden layers containing 40, 40, and 50, nodes respectively. Gaussian noise layers were added between each hidden layer. A mean squared error loss function was used for training. Training was terminated after 5000 epochs. Figure 9 shows the predicted outputs as a function of the real outputs for the validation set.

On a global scale the network is learning the relationship between BPMs and correctors and we are not seeing any overfitting. Next we use the neural network to compute the corrector settings for a desired trajectory. For this problem we used an offset of 0.5mm in both planes and requested that the beam offset be zero along the remainder of the beamline. The neural network provided corrector settings that we gave to the MADX simulation to compute the real trajectory given these proposed corrector settings. Figure 10 shows the uncorrected and the corrected trajectory using the neural network.

While the neural network does does a very good job of tuning the trajectory it is not as good as the result from

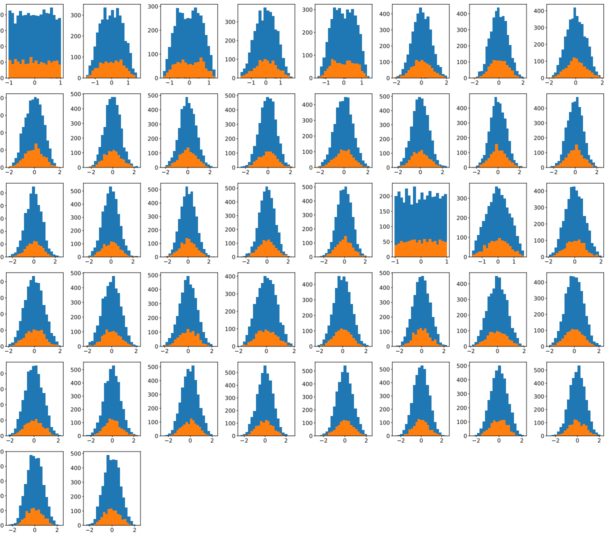

Figure 6: Histogram of the inputs to the neural network used to train the inverse model. The training set is in blue and the validation set is in orange.

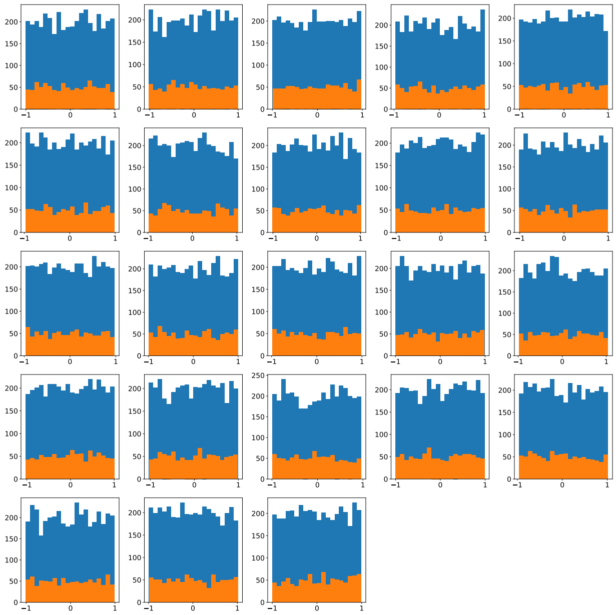

Figure 7: Histogram of the outputs from the neural network used to train the inverse model. The training set is in blue and the validation set is in orange.

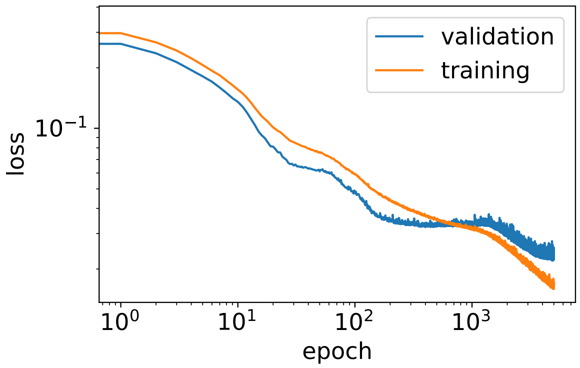

Figure 8: Loss function during training on both the training and validation set.

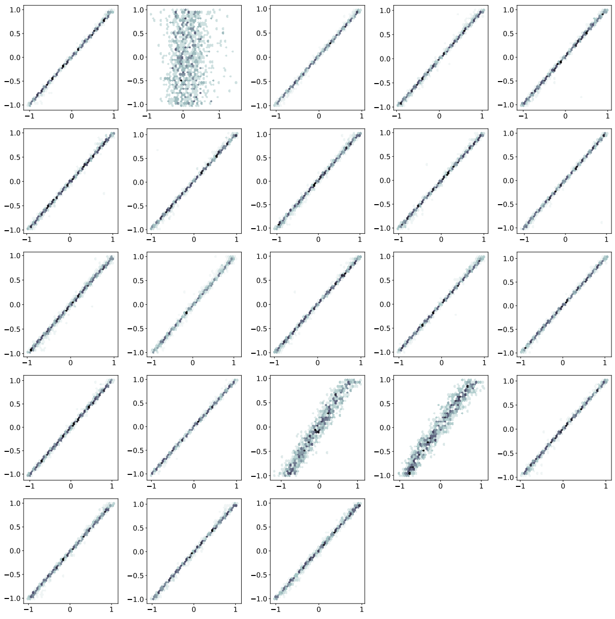

Figure 9: Predicted corrector settings as a function of the real corrector settings on for the validation set.

optimization. This can be due to a variety of factors, either the model is not accurately capturing the input and output relationships or the inputs and outputs to the model do not accurately capture the results from the MADX simulation. To improve the trajectory we can apply hybrid techniques where the neural network provides a starting point for the optimizer.

HYBRID TECHNIQUES

The developed interface enables the use of a hybrid approach, which combines both a neural network model and subsequent optimization steps. Figure 11 shows the initial and final trajectory after we use the neural network to guess the starting position and then use Nelder-Mead to optimize the trajectory.

Here the final trajectory is optimized quite well as expected given results shown on optimization alone. Using the neural network suggested initial settings reduced the number of iterations required by the optimizer by almost half. If the neural network model is improved perhaps using reinforcement learning directly on the MADX simulations, this payoff is expected to improve.

CONCLUSIONS

We have demonstrated the use of new tools that make it easier to interface MADX simulations and custom properties form the machine with modern Python optimization packages and machine learning packages. We have shown that optimization is quite effective at trajectory control for the ATR line and set the foundation for studies at different energies and for matching transverse optics. We have also shown that machine learning can be effective for providing initial settings to local optimizers for trajectory control. Our next efforts will extend this work to control at different energies and control over the transverse beam optics.

REFERENCES

[1] D. Keane, “The beam energy scan at the Relativistic Heavy Ion Collider”, *Journal of Physics: Conference Series*, vol. 878, no. 1, p. 012015, 2017. doi:10.1088/1742-6596/878/1/012015

[2] W. W. MacKay *et al.*, “AGS to RHIC transfer line: Design and commissioning”, in *Proc. 5th European Particle Accelerator Conference (EPAC’96)*, Sitges, Spain, Jun. 1996, paper MOP005G, pp. 2376–2378.

[3] T. Satogata *et al.*, “Physics of the AGS-to-RHIC transfer line commissioning”, in *Proc. 5th European Particle Accelerator Conference (EPAC’96)*, Sitges, Spain, Jun. 1996, paper MOP006G, pp. 2379–2381.

[4] N. Tsoupas *et al.*, N. Tsoupas *et al.*, “Focusing and Matching Properties of the ATR Transfer Line”, in *Proc. 17th Particle Accelerator Conf. (PAC’97)*, Vancouver, Canada, May 1997, paper 8P006, pp. 222-224.

[5] A. L. Edelen, S. G. Biedron, B. E. Chase, D. Edstrom, S. V. Milton, and P. Stabile, “Neural Networks for Modeling and Control of Particle Accelerators”, *IEEE Transactions on Nuclear Science*, vol. 63, no. 2, pp. 878-897, April 2016. 10.1109/TNS.2016.2543203. doi:10.1109/TNS.2016.2543203

[6] A. Scheinker, A. L .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, p. 044801, Jul. 2018. doi:10.1103/PhysRevLett.121.044801

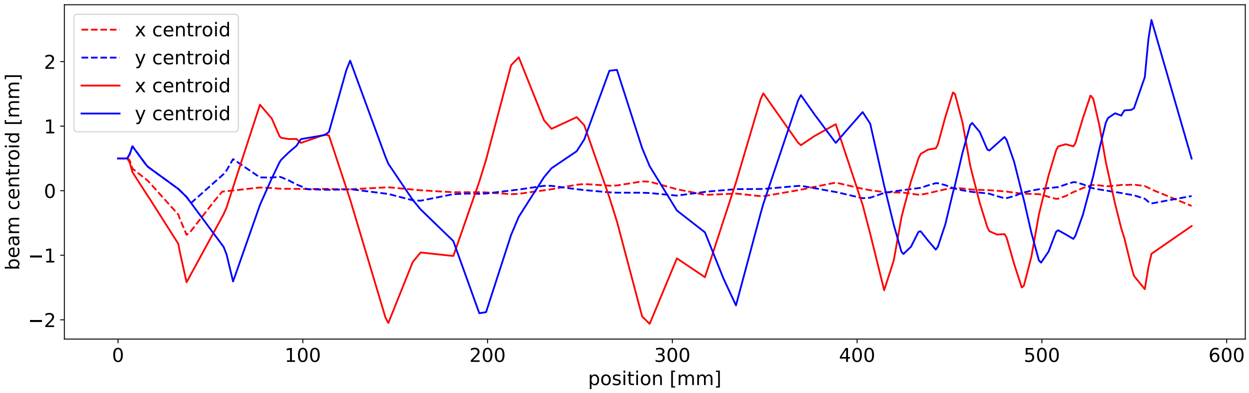

Figure 10: Trajectory of beam using corrector settings computed from the neural network.

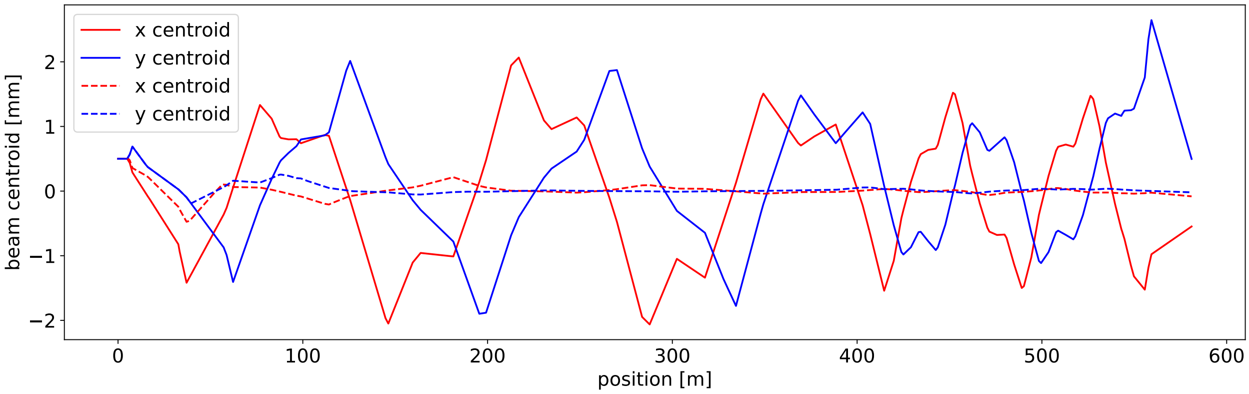

Figure 11: Trajectory of beam using corrector settings computed from the neural network and then optimized using Nelder-Mead.