ISBN: 978-3-95450-214-1 ISSN: 2673-5490 doi:10.18429/JACoW-IPAC2021-WEPAB306

APPLYING MACHINE LEARNING TO OPTIMIZATION OF COOLING RATE AT LOW ENERGY RHIC ELECTRON COOLER*

Y. Gao†, K. A. Brown, P. Dyer, S. Seletskiy, H. Zhao
Collider-Accelerator Department, Brookhaven National Laboratory, Upton, NY, USA

Abstract

The Low Energy RHIC electron Cooler (LEReC) is a novel, state-of-the-art, electron accelerator for cooling RHIC ion beams, which was recently built and commissioned. Beam dynamics in LEReC differs significantly from conventional electron coolers. Optimization of cooling with LEReC requires fine-tuning of numerous LEReC parameters. In this work, initial optimization results of using Machine Learning (ML) methods - Bayesian Optimization (BO) and Q-learning are presented. Specially, we focus on exploring the influence of the electron trajectory on the cooling rate. In the first part, simulations are conducted by utilizing a LEReC simulator. The results show that both methods have the capability of deriving electron positions that can optimize the cooling rate. Moreover, BO takes fewer samples to converge than the Q-learning method. In the second part, Bayesian optimization is further trained on the historical cooling data. In the new samples generated by the BO, the percentage of larger cooling rates data is greatly enhanced compared with the original historical data.

INTRODUCTION

The main purpose [1] of the Low Energy RHIC electron Cooler (LEReC) is to increase a collision rate at RHIC. Since its commissioning, LEReC has increased the event rate at RHIC significantly.

LEReC is the world's first RF-based electron cooler. The layout is schematically shown in Fig. 1. After exiting the 400 keV gun the electron bunches are accelerated to 1.6 – 2 MeV (depending on the energy of the cooled ions) in a superconducting RF cavity. The electrons are transported first to the cooling section in the "Yellow" RHIC ring and then, passing the 180 degree bend, to the CS in the "Blue" RHIC ring, thus cooling the ion bunches in both rings of the collider. Finally, the electron beam is extracted from the Blue CS and dumped in the beam dump. The electron bunches are repeated with 704 MHz frequency and are "packed" into 30 – 36 bunch macrobunches repeated with 9 MHz frequency, which matches the frequency of the RHIC ion bunches. Hence, on its passage through the CS each ion bunch is interacting with the electron macrobunch containing 30 electron bunches.

The main goal of this work is to quantitatively analyze the cooling performance in the LEReC system, and with the rapidly growing applications of Machine Learning (ML), explore possible ways to improve the cooling rate (make the

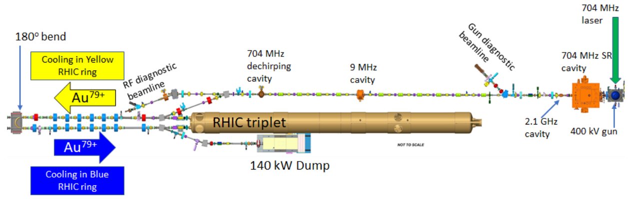

Figure 1: LEReC system layout (not to scale).

cooling process faster). To avoid interrupting the normal operations of the real system, a LEReC simulator developed locally [2] is used for the simulation.

Specially, Bayesian Optimization (BO) and Q-learning are used.

Bayesian optimization is mostly used to solve black box optimization. In such cases, the form of the objective function *f* is unknown and it is usually very expensive to sample from *f*. BO method can optimize the objective function while minimizing the number of samples drawn from *f*.

Bayesian optimization has been applied in a broad variety of fields, such as engineering design, environment science, robotics and machine learning, financial market, etc. Some BO applications are summarized in [3].

Q-learning [4] is an off-policy reinforcement learning algorithm, which is usually used to solve problems formulated as Markov Decision Processes (MDPs). The goal of Q-learning is to develop a policy which guides the agents to collect as many rewards as possible over the entire course of the process.

Q-learning has a wide range of application fields, such as engineering, artificial intelligence, manufacturing, marketing and advertising, etc. However, the nature of the standard Q-learning (using a Q table) limits the problem dimension it can apply to. Another famous variant of Q-learning called Deep Q-learning addresses this problem by replacing the Q-table with a neural network [5, 6].

SIMULATION EVALUATIONS

In this section, we present the simulation results of the cooling rate optimization experiments. We conduct the simulation under some common assumptions.

- Since this work only considers how electron positions affect the cooling rate, all other properties' parameters that could potentially affect the cooling rate are kept fixed, such as beam energy, number of particles, magnet strength, etc.;
- Ion beam is assumed to be centered with respect to the cooling section;

* Work supported by Brookhaven Science Associates, LLC under Contract No. DE-SC0012704 with the U.S. Department of Energy.
† ygao@bnl.gov

ISBN: 978-3-95450-214-1 ISSN: 2673-5490 doi:10.18429/JACoW-IPAC2021-WEPAB306

- To avoid disrupting the normal operations of the real system, a system simulator is used to output the cooling rate[1] when given a set of electron positions.

Here the cooling rate is defined as how quickly the ion beam's emittance changes. A negative rate means the emittance is decreasing, and vise versa. Hence the goal of the experiment is to achieve the largest negative rate.

Simulation Scenario

This simulation scenario considers a realistic case in which the electrons travel the same way as in the real system, as shown in Fig. 2. There are 8 Beam Position Monitors (BPMs) along the cooling section. Each BPM reports 2D data points - x and y coordinates of the electron beam. As stated in the common assumptions, only y-coordinates is considered and has a range of $[-3, 3]$ mm. The corresponding x-coordinates are set to 0. Ions are always in the center positions ($x = 0, y = 0$).

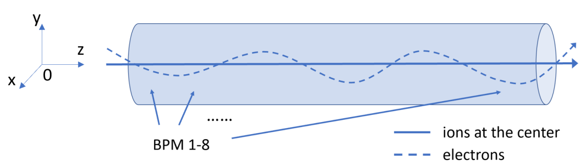

Figure 2: Scenario 2 illustration.

The goal in this scenario is to find out what features of the electrons' positions tend to produce better cooling results. Specifically, we study the Root Mean Square (rms) and the Standard Deviation (std) of the electron positions[2].

First, the Bayesian model is trained on 2100 random data points[3]. After training, the Bayesian model is used to sample 10 points. The result is shown in Fig. 3. The training data are plotted in light blue, and the Bayesian samples in red. As we can see, the Bayesian samples have an obviously better average cooling rate value. It indicates that the model learns a right direction to optimize the cooling rate.

Next, we analyze further the rms and std of the electron positions samples. For each input sample, the rms and std of the 8 BPM positions are calculated and plotted with the corresponding cooling rate. The result is shown in Fig. 4. The top figure shows the rms value distribution, and the bottom one shows the std value distribution. The random historical samples are plotted in light blue and Bayesian samples are in red. As we can see, the historical data do not have any obvious pattern, randomly evenly distributed. On the other hand, the Bayesian samples have a clear pattern that most of their rms and std values are close to 0. It implies that we have chosen the right features for the model to learn. It indicates that an electron trajectory which is closer (smaller rms) to the ion beam and has less variations (smaller std) tends to generate better cooling results, which matches our expectation.

The Q-learning process exhibits a similar behavior. As shown in Fig. 5, it starts at a random state and gradually reduces both the rms and std values to optimize the cooling result.

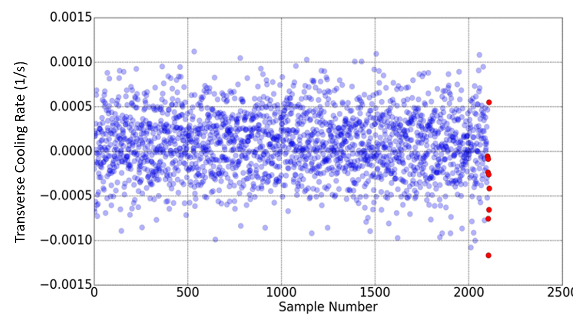

Figure 3: Data distribution of both training samples and Bayesian samples.

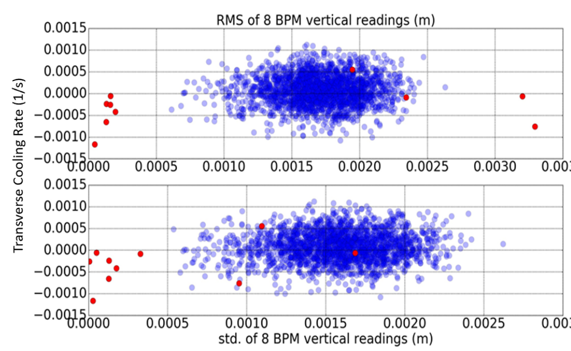

Figure 4: Comparison of two features abstracted from the BPM positions.

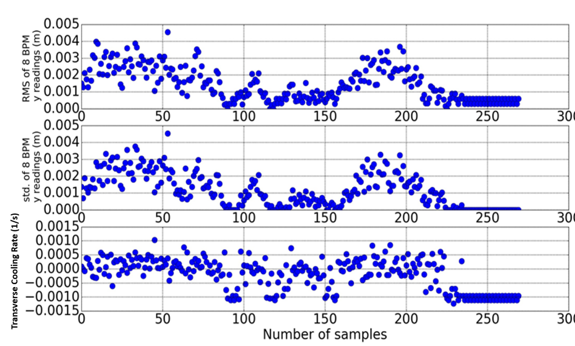

Figure 5: Q-learning simulation results of scenario 2.

EXPERIMENTS ON THE REAL DATA

From the above simulation results, we can see that both the Bayesian method and the Q-learning algorithm optimize the cooling rate. Moreover, the Bayesian method runs faster,

[1] Since the transverse plane and the longitudinal plane are considered symmetric, optimizing the cooling rate in one plane entails optimization in the other. Without loss of generality, in this simulation we choose the transverse plane cooling rate as the optimization objective.

[2] The rms value measures in average how far away the electron trajectory is from the ions'. The std value measures how much the electron trajectory varies during one cooling co-traveling with the ions. Those two features combined can basically summarize any particular electron trajectory, and hence are used for analysis.

[3] More analyses are needed to decide the lower bound of the training data size.

ISBN: 978-3-95450-214-1 ISSN: 2673-5490 doi:10.18429/JACoW-IPAC2021-WEPAB306

which just needs 10 samples to converge whereas the Q-learning takes hundreds of steps.

In this part, we further validate the Bayesian method's performance by optimizing the cooling data from the real system.

Real Cooling Data

Historical cooling data are extracted from the logging system, as shown in Fig. 6. The x-axis is the calendar time, and the y-axis is the transverse ion beam size. The increase in beam size in the middle corresponds to dumping the ion beam and refilling RHIC with new ion beam. Hence the figure shows cooling data over two stores. As we can see, the ion beam size is decreasing during each, hence the cooling is working.

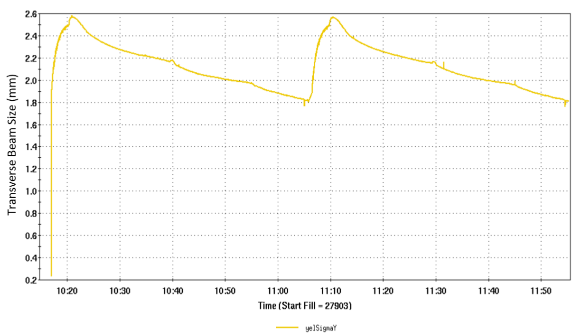

Figure 6: Historical cooling data from two stores.

Both during an individual store and store-to-store, numerous RHIC and LEReC parameters are changing, such as the ion bunch intensity, magnet settings, etc. To focus only on the aspect of trajectory alignment, only a short period of data are actually used for training the Bayesian model so that the beam conditions stay the same during that period. Here we take 100 seconds worth of data (100 samples) right after the cooling starts as the inputs for the Bayesian algorithm.

The changing rate of the transverse beam size is used as an indicator of how good the cooling is, and is calculated as $\lambda = (1/\delta)(d\delta/dt)$. The calculated changing rates[4] of the short period of data and its distribution are shown in Fig. 7. As we can see, the data resembles a Gaussian distribution with a normalized mean of 0.

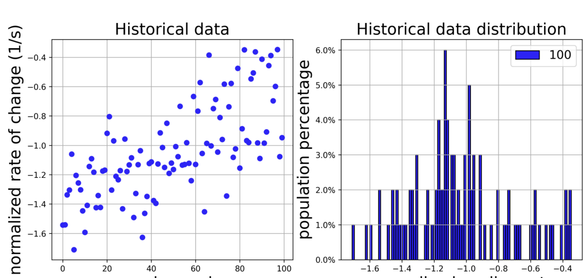

Figure 7: Changing rates distribution of the short period of cooling data.

[4] The actual changing rate values have been normalized using the *z-score* normalization, but that does not affect the shape of the data.

Build the System Approximation Model

A neural network is used as an approximation model of the system. Section shows that two features are very related to optimizing the cooling rates, i.e. the root mean square and the standard deviation of the electron positions. Thus, the neural network takes those two features (both in horizontal and vertical directions) as the inputs, and outputs the corresponding normalized cooling rate.

After trying several combinations of the network hyper-parameters[5] and picking the one with the least validation loss, the final network structure contains two hidden layers, with 10 nodes in the first layer and 6 nodes in the second. It is then trained on a historical cooling data set with size 500.

Simulation Results Bayesian optimization is used to generate 100 samples, and the sample distribution is shown in Fig. 8. As we can see from the second plot, the number of samples (in percentage) with larger cooling rates is greatly improved, thus validating the effectiveness of the procedure. The mean value of the Bayesian samples is -1.65.

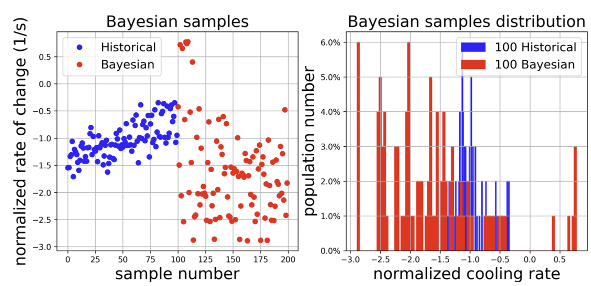

Figure 8: Population distribution of the Bayesian samples.

From the first plot we can see that the new samples tend to converge to a lower bound of the cooling rates computed by the Bayesian method, but the margin is wide. That partially results from the low accuracy of the neural network model. Adding more training data and including more variables (e.g. ion intensity, electron current and energy, etc.) to the model could help to further improve the result.

CONCLUSIONS

In this work, we examine the potential applications of machine learning techniques in optimizing cooling rates in the LEReC system. In the first part, simulations are conducted to demonstrate that both the Bayesian method and the Q-learning algorithm can optimize the cooling rate, and the Bayesian method converges faster. In the second part, the Bayesian method is further tested by using real system cooling data. The result shows that the method effectively improves the number of samples with larger cooling rates, which further validates its performance. This work also serves the purpose of experimenting with machine learning techniques in order to prepare for the transition from simulation to implementation in the live LEReC system.

[5] Such as the number of layers and neurons, the loss function, the optimizer and learning rate, etc.

ISBN: 978-3-95450-214-1 ISSN: 2673-5490 doi:10.18429/JACoW-IPAC2021-WEPAB306

REFERENCES

[1] K. M. Walsh, "Electron Bunches Keep Ions Cool at RHIC", `https://www.bnl.gov/newsroom/news.php?a=215585`

[2] H. Zhao, M. Blaskiewicz, A. V. Fedotov, *et al.*, "Cooling simulation and experimental benchmarking for an rf-based electron cooler", *Phys. Rev. Accel. Beams*, vol. 23, no. 7, p. 074201, Jul. 2020. `doi:10.1103/PhysRevAccelBeams.23.074201`

[3] B. Shahriari, K. Swersky, Z. Wang, *et al.*, "Taking the Human Out of the Loop: A Review of Bayesian Optimization", *Proceedings of the IEEE*, vol. 104, no. 1, p. 148-175, 2016. `doi:10.1109/JPROC.2015.2494218`

[4] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, Cambridge, MA, USA: The MIT Press, 2018.

[5] V. Mnih, K. Kavukcuoglu, D. Silver, *et al.*, "Playing Atari with Deep Reinforcement Learning", 2013. `arXiv:1312.5602`

[6] V. Mnih, K. Kavukcuoglu, D. Silver, *et al.*, "Human-level control through deep reinforcement learning", *Nature*, vol. 518, p. 529-533, Feb. 2015. `doi:10.1038/nature14236`