ISBN: 978-3-95450-227-1 ISSN: 2673-5490 doi:10.18429/JACoW-IPAC2022-TUPOMS037

RCDS-S: AN OPTIMIZATION METHOD TO COMPENSATE ACCELERATOR PERFORMANCE DRIFTS*

Z. Zhang, M. Song[1], X. Huang†, SLAC National Accelerator Laboratory, Menlo Park, CA, USA
[1]also at Department of Physics, Illinois Institute of Technology, Chicago, IL, USA

Abstract

We propose an optimization algorithm, Safe Robust Conjugate Direction Search (RCDS-S), which can perform accelerator tuning while keeping the machine performance within a designated safe envelope. The algorithm builds probability models of the objective function using Lipschitz continuity of the function as well as characteristics of the drifts and applies to the selection of trial solutions to ensure the machine operates safely during tuning. The algorithm can run during normal user operation constantly, or periodically, to compensate the performance drifts. Simulation and online tests have been done to validate the performance of the algorithm.

INTRODUCTION

Online optimization is an effective approach to find accelerator settings with high performance. Efficient optimization algorithms are key to online optimization. Popular optimization algorithms for online accelerator applications include Nelder-Mead simplex [1], robust conjugate direction search (RCDS) [2], particle swarm [3], and Bayesian optimization [4]. During an optimization run, as the algorithm gradually discovers machine settings with high performance, it can also produce solutions with poor performance, which cannot be tolerated for normal user operation. Therefore, online optimization is usually performed during dedicated machine development or study shifts. However, in many cases, an ideal machine setting will not maintain the high performance during the long period of user operation. Small variations in the accelerator components, caused by or coupled with variations of the surrounding environment, can cause the machine performance to drift with time.

In this study, we propose a safe tuning method that can be used during user operation. The new algorithm is called safe robust conjugate direction search (RCDS-S). It employs iterative one-dimensional (1-D) optimization over a conjugate direction set in a similar manner as the RCDS method. However, its 1-D optimization is done by a more prudent and informed fashion, which employs a probability model of the objective function to assess the risk of exceeding a safety threshold by the trial solution.

THE RCDS-S METHOD

Our goal of the study is to develop an optimization method that can be used to optimize accelerator performance during user operation by keeping the performance above a certain threshold. Such a method could be termed "safe" optimization algorithm. A safe optimization algorithm could be used to compensate the performance drift with time, as it can run in the background continuously, periodically, or as needed.

In the following, we first discuss the uncertainty of the objective function as it is probed. By constructing a probability model of the uncertainty and using it to guide the selection of new trial solutions, we devised a safe 1-D optimization method. Combining this safe 1-D optimization method and the conjugate direction search method, we arrived at the new algorithm, RCDS-S.

Modeling Uncertainty of Objective Function

In this study, we assume the optimization problem to be a minimization problem, with measurement error (noise) and time-dependent error (systematic drift). Firstly the gradient of the objective function has to be limited for a safety setting. We assume the objective function to be L-Lipschitz continuous, which means for any $\mathbf{x}, \mathbf{x_0} \in \mathbf{D}$, where $\mathbf{D}$ is the domain of the function, we have,

$$\|f(\mathbf{x}) - f(\mathbf{x_0})\| \leq L \cdot \|\mathbf{x} - \mathbf{x_0}\| .$$

On the other hand, without further information about the specific optimization problem, the drift can be modeled as a random walk process. Under this assumption, the uncertainty of the measurement becomes a time varying random variable,

$$y = f(\mathbf{x}) + \epsilon(t),$$

where $\epsilon(t) \sim N(0, \sigma_n^2 + t\sigma_d^2)$. Here σ_n is the noise level, σ_d^2 represents the increase of the variance within a unit time interval, and t the time elapsed from a reference point.

Given the safety threshold h, to guarantee $y \leq h$ we have:

$$\hat{\epsilon} \leq \frac{h - E_{\max}}{\sqrt{2\sigma_n^2 + t \cdot \sigma_d^2}}, \tag{1}$$

here $E_{\max}$ is the maximum expected value of objective y at point $\mathbf{x}$ which satisfies $E_{\max}(\mathbf{x}) = y_0 + L \cdot \|\mathbf{x} - \mathbf{x_0}\|$, and $\hat{\epsilon} \sim N(0, 1)$.

1D Safety Exploration

For a 1D problem, given a few observations, the safety probability of each candidate can be computed with Eq. (1). The idea is that each observation would provide safety information for all the other candidates along the direction, based on the relative measurement time and position with regard to the candidate of interest, and the final safety probability of one candidate can be determined by combining the safety information from all the observations. One example safety

* Work supported under DOE Contract No. DE-AC02-76SF00515
† xiahuang@slac.stanford.edu

ISBN: 978-3-95450-227-1 ISSN: 2673-5490 doi:10.18429/JACoW-IPAC2022-TUPOMS037

probability distribution along a normalized 1D domain is shown in Fig. 1.

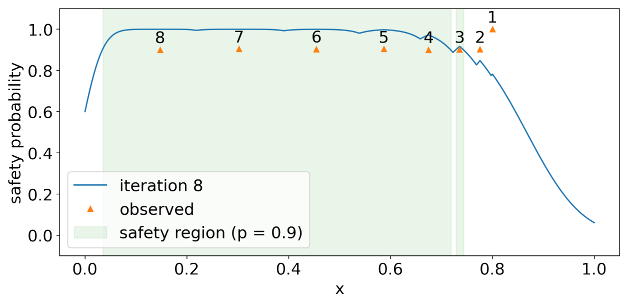

Figure 1: Snapshot of the safety probability along the viable range at the end of 1-D safety exploration. Safety probability curve is calculated with the 8 given observations. The order of sampling is indicated by the number on top of each observation. Orange triangles denote the observations, blue curve shows the calculated safety probability, the safety region with safety probability threshold $p_s = 0.9$ is highlighted in green.

Once the safety probability is available for each candidate, the exploration algorithm would pick one that is safe enough (i.e. the safety probability exceeds a specific threshold, such as $p_s = 0.9$) and could extend the explored region the most, as the next solution to sample.

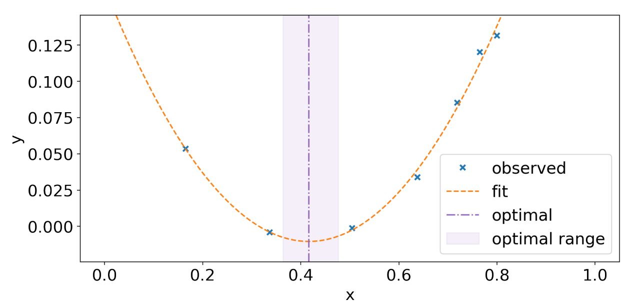

Figure 2: Parabola fitting at the end of each acquisition. The purple dashed vertical line indicates the peak position found by the fitting, the purple region around the peak line shows the uncertainty of the peak position, calculated by the covariance matrix of the parabola fitting.

After evaluating the new sampled solution, a parabola fitting would be performed with the current observations to locate the optimal solution, as demonstrated in Fig. 2. If the fitting could bracket the optimal position with relatively small uncertainty, the safety exploration is terminated and the optimal solution would be reported, otherwise the procedure above is repeated until: 1) the parabola fitting succeeds, or 2) no more safety candidates are available, or 3) maximum attempts exhausted. The process of a complete safety exploration is visualized in Fig. 3.

The RCDS-S Algorithm

Combining the scheme of iterative 1-D optimization over conjugate directions [5], the safety 1-D exploration discussed

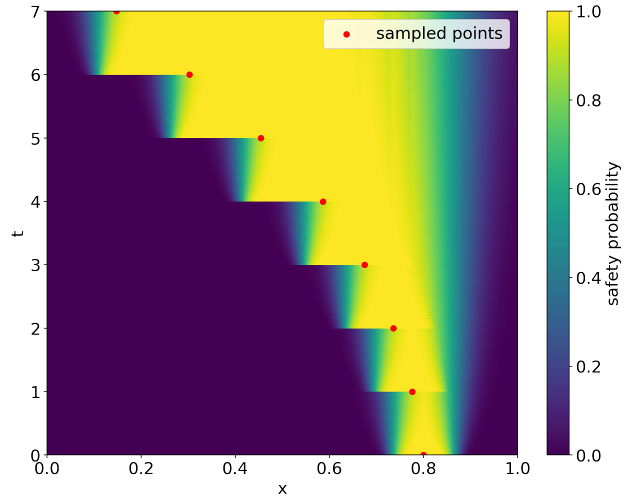

Figure 3: Evolution of the safety probability during a safety exploration on the drifting 1-D test problem. Red dot denotes the time and position of the corresponding sampled point. The safety probability is calculated with the Gaussian random walk drift model.

in previous sections and the use of parabolic fitting to determine the minimum [2], we arrive at the RCDS-S algorithm. In the algorithm implemenation, we choose to normalize all parameters to within the range of [0, 1]. Two most critical hyper-parameters in the modeling of the safety probability are the Lipschitz constant L_v in normalized decision space and the drift rate σ_d in time. Choosing good values for these two parameters is crucial to the safety performance of the proposed algorithm.

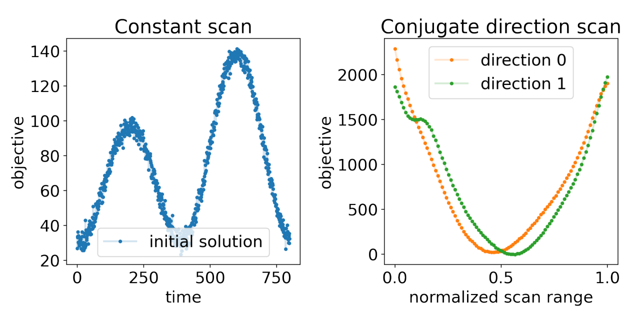

Figure 4: Determination of the hyper-parameters for the kicker bump matching test problem. Left: variation of objective function over time; right: objective function vs. parameter scan in two directions.

The hyper-parameters can be obtained before applying the algorithm to a specific problem by analyzing historic data or performing additional measurements. Examples of the parameter-determination scans are shown in Fig. 4.

TEST RESULTS

Simulation and experimental studies were conducted to test the performance of the RCDS-S algorithm in online optimizations for drifting problems. The test problem is

ISBN: 978-3-95450-227-1 ISSN: 2673-5490 doi:10.18429/JACoW-IPAC2022-TUPOMS037

kicker-bump matching for a storage ring, with a 2-D decision space. In the test, the strength of one kicker, K1, is modulated in a sinusoidal form to simulate the systematic drift, as shown in

$$v(t) = v_0 + \sigma_d \left[\sin\left(\frac{2\pi}{p}(t + t_0)\right) - \sin\left(\frac{2\pi}{p} t_0\right) \right], \quad (2)$$

where p is the drifting period, σ_d the drifting amplitude, t_0 the time origin, v_0 the initial value of the modulated variable.

The strengths of the other two kickers, K2 and K3, are used as tuning knobs of the optimization problem. The rms turn-by-turn horizontal orbit from 256 turns of residual oscillations is used as the objective function.

Simulation Test

For the simulation tests, we set $p = 800$, $\sigma_n = 3\,\mu\text{m}$, $\sigma_d = 0.1\,\mu\text{m}$ per evaluation period. Based on the two pre-scans discussed in the last section, the Lipschitz constant L and strength of Gaussian random walk σ_g are chosen to be 2000 and 0.2, respectively. The safety threshold h can be varied to change the safety search difficulty. In the tests, the safety threshold is set to 40 µm, which is only slightly higher than most of the observed values of the noisy objective at the initial solution.

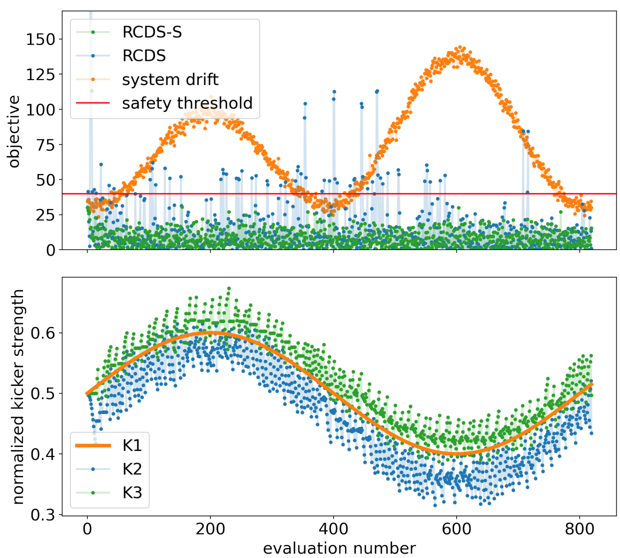

Figure 5: RCDS-S for kicker-bump matching in simulation. Orange dots in the top plot show the objective drift caused by the modulated kicker strength at the initial solution. The modulation on K1 along with the evolution history of K2 and K3 are shown in the bottom plot.

The tests have been run multiple times, and the performance is stable. A typical test is shown in Fig. 5. The top plot compares the objective function over one modulation period for three cases: no optimization, tuning with RCDS, and tuning with RCDS-S. The test result shows that RCDS-S is able to follow the drift and seek the optimum while keeping the objectives of the trial solutions well below the safety threshold. RCDS is also very efficient for the test problem. However, since it is not aware of the safety threshold, the proposed solutions are not guaranteed to be safe.

Experimental Test

For the experimental tests, the voltage amplitude of one kicker (K1) is modulated with a period of 800 data points, with an interval of 2 seconds between data points. The noise level was measured right before the algorithm test and was found to be 5 µm. The initial solution is the same as the operation setting which would give the best objective if the system is not drifting.

Two rounds of tests were performed with the experimental setup. For the first round, the safety threshold is set to 60 µm. As it succeeded, we set the safety threshold to 50 µm for the second round. The results for the second round are shown in Fig. 6, where the objective function for RCDS-S for one modulation period is compared to the results of RCDS and the case without tuning.

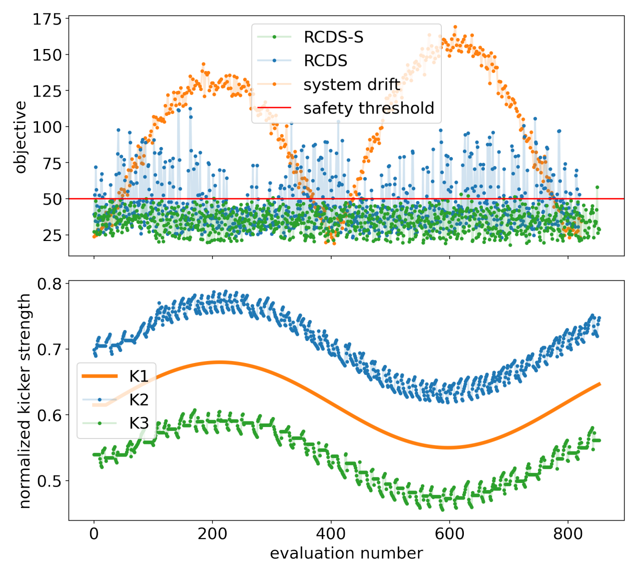

Figure 6: RCDS-S on the kicker-bump matching experiment on the SPEAR3 storage ring, with safety threshold 50 µm.

The experimental performance is similar to the simulation cases for both RCDS-S and RCDS. The objective function has a larger variance for the evaluated trail solutions in experiments. This could be due to the higher measurement noise level.

CONCLUSION

In this study, we propose an optimization algorithm (RCDS-S) that combines robust conjugate direction search and a new 1-D safety exploration algorithm to optimize noisy, drifting machine performances online, while keeping the machine performance within a designated safe envelope. The proposed algorithm has been successfully tested on simulated and experimental accelerator tuning problems.

ISBN: 978-3-95450-227-1 ISSN: 2673-5490 doi:10.18429/JACoW-IPAC2022-TUPOMS037

REFERENCES

[1] J. A. Nelder and R. Mead, "A simplex method for function minimization," *The Computer Journal*, vol. 7, no. 4, pp. 308–313, 1965, doi:10.1093/comjnl/7.4.308

[2] X. Huang, J. Corbett, J. Safranek, and J. Wu, "An algorithm for online optimization of accelerators," *Nucl. Instrum. Methods Phys. Res, Section A*, vol. 726, pp. 77–83, 2013, doi:https://doi.org/10.1016/j.nima.2013.05.046

[3] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95 - International Conference on Neural Networks*, vol. 4, 1995, 1942–1948 vol.4, doi:10.1109/ICNN.1995.488968

[4] J. Duris *et al.*, "Bayesian optimization of a free-electron laser," *Phys. Rev. Lett.*, vol. 124, p. 124 801, 12 2020, doi:10.1103/PhysRevLett.124.124801

[5] M. J. D. Powell, "An efficient method for finding the minimum of a function of several variables without calculating derivatives," *The Computer Journal*, vol. 7, no. 2, pp. 155–162, 1964, doi:10.1093/comjnl/7.2.155