ISBN: 978-3-95450-231-8 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2023-THPL007

ROBUST ADAPTIVE BAYESIAN OPTIMIZATION*

N. Kuklev†, Y. Sun, H. Shang, M. Borland, G. Fystro
Advanced Photon Source, Argonne National Laboratory, Lemont, IL, USA

Abstract

Particle accelerators require continuous adjustment to maintain beam quality. Several machine learning (ML) approaches are being explored for this task. At the Advanced Photon Source (APS), we have recently proposed the adaptive Bayesian optimization (ABO) algorithm and have shown it to be effective experimentally in the APS injector complex. Further testing has suggested several improvements, on which we report here. We introduce dynamic kernel switching, deep kernel learning, and surrogate model prior means, resulting in improved robustness. We also extend our code with multi-dimensional time kernel support and predictive constraint avoidance to make it applicable to a wider range of systems. These changes also improve the general ABO performance, but more importantly expand ABO applicability to systems with rapid or unexpected changes in either optimization parameters or time properties. Notably, this allows for rapid and automated fallback to conservative parameters when optimizer confidence degrades, with alarms raised for further operator review. These features will permit further operational ML adoption at APS.

INTRODUCTION

Particle accelerators face increasing performance demands, resulting in tighter tolerances on accuracy and stability [1]. Continuous parameter adjustment is typically required, often relying on expert guidance and intuition. With the rise of machine learning, there is immense interest in making use of newly-available algorithms to implement generic tools to improve reliability, reduce expert workload, and provide higher performance to users.

A key application of ML for accelerators is in parameter optimization, whereby one or multiple objectives are tuned through an intelligent search of the parameter space. Conventional optimization methods already in use include simplex [2, 3], RCDS [4], genetic algorithms [5], and extremum seeking [6]. New ML methods include Bayesian optimization (BO) [7], reinforcement learning [8], and others. BO is of special interest since it allows efficient black-box function optimization with few samples, taking advantage of any prior physics-model knowledge provided to the algorithm.

At APS, we are working on methods to make BO applicable to a wider range of experimental systems, including those with high-dimensionality, time-dependent drift, or high noise, all without extensive expert tuning. One of our recent algorithms, ABO, has shown good performance in several such cases. This paper discusses several new features that overall help make ABO a more configurable and robust method that can detect, predict, and avoid problematic configurations, with a graceful failover in the worst cases.

* The work is supported by the U.S. Department of Energy, Office of Science, Office of Basic Energy Sciences, under Contract No. DE-AC02-06CH11357.
† nkuklev@anl.gov

ADAPTIVE BAYESIAN OPTIMIZATION

This section briefly reviews the ABO concept as presented in Ref. [9]. The output being optimized is described by

$$\mathbf{y} = f(\mathbf{t}, \mathbf{x}) + \varepsilon, \tag{1}$$

where $f(\mathbf{x})$ is the black-box function of interest and $\varepsilon \sim \mathcal{N}(0, \sigma_\varepsilon^2)$ the added noise. Using Gaussian Processes (GP) [10], a surrogate model for f can be parameterized as a multivariate normal distribution with a mean $m(\mathbf{x})$ and covariance kernel $k(\mathbf{x}, \mathbf{x}')$ as

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')). \tag{2}$$

The kernel is used to evaluate the similarity between values of f at $\mathbf{x}$ and $\mathbf{x}'$, and its' appropriate choice is critical for good GP convergence. In ABO, the black-box function is presumed to depend not only on input parameters $\mathbf{x}$ but also on time and potentially other variables contained within 'auxiliary' vector $\mathbf{t}$. The value of $\mathbf{t}$ is dictated by outside factors and cannot be controlled by the optimizer. Time is a typical fixed parameter, but others are possible - room temperature, upstream accelerator beam parameters, etc. In ABO, standard input dimensions are typically assigned one of the common stationary 'local' kernels, such as the square exponential (SE) kernel:

$$k_{SE,i} = \sigma^2 \exp\left(\frac{-(x_i - x_i')^2}{2l^2}\right). \tag{3}$$

Kernel hyper-parameters are output variance σ and length-scale l. For auxiliary variables however, correlations can have unusual patterns like non-stationary oscillations or irregular motion. Previous studies found that standard local kernels were beneficial for irregular signals, while globally adaptive spectral mixture (SM) [11] or deep (i.e. neural network) kernels were useful to take advantage of long-range patterns. To form the final GP kernel, individual sub-kernels are multiplied, which is equivalent to a logical 'AND':

$$k = k_{x(i)}(x_i, x_i') \times ... \times k_{t(j)}(t_j, t_j'). \tag{4}$$

Once the ABO model is created, the process of choosing the next point is similar to standard BO, except that all auxiliary variables are held fixed at expected values during acquisition function optimization. For example, the time parameter would be set to the time of next measurement, including any lag in setting devices.

14th International Particle Accelerator Conference,Venice, Italy JACoW Publishing
ISBN: 978-3-95450-231-8 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2023-THPL007

KERNEL IMPROVEMENTS

Kernel Switching

The behavior of drifting systems can change abruptly, making a pre-trained time-aware model inaccurate and potentially worse than one with no historical data, at least until the new system behavior is learned sufficiently. We have previously proposed SM kernels as a universal choice suitable for signals with both local correlations and long-range patterns. This flexibility came at the cost of a higher parameter count and thus longer training requirements. When a SM model is exposed to rapid changes, reconvergence can be slow and large exploration steps can occur. While such steps may give optimal information gain, output swings can be disruptive to operational systems and need to be avoided.

To mitigate this issue, we implemented a heuristic to dynamically switch to more robust and conservative local models until the system stabilizes, trading off convergence speed for better worst-case performance. GP models provide an elegant way to detect changes through kernel lengthscales, l in Eq. (3), which can be thought of as the ranges within which data is locally correlated [12]. Rapid changes in lengthscales can thus indicate high model uncertainty. Similar logic can be applied to output variance and other hyperparameters.

Several implementation details are critical. First, the concept of lengthscale as a single parameter does not exist when fully Bayesian priors are used. Instead, lengthscales are defined through prior distributions, such as the Gamma distribution. $\Gamma(\alpha, \beta) \equiv \text{Gamma}(\alpha, \beta)$. Expected values of lengthscale are used for such parameters. Second, with automatic relevance determination, each dimension has its own lengthscale and thus no single threshold can be defined. We considered both vector-based similarity metrics (euclidean or cosine distance) and aggregated metrics (mean and median relative change). Median relative change r was chosen since it is a robust statistic and showed good performance:

$$r = M\left[\frac{\Delta l/\bar{l}}{\Delta t}\right], \tag{5}$$

with Δt the application-dependent time window, typically 10-300s. Only lengthscale decreases are typically considered.

In Fig. 1, a simple trajectory simulation is used to demonstrate the dynamic switching when a system experiences a sudden change due to a ramp from one optimum location to another.

Multi-Dimensional Auxiliary Spaces

While time is correlated with all drift signals and can thus be used as the only auxiliary variable, the direct relationship between the disturbance and the objective can often be represented with a simpler function. For example, variation in stored beam current might be hard to model as a function of time but its effect on lifetime is well understood and easier to fit. Thus, if drifts can be attributed to just a few parameters, it is advantageous to use them directly. To support such cases, we extended ABO to support multi-dimensional auxiliary spaces. In the future, we plan to add the ability to turn on/off dimensions and suggest to the user which auxiliary variables have the largest impact on optimization quality.

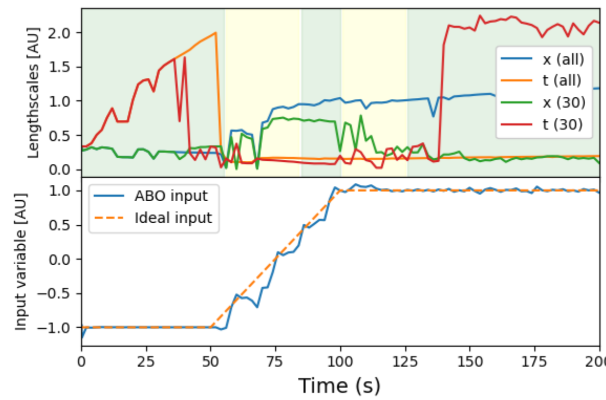

Figure 1: Kernel switching demonstration, with yellow regions corresponding to a safe SE kernel and green to an SM kernel. Both 30 point window and full dataset parameters are plotted, with the latter being less useful in detecting changes. Note how only decreases in lengthscales trigger kernel swaps.

Deep Kernels and Prior Means

As described above, GP models estimate the black-box function through mean and covariance. Both of these components are functions and can be replaced with different heuristics. Doing so with neural networks that are trained on historical or live data is referred to as deep kernel learning [13], and we have implemented support for it in ABO. However, many of our injector models were found to lack quantitative fidelity and could not be used for kernel training. As such, we focused more effort on prior means. Previously, only constant prior mean functions were used, with the value either learned as part of the GP fitting process or fixed to a pessimistic value (to discourage exploration). ABO performance can be improved if the auxiliary-space prior mean can be biased to a region where most outputs are expected. We added a feature to automatically create prior means with either simple distributions or neural-network surrogates based on historical data. Even when no previous data is available, we found it useful to define a broad uniform prior over maximum expected range to limit spurious exploration, which can be thought of as a 'soft' trust region.

CONSTRAINT PREDICTION AND AVOIDANCE

ABO supports constraints by using additional GP models that apply weights to candidate points based on predicted feasibility [14]. Initial implementation was limited to only checking the next step, but time-aware GP models can be used for long range forecasting. To exploit this knowledge, we added a constraint avoidance procedure that can take preventive action several steps ahead of a likely violation.

ISBN: 978-3-95450-231-8 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2023-THPL007

It works by 'fantasizing' (drawing from model posterior) several sequences of steps and detecting any that could potentially result in violations. In such cases, the acquisition function is further biased to avoid problematic regions.

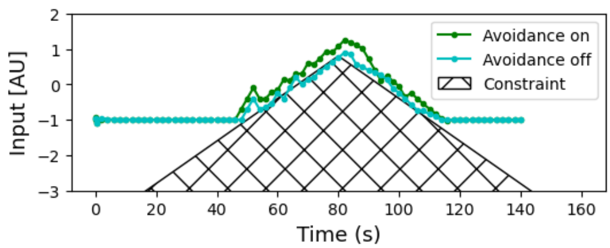

Figure 2: Constraint avoidance test with a triangular keepout area. Optimum is kept stationary at -1, forcing inputs to deviate away to satisfy constraints.

Figure 2 visualizes the paths taken with and without the avoidance feature, with the latter deviating more smoothly from the naive optimum ahead of actual constraint violation. However, it also exhibits slight lag at the peak as a tradeoff for the more conservative strategy. Note that both cases are technically safe — when violation is inevitable on the next step, algorithms can be configured to abort. We are exploring more flexible ways to distinguish between soft and hard constraints, and how to apply them with different avoidance policies.

EXPERIMENTAL TESTS

We have tested several combinations of new ABO options in the APS linac [15]. We chose beam size and trajectory stabilization tasks in the L1 and L3 sections, since these objectives were highly reproducible and accurately measured. Absent significant natural drift, we simulated drifts by changing upstream quadrupoles and correctors in various patterns. ABO had no knowledge of these changes, and had to respond purely based on the objective value. In Figure 3, a trajectory test is shown with a customized prior mean based on training data earlier in the shift. Data indicates highly regular ABO behavior both in the standard case and during anomalies like hitting optimization boundaries or sudden system changes.

A second test was performed with two input dimensions and kernel switching enabled to test a common case of pausing/resuming optimization after some system changes. Results are shown in Figure 4, and demonstrate that after resuming only a small interval was necessary to stabilize the algorithm and switch to a more complex kernel.

CONCLUSION

In order to become true operational tools, ML algorithms must be robust and flexible. We presented several improvement to one such algorithm, ABO — dynamic kernels, smart priors, and other changes. They are expected to improve ABO robustness and expand its applicability to more demanding systems. Experimental results using the new features have shown promising improvements in stability and

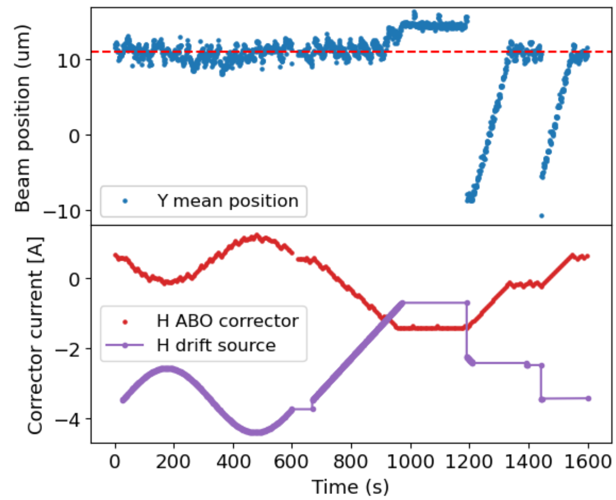

Figure 3: 1D ABO test with a prior mean. Beam was first oscillated to confirm functionality (left), then ramped past optimizer variable limit and suddenly changed back (right).

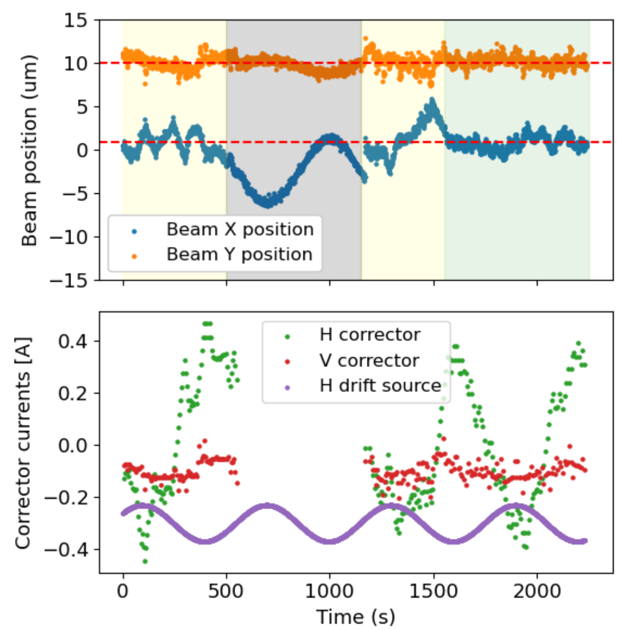

Figure 4: 2D ABO test with kernel switching. Background denotes safe local kernel (yellow), paused ABO (gray), and full SM kernel (green). Red lines show position setpoints.

performance. Much work remains in automating the various configuration choices that for now require an expert. For future work, we want to focus on an operational implementation as a control room tool and also to make a collection of machine-specific bespoke kernel/prior neural network surrogates that are automatically retrained on live data.

ACKNOWLEDGMENTS

Authors acknowledge ANL Laboratory Computing Resource Center (LCRC) for providing the computational resources, and APS operators for help in data collection.

ISBN: 978-3-95450-231-8 ISSN: 2673-5490 doi: 10.18429/JACoW-IPAC2023-THPL007

REFERENCES

[1] A. Scheinker, "Adaptive machine learning for time-varying systems: Low dimensional latent space tuning," *J. Instrum.*, vol. 16, no. 10, P10008, 2021. doi:10.1088/1748-0221/16/10/P10008

[2] L. Emery, M. Borland, and H. Shang, "Use of a General-Purpose Optimization Module in Accelerator Control," in *Proc. PAC'03*, Portland, OR, USA, May 2003. https://jacow.org/p03/papers/WPPE002.pdf

[3] X. Huang, "Robust simplex algorithm for online optimization," *Phys. Rev. Accel. Beams*, vol. 21, p. 104 601, 2018. doi:10.1103/PhysRevAccelBeams.21.104601

[4] X. Huang, J. Corbett, J. Safranek, and J. Wu, "An algorithm for online optimization of accelerators," *Nucl. Instrum. Methods Phys. Res., Sect. A*, vol. 726, 2013. doi:10.1016/j.nima.2013.05.046

[5] M. Borland, V. Sajaev, L. Emery, and A. Xiao, "Multi-objective direct optimization of dynamic acceptance and lifetime for potential upgrades of the Advanced Photon Source," ANL/APS/LS-319, 986303, 2010, ANL/APS/LS–319, 986 303. doi:10.2172/986303

[6] A. Scheinker, "Adaptive Machine Learning for Robust Diagnostics and Control of Time-Varying Particle Accelerator Components and Beams," *Information*, vol. 12, no. 4, p. 161, 2021. doi:10.3390/info12040161

[7] M. Balandat *et al.*, "BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization," in *Advances in Neural Information Processing Systems*, vol. 33, 2020, pp. 21 524–21 538. https://proceedings.neurips.cc/paper/2020/hash/f5b1b89d98b7286673128a5fb112cb9a-Abstract.html

[8] V. Francois-Lavet, P. Henderson, R. Islam, M. G. Bellemare, and J. Pineau, "An Introduction to Deep Reinforcement Learning," *Foundations and Trends® in Machine Learning*, vol. 11, no. 3-4, pp. 219–354, 2018. doi:10.1561/2200000071

[9] N. Kuklev, M. Borland, G. I. Fystro, H. Shang, and Y. Sun, "Online Accelerator Tuning with Adaptive Bayesian Optimization," in *Proc. NAPAC'22*, Albuquerque, New Mexico, USA, 2022, p. 842. https://jacow.org/napac2022/papers/THXD4.pdf

[10] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. 23, 2005. doi:10.7551/mitpress/3206.001.0001

[11] A. G. Wilson and R. P. Adams, *Gaussian Process Kernels for Pattern Discovery and Extrapolation*, 31, 2013. doi:10.48550/arXiv.1302.4245

[12] F. M. Nyikosa, M. A. Osborne, and S. J. Roberts, *Bayesian optimization for dynamic problems*, 2018. doi:10.48550/arXiv.1803.03432

[13] A. G. Wilson, Z. Hu, R. Salakhutdinov, and E. P. Xing, *Deep Kernel Learning*, 6, 2015. doi:10.48550/arXiv.1511.02222

[14] D. Eriksson and M. Poloczek, *Scalable Constrained Bayesian Optimization*, 28, 2021. doi:10.48550/arXiv.2002.08526

[15] A. Nassiri, G. Mavrogenes, and W. Wesolowski, "The Linac Injector for the ANL 7 GeV Advanced Photon Source," in *Proc. LINAC'90*, Albuquerque, NM, USA, Sep. 1990, pp. 611–613. https://jacow.org/l90/papers/TH411.pdf