Supplemental Information for

Unlocking the Black Box beyond Bayesian Global Optimization for Materials Design using Reinforcement Learning

Yuehui Xian[a], Xiangdong Ding[a,*], Xue Jiang[b], Yumei Zhou[a], Jun Sun[a], Dezhen Xue[a,*], Turab Lookman[a,c,*]

[a] State Key Laboratory for Mechanical Behavior of Materials, Xi'an Jiaotong University, Xi'an 710049, China

[b] Beijing Center for Materials Genome, University of Science and Technology, Beijing, China

[c] AiMaterials Research LLC, Santa Fe, USA 87501

This PDF file includes 2 sections with Figures S1 and Tables S1.

The EI value problem

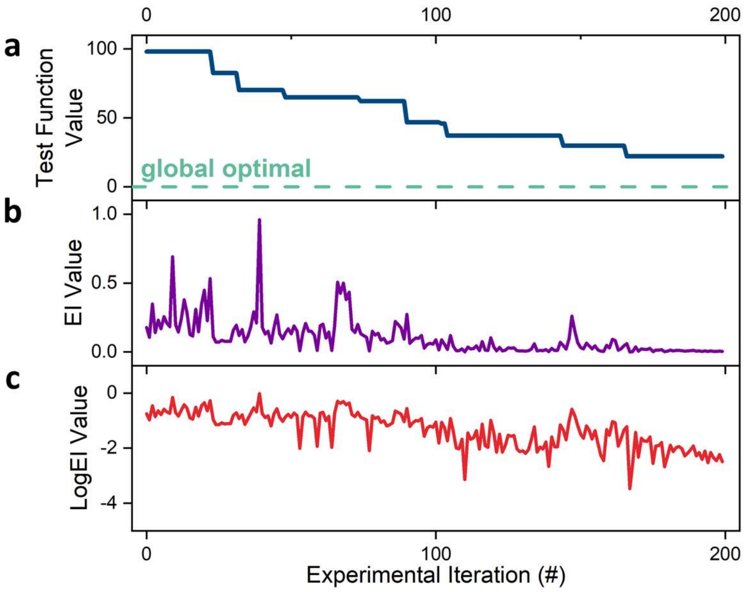

Figure S1. Decay of Expected Improvement values during Bayesian Optimization. Optimization performance using Bayesian Optimization with EI on the Rastrigin function, illustrating the decline of EI value over iterations: (a) Best-so-far values, (b) EI values and (c) logEI values as a function of experimental iterations.

Figure S1 illustrates the common issue encountered when using Bayesian Optimization with Expected Improvement (EI), where the EI value often approaches zero. Using an optimization curve for the Rastrigin function as an example, panel a) shows the test function value over 200 experimental iterations. In the early iterations, the EI value (panel b) remains within a normal range. However, as iterations progress, the EI value diminishes to nearly zero. Panel c) displays the logarithm of the EI value, further highlighting this decline. This phenomenon presents a significant challenge to applying EI as an acquisition function in black-box optimization, as the reduction in EI value limits the algorithm's effectiveness.

To investigate vanishing EI values commonly encountered in BO, we examined the effectiveness of using logarithmic Expected Improvement (logEI)[1] as an alternative acquisition function to address the issue of EI values tending to zero in later iterations (as shown in Figure S1). We conducted comparative experiments on the two test functions with dimensionality of 10 (D=10). For the Ackley function (Figure 3a), BO-logEI showed improved convergence compared to standard BO. However, for the Rastrigin function (Figure 3b), despite the implementation of logEI, both BO variants still struggled to match the performance of RL, suggesting that the challenges faced by BO in high-dimensional spaces extend beyond the vanishing EI problem, particularly in landscapes with multiple local optima. The RL approach outperformed both BO variants for the Rastrigin function, showing more robust exploration and better final optimization results. This superior performance can be attributed to RL's ability to learn and adapt its exploration strategy through experience, rather than relying solely on myopic acquisition function values. These findings further support that RL provides a more effective framework for high-dimensional optimization problems, particularly in complex landscapes where traditional BO approaches, even with modifications such as logEI, often face limitations.

The HEA test environment

To establish reliable ground truth models for evaluating optimization algorithms, we developed neural networks trained on experimental data from high-entropy alloys (HEAs). These networks serve as efficient substitutes for time-consuming experimental measurements during the optimization process evaluation. The neural networks were trained using 501 composition-property pairs for HEAs, focusing on yield strength (σ_{γ}), ultimate strength (σ_u), and elongation (ε) to balance strength and ductility. Data were sourced from both laboratory experiments and published literature. Following the training procedure outlined in[2], each dataset was randomly split into training (70%), validation (15%), and test (15%) subsets. Training was conducted for 500 epochs with a learning rate of 5×10^{-4}. Elemental features serve as part of the input into the neural networks to improve the predictive performance, with no gradient flow needed. Each network begins with a convolutional section followed by a residual connection, where process conditions are concatenated with outputs before entering the fully connected section. The convolutional section consists of two layers with a kernel size of $3\times N_{\mathrm{elem}}\times N_{\mathrm{feature}}$, with batch normalization applied after each layer to enhance robustness. The output is flattened and passed through two fully connected layers, configured as $(N_{\mathrm{elem}}\times N_{\mathrm{feature}})\times128$ and 128×1, incorporating Exponential Linear Units (ELUs) for nonlinear activation and a dropout layer for stability. The networks were trained using the Adam optimizer with a batch size of 16. These neural networks serve as ground truth models to evaluate the optimization performance of different methods, providing figures of merit (FOM) values that combine multiple properties into a single objective to optimize, reflecting practical material development priorities and providing a complex, multi-objective optimization landscape suitable for comparing the relative strengths of BO and on-the-fly DQN.

Table S1 Composition constraints and design space for HEAs

Element	Lower limit (at. ratio)	Upper limit (at. ratio)	Step (at. ratio)	Maximum number of dimensions[†]			
				4 dimensions	6 dimensions	8 dimensions	10 dimensions
C	0	0.06	0.001		✓	✓	✓
Al	0	0.16			✓	✓	✓
V	0	0.33		✓	✓	✓	✓
Cr	0	0.4		✓	✓	✓	✓
Mn	0	0.5		✓	✓	✓	✓
Fe	0	0.6		✓	✓	✓	✓
Co	0	0.5				✓	✓
Ni	0	0.6				✓	✓
Cu	0	0.36					✓
Mo	0	0.1					✓

[†] The actual optimization dimensionality is one less than the number of elements shown, as the atomic ratios must sum to 1.

For the on-the-fly DQN agent, we have implemented a three-layer fully connected architecture, employing an ε-greedy policy and experience replay mechanism to predict action values, as detailed in the Methods section.

Following the implementation of BoTorch[3], our BO here utilized the Matérn 2.5 kernel, with the LBFGS-2 gradient-based optimizer employed to maximize the Expected Improvement (EI) acquisition function in the inner loop. Each BO run started with 20 random initial data points for training of the surrogate. While on-the-fly DQN agents required a randomly initialized memory buffer of 300 state-action-reward-next state (s, a, r, s') transition tuples to facilitate effective learning, fulfilling the sample requirements characteristic of reinforcement learning algorithms. To ensure fair comparison, we conducted 64 independent runs for each method across all dimensionalities tested. Statistical significance was assessed using paired T-tests comparing the final FOM values achieved by different methods for the case of 10 components.

Citations

1. Ament, S., Daulton, S., Eriksson, D., Balandat, M. & Bakshy, E. Unexpected improvements to expected improvement for bayesian optimization. In *Proc. Advances in Neural Information Processing Systems* Vol. 36 (2023).
2. Wang, J., Kwon, H., Kim, H. S. & Lee, B.-J. A neural network model for high entropy alloy design. *npj Comput. Materials* **9**, 60 (2023).
3. Balandat, M. *et al.* BoTorch: A framework for efficient Monte-Carlo Bayesian optimization. In *Proc. Advances in Neural Information Processing Systems* Vol. 33 (2020).