Biological Sequence Design using Batched Bayesian Optimization

David Belanger[1], **Suhani Vora**[1], **Zelda Mariet**[1], **Ramya Deshpande**[2], **David Dohan**[1]
Christof Angermueller[1], **Kevin Murphy**[1], **Olivier Chapelle**[1], **Lucy Colwell**[1,3]
[1] Google Research, [2] Caltech (work done while interning at Google Research), [3] Cambridge University
{dbelanger,svora,zmariet,ddohan,christofa,kpmurphy,
chapelle,lcolwell}@google.com, rdeshpan@caltech.edu

Abstract

Being able to effectively design biological molecules like DNA and proteins to desired specifications would have a transformative effect on science. Currently, the most popular design method in biomolecular engineering is *directed evolution* [1, Nobel Prize 2018], which explores sequence space by making small mutations to existing sequences. Alternatively, Bayesian optimization (BO) is an attractive framework for model-based black-box optimization, and has recently been successful in molecular design [26, 19, 10, 9, 7, 14, 17]. However, most large-scale BO efforts within the ML community have focused on hyper-parameter tuning for ML; such methods often do not translate to biological sequence design, where the search space is over a discrete alphabet, wet-lab experiments are run with considerable parallelism (many sequences measured simultaneously), experiments are sufficiently time consuming and expensive that only few rounds of experiments are feasible, and we must account for the safety of patients that will be treated with the sequence. This paper discusses the particularities of batched BO within this unique context, and investigates the design choices required for robust and scalable design.

1 Introduction: Protein and DNA Design as an ML Problem

This section provides a brief background on Bayesian optimization, with a focus on details that are specific to biomolecular engineering. Let f be a black-box function over a high-dimensional discrete space, which will be evaluated using batches of $B > 1$ sequences. Write $D_t = \{x, y = f(x)\}$ for the data collected after i rounds of experiments. In intermediate rounds, we select B sequences to evaluate in the lab by optimizing an *acquisition function* $a(x)$, based on a posterior distribution $P(r|D_t)$ over regressor functions $y \approx r(x)$. The acquisition function approximates the long-term utility of measuring a given set of sequences, and is (approximately) optimized using an *inner loop optimizer*.

There exist many surveys of Bayesian optimization methods, *e.g.*, [27]. Gaussian processes (GPs) are a common choice for the regressor [26, 32], but scale poorly, are sensitive to hyperparameter choice and spatial non-stationarity. Alternatively, network ensembles have achieved promising results for model guided biomolecular engineering [20]. Deep auto-encoders have also been used to learn feature-spaces over which to perform BO [7, 10]. This paper does not consider such an approach because it relies crucially on an available pretraining data for representation learning. Batching for BO broadly falls into two categories: building batches iteratively [6, 8] or using acquisition functions over batches [29, 31, 4, 28].

After the final round of experiments, our estimate of the optimum of f is $\text{argmax}_x \mathbb{E}_{r\sim P(r|D_T)}[r(x)]$ (based on the posterior mean of the regressor) or $argmax_x f(x)|x \in D_1 ... D_T$ (based on sequences measured in the lab). However, in many biological sequence design tasks, we need to protect the health of patients that will be treated with these sequences. Therefore, the proposed sequences will require validation using an expensive procedure (*e.g.*, phage display or ELISA to measure relative

Preprint. Under review.

antibody affinities to a target antigen), but may later be rejected based on characteristics independent of f (*e.g.*, toxicity or non-specificity). To maximize the chance of a proposed sequence passing this additional screening, and to help secure the safety of patients, we request that BO discover a *diverse* set of high-quality sequences. Therefore, we consider evaluation metrics below that assess our ability to recover multiple local optima of f. Here, proposing diverse sequences is an end goal of the experimental design, rather than an exploration strategy.

2 Design Choices for Bayesian Optimization for Biological Sequences

Batched BO over discrete sequences where the black-box function is evaluated in a wet lab requires design decisions and exhibits experimental behaviours that are highly specific to this setting.

Bayesian regressor. GPs scale poorly to large datasets; we instead construct an approximate posterior in terms of an ensemble of neural networks. The posterior predictive distribution is obtained using an un-weighted average of the models' predictions. To sample from the posterior distribution over models, we select a model uniformly at random. Variation across models is due to parameter initialization, SGD randomness, and bootstrap re-sampling of training data.

Batch selection. As wet-lab experiments are time-consuming but easily parallelized, each BO iteration should propose a *batch* of sequences. We employ a simple, flexible, and scalable batching strategy that first generates nB candidate sequences and then filters these down to B sequences. To select the nB candidates, we use a single-input acquisition function (*e.g.*, expected improvement [23]), and select the best sequences discovered by an inner loop solver. A simple modification to the candidate generation phase uses Thompson sampling [12, 15]: B regressors are sampled from the posterior, and for each an inner-loop optimization is performed. As this is intractable for large B, we instead employ a simple modification using $k \ll B$ regressors, where we select a set of N/k candidates for each regressor. Being able to support Thompson sampling also provides additional motivation to use DNN ensemble over GPs: while DNN ensembles allow one to directly sample a parametric function $r(x)$ from the approximate posterior, for GPs the sampled function is defined implicitly in terms of a procedure with cost that grows with the number of times the function has been evaluated.

Candidate filtering. As discussed at the end of Section 1, for biology sequence design tasks, we require a *diverse* set of final sequences to ensure that some of the selected sequences will pass an additional validation phase independent of f. We choose to enforce this by requiring diversity at the batch level; this approach is a known strategy in BO to minimize posterior uncertainty [5, 16], and also encourages exploration around multiple local optima. To this end, we investigate two methods for filtering the nB candidates down to B sequences.

- Type I Matérn hardcore point process (HCPs) [22], which remove points that are closer than a tunable distance d from previously selected points;
- Determinantal point processes (DPPs) [18], which assign probability $\Pr(S) \propto \det(L_S)$ to any given set S where the matrix $L \in \mathbb{R}^{nB \times nB}$ encodes predicted sequence quality and pairwise similarity between sequences. Previous work [16, 30] has shown that DPPs are valuable models for exploration in batched BO, and can be used to model GB-UCB-PE [5].

Inner loop solver. The wet lab experiment is expensive to evaluate, whereas the acquisition function $a(x)$ — which requires no wet lab experiments — is cheap, allowing us to spend computational time on optimizing $a(x)$. When designing longer sequences, we use *regularized evolution* [25], a general-purpose local search method. For medium-scale problems, the exact optimum can be found by brute-force enumeration of the entire search space. Note that this approach is highly characteristic of the discrete space we operate on, and would be impossible in a continuous setting.

3 Experiments

Our experiments consider two black-box optimization problems described in App. A that simulate biological sequence design tasks. To compare our system to computationally expensive baselines, we consider problems with shorter sequences and smaller batch sizes than real-world problems. In this small-scale regime, we expect the computationally-expensive methods to excel. If scalable methods are competitive, this gives us faith that our scalable methods perform well on large-scale problems, where these baselines cannot be applied.

Unless otherwise noted, we use the expected improvement acquisition function. The quality of sequences sampled in the first batch (where nothing is yet known about the space) provides considerable variance in a given method's performance. Finally, our evaluation considers the quality of the sequences proposed by solvers over the course of optimization, rather than looking only at the performance of a final 'exploit' step run in the final round. This simulates the setting where experimentalists have a budget for N rounds, but seek to find high-quality sequences as quickly as possible.

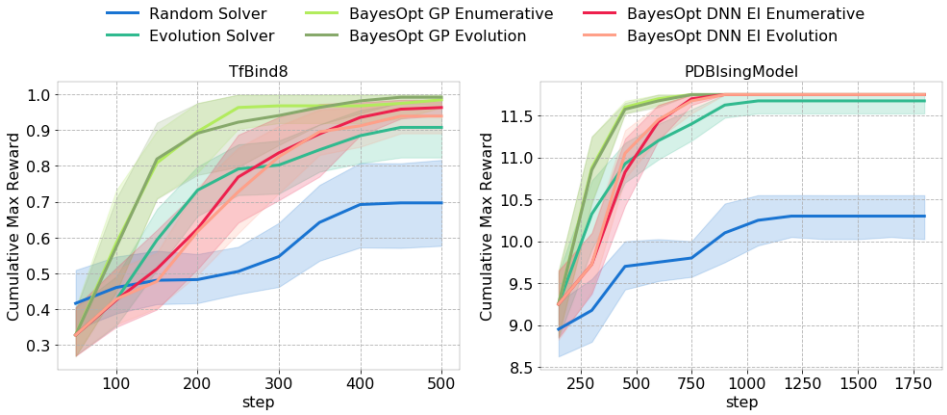
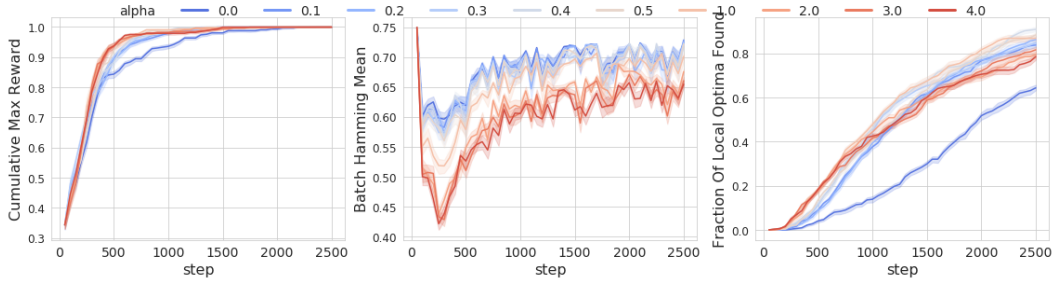
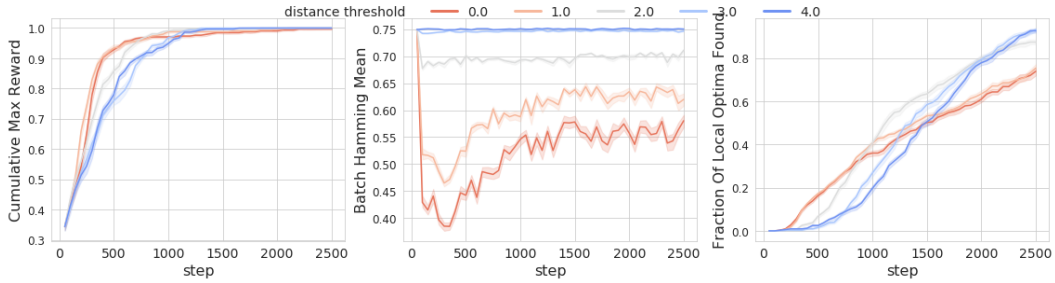


Figure 1: **Scalability-accuracy tradeoff:** two noteworthy design choices provide graceful scaling with batch size and the sequence length: switching from a GP to a DNN ensemble and from an enumerative acquisition solver to evolutionary search. The evolution solver performs similarly to enumeration (for the Ising model, their mean performances are indistinguishable). Achieving this performance crucially depends on warm-starting evolution with promising sequences from earlier rounds. For the choice of regressors, a GP with RBF kernel on a one-hot representation of the sequence performs favorably to the DNN ensemble on medium-scale problems when setting the GP hyper-parameters to their hindsight optimum. Overall, however, the BO with the GP is sensitive to choice of hyper-parameters, and in real applications inference over these would be performed on the fly. We also compare to two additional baselines: random selection and directly using the evolution solver to optimize f.



(a) DPP diversification on TfBind8



(b) HCP diversification on TfBind8

Figure 2: **Diversifying batches**: we investigate two mechanisms for filtering candidates, which are obtained using Thompson sampling. The first does greedy MAP inference over an HCP: given nB candidates $x_1, \ldots, x_{nB}$ sorted by decreasing predicted quality and a given distance d, we add point i to the batch if (a) we have selected less than B points and (b) if x_i is not within Hamming distance d of points that have already been selected. This approach has the advantage of being highly efficient, but aggressively rejects points (*e.g.*, if two high-scoring sequences are within distance $< d$ of each other, only one will be chosen). For this method, we recover standard batch BO without batch diversification for $d = 0$. The second approach builds a DPP over the nB sequences then greedily select the batch with highest probability under the DPP. This allows a graceful trade-off between the quality of each sequence and the diversity of the batch but scales poorly with nB. We investigate the family of DPPs with kernels of the form $L = q_i^\alpha k(x_i, x_j) q_j^\alpha$ for $\alpha \in [0.1, 4]$, following the quality/diversity decomposition advocated in [18]. Here, q_i is acquisition function value rescaled to $[0, 1]$, and $k(x_i, x_j)$ is the Hamming distance kernel function between sequences i, j. Both diversification methods improve the fraction of discovered optima (see App. A for a description of this metric). Unsurprisingly, we see for HCPs that a small distance threshold d is better for fast optimum discovery (left-most graph); conversely, larger values of d are better to find *many* optima. On the other hand, as DPPs gracefully trade-off point quality with set diversity, DPPs are able to simultaneously optimize for fast optimum discovery and fraction of discovered optima (*e.g.*, $\alpha = 2.0$ in (a) achieves good results across both metrics.

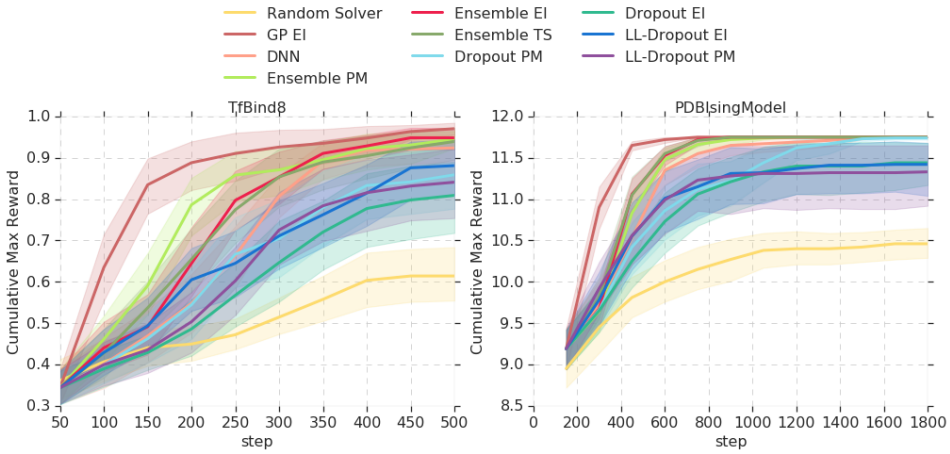


Figure 3: **Acquisition functions and uncertainty estimates:** we investigate the performance of simple methods for uncertainty estimation in neural networks: ensembling and test time dropout, including last-layer dropout (LL-Dropout) [24], when applied with the expected improvement (EI) acquisition function. We also consider alternative acquisition functions that do not require explicit modeling of the posterior predictive distribution: Thompson sampling (TS) using an ensemble and performing 'pure exploitation,' where the approximate posterior mean (PM) is the acquisition function. These are compared to using a single neural network (DNN) without uncertainty estimates, expected improvement with a GP regressor (which provides closed-form uncertainty estimates), and a negative control method which selects random sequences for each batch. Among scalable methods, we find ensembles perform better than a single neural network or dropout based methods. Additionally, we find the use of approximate posterior mean as an acquisition function is at least as effective, if not more effective, as expected improvement or Thompson sampling for these tasks.

4 Conclusion

Many intuitions and design decisions that are made for Bayesian optimization as it is commonly used in machine learning (*e.g.*, for hyper-parameter tuning) do not apply in the case of Bayesian optimization over discrete biological sequences. In this paper, we presented some of the particularities of the interactions between BO and wet-lab experiments: large-scale parallelism, the discrete search space, and the need to find not one but *a set* of high-quality points. Experiments describe how each of these design decisions impact BO performance, and aim to provide insight for practitioners performing design in bio-medical applications.

A Appendix: in-silico biological sequence design problems

The next sections provide details for the optimization problems and metrics considered above.

Transcription Factor Binding (TfBind8): Genetic variation can impact the DNA binding specificity of transcription factor proteins and result in altered gene expression levels. Such variants have been associated with various human diseases and have been implicated in Mendelian diseases [2]. In [2], protein-binding microarrays were used to evaluate DNA binding activity of all possible 8-mer DNA sequences on 201 protein targets. Later, this dataset was adapted into an objective function to optimize, where the fitness function is negative binding affinity [11]. The function is desirable for benchmarking because it can be evaluated on all possible proposed sequences in software, but is based on real experimental data. We use a batch size of 50 for this dataset.

Protein Contact Ising Model: Ising models compute the energy of a lattice of sites that can take on one of a number of configurations and were developed as a theory for describing magnetic spin [13]. We have constructed an artificial version using binary protein contact maps from the Protein Data Bank (PDB) [3, 21]. This contact map defines an Ising model where each node corresponds to an amino acid in the protein, and an edge exists between two amino acids if the distance in 3-D space is less than a predefined threshold when the protein is folded. The goal is to maximize the energy of this model, using experiments with a batch size of 150. The optimization landscape is non-trivial because the pairwise potentials for each edge encourage nodes to take on opposing values. The model we employ consists of binary sequences of length 20, resulting in a space of over a million sequences.

Diversity-Based Metrics: Local optima for TfBind8 were computed, and the fraction of local optima found was utilized as a metric. Since the TfBind8 problem applies the same reward (normalized to [0, 1]) to sequences that are reverse complements of each other, the forward sequences in each pair of reverse complements were gathered by ordering the sequence space lexicographically and including each sequence in the set of forward sequences unless the set already contained its reverse complement. For each of 12 binding sites, these forward sequences were thresholded at a reward value to obtain a smaller set of good quality sequences. Agglomerative clustering was performed on this set of sequences, using a distance matrix consisting of pairwise hamming distances, and the maximum reward sequences in each cluster were determined to be local optima for the space. The reverse complement of each optima were also considered as optima.

We also report the average Hamming distance between pairs of sequences within a batch. Although recently there has been investigation of using the set likelihood under a DPP as a metric for diversity, we do not use this metric as it would unfairly advantage DPP-based batching.

References

[1] Frances H Arnold. Design by directed evolution. *Accounts of chemical research*, 31(3):125–131, 1998.

[2] Luis A. Barrera, Anastasia Vedenko, Jesse V. Kurland, Julia M. Rogers, Stephen S. Gisselbrecht, Elizabeth J. Rossin, Jaie Woodard, Luca Mariani, Kian Hong Kock, Sachi Inukai, Trevor Siggers, Leila Shokri, Raluca Gordân, Nidhi Sahni, Chris Cotsapas, Tong Hao, Song Yi, Manolis Kellis, Mark J. Daly, Marc Vidal, David E. Hill, and Martha L. Bulyk. Survey of variation in human transcription factors reveals prevalent DNA binding changes. *Science*, 351(6280):1450–1454, 2016.

[3] Helen M. Berman, John Westbrook, Zukang Feng, Gary Gilliland, T. N. Bhat, Helge Weissig, Ilya N. Shindyalov, and Philip E. Bourne. The Protein Data Bank. *Nucleic Acids Research*, 28(1):235–242, 01 2000.

[4] Clément Chevalier and David Ginsbourger. Fast computation of the multi-points expected improvement with applications in batch selection. In *International Conference on Learning and Intelligent Optimization*, pages 59–69. Springer, 2013.

[5] Emile Contal, David Buffoni, Alexandre Robicquet, and Nicolas Vayatis. Parallel Gaussian process optimization with upper confidence bound and pure exploration. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 225–240. Springer, 2013.

[6] Thomas Desautels, Andreas Krause, and Joel W Burdick. Parallelizing exploration-exploitation tradeoffs in Gaussian process bandit optimization. *The Journal of Machine Learning Research*, 15(1):3873–3923, 2014.

[7] Rafael Gómez-Bombarelli, Jennifer N Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D Hirzel, Ryan P Adams, and Alán Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science*, 4(2):268–276, 2018.

[8] Javier González, Zhenwen Dai, Philipp Hennig, and Neil Lawrence. Batch Bayesian optimization via local penalization. In *Artificial intelligence and statistics*, pages 648–657, 2016.

[9] Javier Gonzalez, Joseph Longworth, David C James, and Neil D Lawrence. Bayesian optimization for synthetic gene design. *arXiv preprint arXiv:1505.01627*, 2015.

[10] Ryan-Rhys Griffiths and José Miguel Hernández-Lobato. Constrained Bayesian optimization for automatic chemical design. *arXiv preprint arXiv:1709.05501*, 2017.

[11] Tatsunori B Hashimoto, Steve Yadlowsky, and John C Duchi. Derivative free optimization via repeated classification. *arXiv preprint arXiv:1804.03761*, 2018.

[12] José Miguel Hernández-Lobato, James Requeima, Edward O Pyzer-Knapp, and Alán Aspuru-Guzik. Parallel and distributed Thompson sampling for large-scale accelerated exploration of chemical space. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1470–1479. JMLR. org, 2017.

[13] Ernst Ising. Beitrag zur Theorie des Ferromagnetismus. *Zeitschrift für Physik A Hadrons and Nuclei*, 31(1):253–258, 1925.

[14] Wengong Jin, Kevin Yang, Regina Barzilay, and Tommi Jaakkola. Learning multimodal graph-to-graph translation for molecular optimization. *arXiv preprint arXiv:1812.01070*, 2018.

[15] Kirthevasan Kandasamy, Akshay Krishnamurthy, Jeff Schneider, and Barnabás Póczos. Parallelised Bayesian optimisation via Thompson sampling. In *International Conference on Artificial Intelligence and Statistics*, pages 133–142, 2018.

[16] Tarun Kathuria, Amit Deshpande, and Pushmeet Kohli. Batched Gaussian process bandit optimization via determinantal point processes. In *Advances in Neural Information Processing Systems*, pages 4206–4214, 2016.

[17] Ksenia Korovina, Sailun Xu, Kirthevasan Kandasamy, Willie Neiswanger, Barnabas Poczos, Jeff Schneider, and Eric P Xing. ChemBO: Bayesian Optimization of Small Organic Molecules with Synthesizable Recommendations. *arXiv preprint arXiv:1908.01425*, 2019.

[18] Alex Kulesza, Ben Taskar, et al. Determinantal point processes for machine learning. *Foundations and Trends® in Machine Learning*, 5(2–3):123–286, 2012.

[19] Matt J Kusner, Brooks Paige, and José Miguel Hernández-Lobato. Grammar variational autoencoder. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1945–1954. JMLR. org, 2017.

[20] Ge Liu, Haoyang Zeng, Jonas Mueller, Brandon Carter, Ziheng Wang, Jonas Schilz, Geraldine Horny, Michael E Birnbaum, Stefan Ewert, and David K Gifford. Antibody complementarity determining region design using high-capacity machine learning. *bioRxiv*, page 682880, 2019.

[21] Laura M. Luh, Robert Hänsel, Frank Löhr, Donata K. Kirchner, Katharina Krauskopf, Susanne Pitzius, Birgit Schäfer, Peter Tufar, Ivan Corbeski, Peter Güntert, and Volker Dötsch. Molecular crowding drives active pin1 into nonspecific complexes with endogenous proteins prior to substrate recognition. *Journal of the American Chemical Society*, 135(37):13796–13803, 2013. PMID: 23968199.

[22] Bertil Matérn. Spatial variation: Stochastic models and their application to some problems in forest surveys and other sampling investigations. *Meddelanden från statens Skogsforskningsinstitut*, 49:144, 1960.

[23] J. Mockus, Vytautas Tiesis, and Antanas Zilinskas. The application of Bayesian methods for seeking the extremum. *Towards Global Optimization*, 2:117–129, 09 2014.

[24] Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, D Sculley, Sebastian Nowozin, Joshua V. Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can You Trust Your Model's Uncertainty? Evaluating Predictive Uncertainty Under Dataset Shift. *arXiv preprint arXiv:1903.06694*, 2019.

[25] Esteban Real, Alok Aggarwal, Yanping Huang, and Quoc V Le. Regularized evolution for image classifier architecture search. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4780–4789, 2019.

[26] Philip A Romero, Andreas Krause, and Frances H Arnold. Navigating the protein fitness landscape with gaussian processes. *Proceedings of the National Academy of Sciences*, 110(3):E193–E201, 2013.

[27] Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando De Freitas. Taking the human out of the loop: A review of Bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2015.

[28] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical Bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, pages 2951–2959, 2012.

[29] Jialei Wang, Scott C Clark, Eric Liu, and Peter I Frazier. Parallel Bayesian global optimization of expensive functions. *arXiv preprint arXiv:1602.05149*, 2016.

[30] Zi Wang, Chengtao Li, Stefanie Jegelka, and Pushmeet Kohli. Batched high-dimensional Bayesian optimization via structural kernel learning. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 3656–3664, International Convention Centre, Sydney, Australia, 06–11 Aug 2017. PMLR.

[31] James Wilson, Frank Hutter, and Marc Deisenroth. Maximizing acquisition functions for Bayesian optimization. In *Advances in Neural Information Processing Systems*, pages 9884–9895, 2018.

[32] Kevin K Yang, Zachary Wu, and Frances H Arnold. Machine-learning-guided directed evolution for protein engineering. *Nature methods*, page 1, 2019.