Benchmarking the Performance of Bayesian Optimization across Multiple Experimental Materials Science Domains

Qiaohao Liang[1], **Aldair E. Gongora**[2], **Zekun Ren**[3], **Armi Tiihonen**[4], **Zhe Liu**[4], **Shijing Sun**[4], **James R. Deneault**[5], **Daniil Bash**[6], **Flore C.L. Mekki-Berrada**[7], **Saif A. Khan**[7], **Kedar Hippalgaonkar**[6], **Benji Maruyama**[5], **Keith A. Brown**[2], **John Fisher III**[8], and **Tonio Buonassisi**[4]

[1]Department of Materials Science, Massachusetts Institute of Technology
[2]Department of Mechanical Engineering, Boston University
[3]Singapore-MIT Alliance for Research and Technology
[4]Department of Mechanical Engineering, Massachusetts Institute of Technology
[5]Air Force Research Laboratory
[6]Agency for Science, Technology and Research (A*STAR)
[7]Department of Chemical and Biomolecular Engineering, National University of Singapore
[8]Computer Science & Artificial Intelligence Lab, Massachusetts Institute of Technology

Abstract

In the field of machine learning (ML) for materials optimization, active learning algorithms, such as Bayesian Optimization (BO), have been extensively used to guide high-throughput autonomous experimentation systems. However, very few studies have evaluated the efficiency of BO as a general optimization algorithm for a broad range of material systems. In this work, we benchmark the performance of BO with a collection of surrogate model and acquisition function pairs across five diverse experimental materials science domains. By defining acceleration and enhancement metrics for specific research objectives, we found that for surrogate model selection, Gaussian process (GP) equipped with automatic relevance detection (ARD) and random forests (RF) have comparable performance and both outclass GP without ARD. From our two-way analysis of utilizing RF and GP surrogate models with acquisition functions such as expected improvement (EI) and lower confidence bound (LCB) in the BO framework, we provide practical insights for scientists interested in utilizing BO for materials design and optimization.

1 Introduction

Closed-loop, high-throughput autonomous experimentation systems have recently emerged as the new frontier of accelerated materials research and have sparked efforts in adapting and integrating advanced lab automation components and state-of-the-art machine learning (ML) algorithms. The data efficient Bayesian optimization (BO) has gained great popularity in several materials optimization applications[1-3] because it allows one to take advantage of the full information provided by the history of the optimization sequence [4].

While the field enjoyed success in optimizing materials through high-throughput experiments driven by BO and its derivatives [5-7], the priority has been to demonstrate how one can reach an improved objective within a smaller experimental budget, and thus often the precise acceleration and improvement factors resulting from applying these ML algorithms have not consistently quantified. To our knowledge, one study [8] has benchmarked BO's performance within a specific electrocatalyst composition space using multiple ML models and acquisition functions, but applicability to a broader array of materials research remains unanswered. The paucity of comprehensive benchmarks could slow down the development of machine learning algorithms specifically designed for experimental materials optimization, especially as the field seeks to tackle the optimization in new materials design spaces with higher space complexity and dimensions.

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

In this work, we benchmark the performance of BO across five experimental materials science domains. We demonstrate that RF [9] as a surrogate model can compete against GP [10] equipped with automatic relevance detection (ARD) [11], and both significantly outperform GP without ARD. Our detailed two-way comparison of the RF and GP's implicit assumptions, robustness to noise, and hyperparameter tuning will yield deeper insights on surrogate model selection for materials optimization campaigns. We also observe that the lower confidence bound (LCB) acquisition function exhibits consistent performance advantages over other myopic acquisition functions such as expected improvement (EI). We share our insights on using BO for general materials optimization research and provide open source implementation of benchmarking code to support future ML algorithm development in the field.

2 Methods

2.1 Benchmarking datasets

As seen in Table 1, we assembled a list of five high-throughput experimental datasets with varying dimensions, features, and materials domains. Such diversity is intended to allow us to observe different BO algorithm's performance across a broader range of materials systems. The datasets' differences in objective materials response distribution (Fig 1a) and how uniformly sampled data points are distributed in their respective design spaces (Fig 1b) are also visualized, which demonstrates the diverse and complementary nature of global optimization problems in the study. Also all materials optimization tasks have been turned into global minimization problems for consistency of the benchmarking framework applied below.

Table 1: Description of experimental materials science datasets

Dataset	Domain	Synthesis	Size	n_{dim}	Optimization Objective
P3HT/CNT [12]	Composite blends	Drop casting	178	5	Conductivity
AgNP [13]	Silver Nanoparticles	Flow synthesis	164	5	Spectrum
Perovskite [6]	Perovskite Stability	Spin coating	94	3	Stability
Crossed barrel [7]	3D Printed Structure	3D Printing	600	4	Toughness
AutoAM [*]	Materials Manufacturing	3D Printing	100	4	Shape score

Note the dataset with [*] will be published by original authors in journal or on arxiv soon.

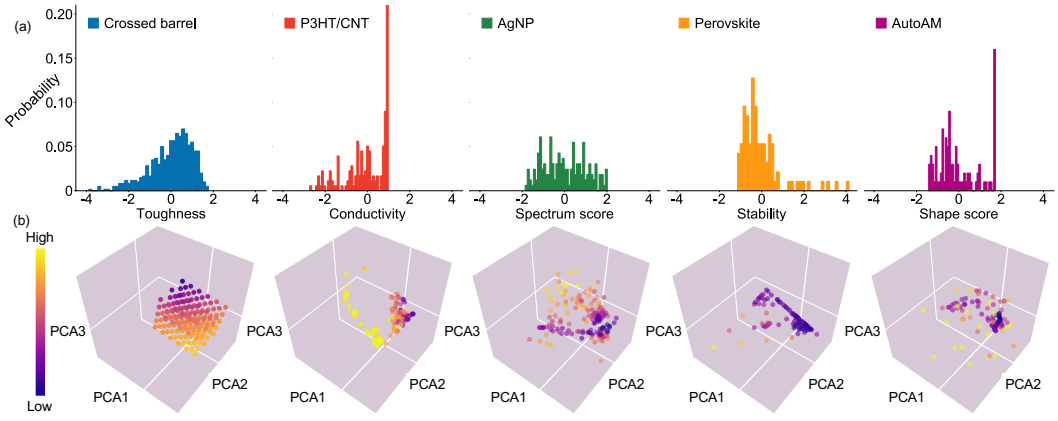

Figure 1: Dataset manifold complexity analysis. a) Histogram of normalized objective values. Each dataset's objective values are independently centered to their mean and scaled to unit variance. b) Design space after dimension reduction to 3D via principal component analysis.

2.2 Bayesian Optimization

In this study, for each dataset, to implement a Bayesian Optimization [10] algorithm, we first select surrogate model between scikit-learn random forest (RF) [9], GPy's [14] Gaussian process (GP) regression, and GP equipped with automatic relevance detection (ARD) [11]. Assuming little prior information is known of respective materials domains in our simulated optimization campaign, RF has pre-selected hyperparameters $n_{\text{tree}} = 50$, bootstrap = True and GP has kernel choices of Matérn52,

Matérn32, Matérn12, radial basis function (RBF), and multilayer perceptron (MLP). For training of GP, ARD allows a GP's kernel to have individual characteristic lengthscales l_i [10-11] for each of the input feature dimensions i. l_i can be used to estimate distance moved along i^{th} dimension in input space before the objective values become uncorrelated, and $\frac{1}{l_i}$ represents relevancy of feature i to predicting the objective. We pair the selected surrogate model with one of the myopic acquisition functions, including expected improvement (EI), maximum probability of improvement (MPI), and LCB with varying mean and standard deviation weights $\text{LCB}_{w_1 w_2}(\vec{x}) = -w_1\mu(\vec{x}) + w_2\sigma(\vec{x})$, whose notation is simplified to $\text{LCB}_{\overline{\lambda}}$, $\overline{\lambda} = \frac{w_1}{w_2}$.

The above surrogate models, their hyperparameters, and acquisition functions are chosen because they represent the most off-the-shelf options readily available and widely applied for materials optimization campaigns in the field. We welcome researchers to test their algorithms utilizing Bayesian Neural Network surrogate model or customized acquisition functions with embedded physics [6] on the five datasets above for general performance.

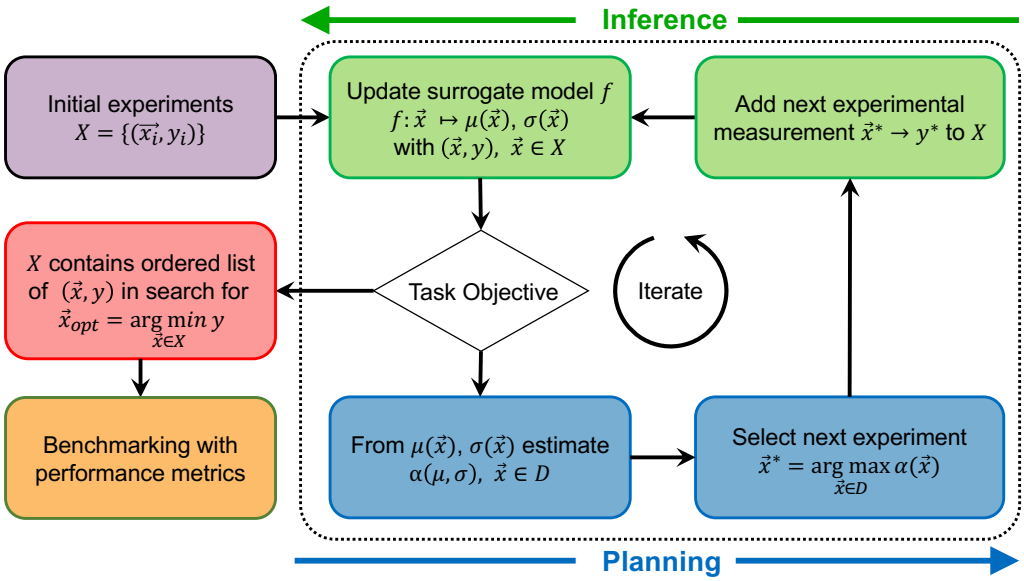

Figure 2: Benchmarking framework including a simulation of BO performing closed-loop optimization with alternating inference and planning stages.

2.3 Benchmarking framework

Within each respective experimental dataset, the set of single data points form representations of ground truth in the materials design-performance space. Figure 2 shows the exact benchmarking framework we adapt [8] to simulate optimization campaigns guided by BO in each materials system. To imitate early stage exploration, we randomly draw $n = 2$ initial experiments with no replacement from original set $D = \{(\vec{x_i}, y_i) | i = 1, 2, \ldots, N\}$ and add to collection $X = \{(\vec{x_i}, y_i) | i = 1, 2, \ldots, n\}$. During planning stage, surrogate model f is used to estimate mean $\mu(\vec{x})$ and standard deviation $\sigma(\vec{x})$. We can calculate acquisition function values $\alpha(\mu(\vec{x}), \sigma(\vec{x}))$ for each remaining experimental action $\vec{x} \in D$ in parallel. At each cycle, action $\vec{x}^* = \arg\max_x \alpha(\vec{x})$ will be selected as next experiment. During inference stage, after selecting action $\vec{x}^*$, the corresponding sample observation y^* is obtained, and $(\vec{x}^*, y^*)$ is added to X and removed from set D. Surrogate model f is retrained using values from updated X, and the sequential alternation between planning and inference is repeated until a defined task objective is met. Each BO algorithm is run for 50 ensembles with different initiation and with its aggregated performance (50 averaged runs resulting from 10 random 5-fold splits using the 50 raw runs) compared against random search, we can quantitatively evaluate its performance via active learning metrics [8] further defined below.

3 Results

While the five datasets covered a breadth of materials domains, the tested BO algorithms' relative performances were observed to be consistent. RF, GP ARD (Matérn52), and GP (Matérn52) in Figure 2 are meant to show such trend. For full results including all combination of GP kernels and acquisition functions, please refer to supporting information of our work's preprint.

Across all the investigated datasets,We showcase the benchmarking results using the crossed barrel dataset [7], which was collected by grid sampling of the design space through an robotic experimental system while optimizing the mechanics of additive manufactured components. With top 5% toughness as objective threshold for target candidates in materials discovery, we use the metric $All \in [0, 1]$ to show the fraction of the top 5% crossed barrel structures that have been evaluated by cycle $i = 2, 3, \ldots, N$. All describes how quickly can we identify multiple top candidates in a materials design space. Keeping multiple well-performing candidates allows one to not only observe regions in design space that frequently yield high-performing samples but also have backup options for further evaluation should the most optimal candidate fail in subsequent materials applications evaluations. There are research objectives related to finding any good materials candidate, yet in those cases, random selection could outperform optimization algorithms due to luck in a simple design space. Our objective of finding multiple or all of top tier candidates is are more applicable to real materials optimization scenario and suitable for demonstrating the true efficacy and impact of BO.

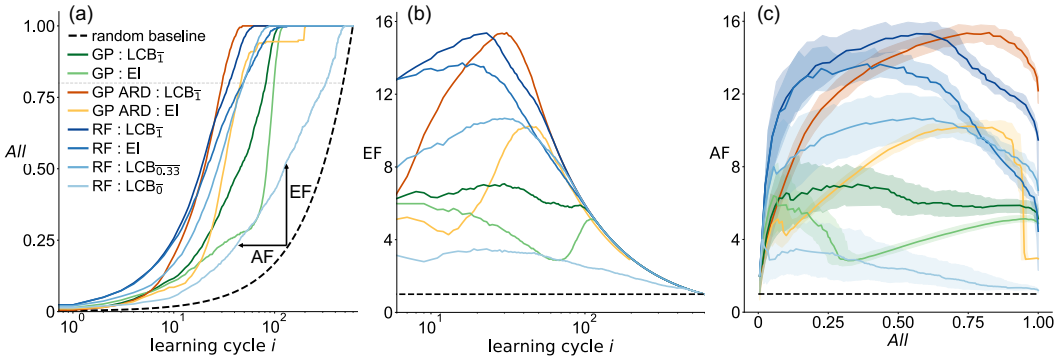

Figure 3: BO algorithms' aggregated performance on the crossed barrel dataset measured by a) All vs. learning cycle i against random baseline, and how b) EF and c) AF are derived from it. The algorithms include surogate models GP (green), GP ARD (orange) and RF (blue) paired with acquisition functions EI and LCB$_{\bar{\lambda}}$. Higher saturation is correlated with better performance. Median value (line) and 25^{th} to 75^{th} percentile (shades) of the 50-run ensemble are shown at each cycle i.

Figure 3a illustrates learning rates based on All metric for RF and GP with multiple acquisition functions. We have the following observations: (1) Under the same acquisition function, RF's performance as a surrogate model is on par, if not slightly worse, when compared to the performance of GP with ARD. (2) Similarly, both RF and GP with ARD outclass GP without ARD. (3) We investigate a spectrum of mean and standard deviation weights for LCB to adjust the ratio between exploration and exploitation. We see that LCB$_{\bar{1}}$ (which has the most balanced ratio between exploration and exploitation) outperforms other acquisition functions, namely EI, known to make excessive greedy decisions[15], and other LCBs that overly emphasize exploration.

To further quantify the relative performance, we set $All = 0.8$ as a realistic goal to indicate we have identified 80% of the structures with top 5% toughness (Figure 2a). For surrogate models paired LCB$_{\bar{1}}$, we see that GP with ARD and RF reach that goal by evaluating approximately 30 candidates out of the total of 600, whereas GP without ARD needs about 90 samples. We also adapt two other metrics to quantify the acceleration of optimization due to BO. Both compared to a statistical random baseline, enhancement factor $\text{EF}(i) = \dfrac{All_{\text{BO}}(i)}{All_{\text{random}}(i)}$ shows how much improvement in a metric we would receive at cycle i, and acceleration factor $\text{AF}(All = a) = \dfrac{i_{\text{BO}}}{i_{\text{random}}}$, the ratio of cycle numbers showing how much faster we could reach a specific $All(i_{\text{BO}}) = All(i_{\text{random}}) = a$ value. The aggregated performance of BO algorithms is further quantified via EF and AF curves in Figure 4b, 4c: starting off with low EFs or AFs before the surrogate model gains more accuracy;

peaking at high EFs and AFs of up to $16\times$; eventually, the learning algorithms show diminishing returns from an information gain perspective as we progress deeper into our optimization campaigns. Noticeably, EFs and AFs for other four datasets were in the $2\times$ to $5\times$ range. The difference can be attributed to the data collection methodology of each dataset: while the crossed barrel dataset was collected using a grid sampling approach, the other four datasets were collected under the guidance of BO. Their datasets not only avoid oversampling in non-optimal spaces but also already carry an intrinsic enhancement and acceleration bias, resulting in lower EFs and AFs.

4 Discussion

From observations above, we believe in the context of experimental materials optimization, besides GP with ARD, RF also merits consideration as surrogate model in materials optimization campaigns.

There are many reasons to consider RF as surrogate model aside GP in future materials optimization campaigns: (1) With n as the number of training data, n_{dim} as input feature dimension, n_{tree} as number of decisions trees kept in RF model, in terms of time complexity, we have $t_{RF} = \mathcal{O}(nlog(n) \cdot n_{\text{dim}} \cdot n_{\text{tree}}) < t_{GP} = \mathcal{O}(n^3)$. For reasonable choice of n, n_{dim}, and n_{tree} in physical science research, RF trains faster via parallel computing of its decision trees, scales better with increasing n than GP, and thus allow researchers to speed up the feedback loop between ML and automated experiments. (2) RF's performance across these noisy experimental datasets can be attributed to it being an ensemble frequentist learning method, thus likely more robust to noise and applicable for generalized prediction. The aggregation process in RF of different decision trees mitigates overfitting while keeps decision tree's insensitivity to outliers from recursive partitioning and local model fitting, resulting in model with relatively low bias and medium variance. RFs have advantage in modeling step-wise or discrete objective functions, yet perform poorly in extrapolation beyond the search space covered by its training data. Yet in the context of materials optimization campaigns, this disadvantage can be mitigated by adapting sampling strategies like latin hypercube sampling (LHS) to pseudo-randomly cover a wider range of data in each dimension so that RF would not have to often extrapolate to completely unknown regions. (3) While RF has potentially more hyperparameters to tune, in the context of materials optimization, sub-optimal choice of RF's hyperparameters has less penalty compared to choosing a incompatible GP kernel with the domain manifold, which could significantly slow down optimization conversion. Instead of devoting nontrivial of experimental budget to optimize GP's kernels using adaptive kernels[16], automating kernel selection[17], RF is an easier off-the-shelf option that allows one to make fewer structural assumptions about unfamiliar domain manifolds.

GP with ARD also has its unique advantage: it allows us to assign individual lengthscales for each input dimension i in a GP's kernel function. With high l_i values implying low relevancy, these lengthscales provide a "weight" for understanding the objective's sensitivity to each input. GP without ARD will only have single lengthscale l as scaling parameter controlling GP's kernel, which is at odds with the fact that each input feature has its distinct contribution to the materials objective. In machine learning, l_i have been used for removing irrelevant inputs [10-11]. In the context of materials optimization, ARD could identify a few directions in the input space with specially high "relevance," and their lengthscales give the inverse characteristic length-scale for those directions [10]. This means that if we train GP with ARD on input data in their original units without normalization, once we extract the lengthscale of each feature l_i, our model in theory won't be able to extrapolate more than l_i units away from our data in dimension i, and thus limits the range of next experiments to our benefit. For a particular feature with a relative small l_i, it means that for small change in this input dimension i's value, we would have quite significant large change in objective value; thus, the sampling density in this dimension should be high enough to capture such sensitivity. For the above mentioned reasons, it would be good practice for materials researchers in the field to emphasize their use of ARD in GP.

5 Conclusion

We benchmarked the performance of BO algorithms across five different experimental materials science domains. We used active learning metrics to quantitatively evaluate the enhancement and acceleration of BO for common research objectives in materials optimization campaigns. We demonstrate that RF as surrogate model can compete with GP with ARD, and both significantly outperform GP without ARD. The observations are followed by detailed two way analysis of considerations in using RF and GP with ARD as surrogate model in the context of materials optimization campaigns. Through our benchmarking effort, we hope to share our insights with the field and encourage a closer collaboration between machine learning and broader physical science communities.

6 Broader Impact

Successful benchmarks of active learning algorithms for closed-loop materials optimization is only a starting point. Our observations demonstrate how the choice of active learning algorithms has to adapt to their applications in materials science, motivating more efficient ML-guided experimentation, and will likely directly result in a larger number of successful optimization of materials with record breaking properties. The impact of this work could extend beyond the materials science, and will motivate algorithm development to substantially accelerate research and realize the paradigm shifts envisioned by early adopters of ML for physical science.

Acknowledgements

Q.L. acknowledges generous funding from TOTAL S.A. for supporting his graduate research. A.G., K.H. thank Google LLC, the Boston University Dean's Catalyst Award, The Boston University Rafik B. Hariri Institute for Computing and Computational Science and Engineering, and NSF (CMMI-1661412) for support in this work and studies generating crossed barrel dataset. Z.K., A.T., Z.L., S.S., T.B. acknowledge support from DARPA under Contract No. HR001118C0036, TOTAL S.A., US National Science Foundation grant CBET-1605547, and the Skoltech NGP program for research generating Perovskite dataset. J.D., B.M. thank AFOSR Grant 19RHCOR089 for supporting their work in generating AutoAM dataset. D.B., K.H. acknowledge funding from the Accelerated Materials Development for Manufacturing Program at A*STAR via the AME Programmatic Fund by the Agency for Science, Technology and Research under Grant No. A1898b0043 and A*STAR Graduate Academy's SINGA programme for producing P3HT/CNT dataset. F.M-B., S.K. acknowledge support from the Accelerated Materials Development for Manufacturing Program at A*STAR via the AME Programmatic Fund by the Agency for Science, Technology and Research under Grant No. A1898b0043.

References

[1] Yamawaki, Masaki, Masato Ohnishi, Shenghong Ju, and Junichiro Shiomi. "Multifunctional structural design of graphene thermoelectrics by Bayesian optimization." Science advances 4, no. 6 (2018): eaar4192.

[2] Bassman, Lindsay, Pankaj Rajak, Rajiv K. Kalia, Aiichiro Nakano, Fei Sha, Jifeng Sun, David J. Singh et al. "Active learning for accelerated design of layered materials." npj Computational Materials 4, no. 1 (2018): 1-9. Harvard

[3] Rouet-Leduc, Bertrand, Kipton Barros, Turab Lookman, and Colin J. Humphreys. "Optimisation of GaN LEDs and the reduction of efficiency droop using active machine learning." Scientific reports 6 (2016): 24862.

[4] Shahriari, B., Swersky, K., Wang, Z., Adams, R.P. and De Freitas, N., 2015. Taking the human out of the loop: A review of Bayesian optimization. Proceedings of the IEEE, 104(1), pp.148-175.

[5] Häse, Florian, Loïc M. Roch, Christoph Kreisbeck, and Alán Aspuru-Guzik. "Phoenics: A Bayesian optimizer for chemistry." ACS central science 4, no. 9 (2018): 1134-1145.

[6] Sun, Shijing, Armi Tiihonen, Felipe Oviedo, Zhe Liu, Janak Thapa, Noor Titan Putri Hartono, Anuj Goyal et al. "A Physical Data Fusion Approach to Optimize Compositional Stability of Halide Perovskites." ChemRxiv preprint(2020).

[7] Gongora, Aldair E., Bowen Xu, Wyatt Perry, Chika Okoye, Patrick Riley, Kristofer G. Reyes, Elise F. Morgan, and Keith A. Brown. "A Bayesian experimental autonomous researcher for mechanical design." Science Advances 6, no. 15 (2020): eaaz1708.

[8] Rohr, Brian, Helge S. Stein, Dan Guevarra, Yu Wang, Joel A. Haber, Muratahan Aykol, Santosh K. Suram, and John M. Gregoire. "Benchmarking the acceleration of materials discovery by sequential learning." Chemical Science 11, no. 10 (2020): 2696-2706.

[9] Pedregosa, Fabian, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel et al. "duchesnay E." Scikit-learn: Machine Learning in Python. J. Mach. Learn. Res 12 (2010): 2825-2830.

[10] Williams, Christopher KI, and Carl Edward Rasmussen. Gaussian processes for machine learning. Vol. 2, no. 3. Cambridge, MA: MIT press, 2006.

[11] Neal, R.M., 2012. Bayesian learning for neural networks (Vol. 118). Springer Science & Business Media.

[12] Bash, Daniil; Cai, Yongqiang; Vijila, Chellappan; Wong, Swee Liang; Xu, Yang; Kumar, Pawan; et al. (2020): Machine Learning and High-Throughput Robust Design of P3HT-CNT Composite Thin Films for High Electrical Conductivity. ChemRxiv. Preprint.

[13] Mekki-Berrada, Flore, Zekun Ren, Tan Huang, Wai Kuan Wong, Fang Zheng, Jiaxun Xie, Isaac Parker Siyu Tian et al. "Two-Step Machine Learning Enables Optimized Nanoparticle Synthesis." ChemRxiv preprint(2020).

[14] GPy. "A gaussian process framework in python." (2012).

[15] Ryzhov, Ilya O. "On the convergence rates of expected improvement methods." Operations Research 64.6 (2016): 1515-1528

[16] Wilson, A. and Adams, R., 2013, February. Gaussian process kernels for pattern discovery and extrapolation. In International conference on machine learning (pp. 1067-1075).

[17] Schlessinger, Louis, Gustavo Malkomes, and Roman Garnett. "Automated model search using Bayesian optimization and genetic programming." NeurIPS Meta-learning workshop (2019).