Self-driving labs: how to tell the optimizer not to sample in an undesired region?

Flore Mekki-Berrada[a], **Andre K.Y. Low**[b,c], **Kedar Hippalgaonkar**[b,c] **and Saif A. Khan**[a]

[a] *Department of Chemical and Biomolecular Engineering, National University of Singapore, 117585 Singapore*

[b] *School of Materials Science and Engineering, Nanyang Technological University, Singapore, 639798, Singapore*

[c] *Institute of Materials Research and Engineering (IMRE), Agency for Science, Technology and Research (A*STAR), 2 Fusionopolis Way, Innovis #08-03, Singapore, 138634, Republic of Singapore*

flore@nus.edu.sg

Recent years have seen a fast development of high-throughput experimental platforms (HTEP) in material science and a growing interest in expanding the use of such platforms to solve constrained multi-objective optimization problems (cMOOPs) [1,2,3]. When objectives are conflicting with each other, the optimizer not only seeks for a single optimum but for a set of solutions – called Pareto Front (PF) - that would give the best trade-off while satisfying experimental constraints. However, handling constraints in such multi-objective optimizers can be a challenge, as experimental constraints are not always quantifiable.

Two types of constraints are generally encountered in automated experimental systems. The most common ones are *a priori* quantifiable constraints - e.g. the box constraints limiting the boundaries of the input parameter space – for which sampling feasibility is known without running the experiment. These constraints are generally quantified through mathematical equalities or inequalities and are widely implemented in optimization algorithms.

Less common, but more critical for self-driving labs, are experimental non-quantifiable constraints, also referred to as implicit constraints. They require the experiment to be run to determine sampling feasibility. These constraints typically come into play when the optimizer suggests conditions that would affect measurement accuracy – e.g. signal saturation, unexpected phase separation distorting the signal –, or cause deterioration of the experimental system – like clogging, pressure build-up. In the latter case, avoiding repeated sampling in infeasible regions is crucial, as this might lead to irreversible damage to the experimental platform. The challenge is therefore to quickly learn the boundaries of the infeasible regions in the input parameter space and minimize the number of infeasible candidates suggested by the optimizer [4].

Although constraint handling has been widely studied in the context of evolutionary algorithms [4,5], only a few implicit constraint handling methods have been implemented in self-driving labs [6]. Recently, Low et al [7] proposed a multi-objective Evolution-Guided Bayesian Optimization (EGBO) algorithm that integrates the selection pressure into a q-Noisy Expected Hypervolume Improvement (qNEHVI) optimizer. This hybrid algorithm shows good performance both in terms of fast convergence towards the Pareto Front (PF) and good coverage of the PF. Here, we propose to explore how to integrate quantifiable and non-quantifiable constraint-handling in EGBO to decrease sampling in the infeasible regions.

To handle non-quantifiable constraints, we developed a constraint metric based on the nearest neighbor distance between two populations: feasible and infeasible candidates. Such a metric can easily be implemented in multi-objective optimization algorithms like qNEHVI. We test the validity of such approach on a colorimetric experimental platform designed to find the best recipe for a universal pH indicator. We demonstrate the generality of this method on different synthetic problems (see Fig.1) and show that, in highly constrained problems where constraints are a non-linear function of the parameter space, our proposed constraint-handling method can lead to a more efficient and more precise resolution of the Pareto front than binary constraint handling or even explicit constraint handling methods.

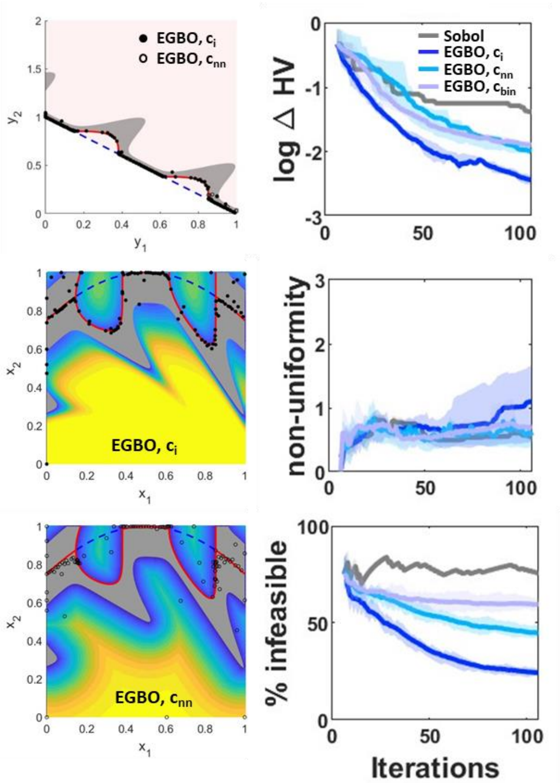

Fig. 1: a. Optimization trajectory in the objective (top) and decision (middle and bottom) spaces, obtained on MW3 problem [8] with EGBO implemented with the exact values of the constraints c_i (explicit constraint handling) or the nearest neighbor-based metric c_{nn} (implicit constraint handling). Feasible space is highlighted in grey and the PF represented by a red line. **b.** EGBO performance indicators on MW3 problem for different constraint handling: c_i (explicit handling, blue line), c_{nn} (two-population nearest neighbor implicit handling, cyan line), c_{bin} (binary implicit handling, purple line). Sobol sampling performance (grey line) is indicated as baseline.

Self-driving labs: how to tell the optimizer not to sample in an undesired region?

Flore Mekki-Berrada[a], **Andre K.Y. Low**[b,c], **Kedar Hippalgaonkar**[b,c] **and Saif A. Khan**[a]

[a] *Department of Chemical and Biomolecular Engineering, National University of Singapore, 117585 Singapore*

[b] *School of Materials Science and Engineering, Nanyang Technological University, Singapore, 639798, Singapore*

[c] *Institute of Materials Research and Engineering (IMRE), Agency for Science, Technology and Research (A*STAR), 2 Fusionopolis Way, Innovis #08-03, Singapore, 138634, Republic of Singapore*

flore@nus.edu.sg

Acknowledgments

The authors acknowledge funding from the 27th NRF CRP program (NRF-CRP27-2021-0001).

References

[1] R.W. Epps, A.A. Volk, K.G. Reyesb, M. Abolhasani. Accelerated AI development for autonomous materials synthesis in flow. In *Chem. Sci.* 12, 6025, 2021.

[2] H. Tao, T. Wu, S. Kheiri, M. Aldeghi, A. Aspuru-Guzik, E. Kumacheva. Self-Driving Platform for Metal Nanoparticle Synthesis: Combining Microfluidics and Machine Learning. In *Adv. Funct. Mater.* 2106725 2021.

[3] L. Cao et al. Optimization of Formulations Using Robotic Experiments Driven by Machine Learning DoE. In *Cell Reports Physical Science* 2, 100295, 2021.

[4] Z. Michalewicz. Do Not Kill Unfeasible Individuals. In *Proceedings of the Fourth Intelligent Information Systems Workshop,* 1995.

[5] C.A. Coello Coello. Theoretical and numerical constraint-handling techniques used with evolutionary algorithms: a survey of the state of the art. In *Comput. Methods Appl. Mech. Engrg.* 191, 245–1287, 2002.

[6] R.J. Hickman, M. Aldeghi, A. Aspuru-Guzik. Anubis: Bayesian optimization with unknown feasibility constraints for scientific experimentation. In *ChemRxiv*, 2023.

[7] A.K.Y. Low, F. Mekki-Berrada et al. Evolution-guided Bayesian optimization for constrained multi-objective optimization in self-driving labs. In *npj Comput Mater* 10, 104, 2024.

[8] Ma, Z. & Wang, Y. Evolutionary constrained multi-objective optimization: Test suite construction and performance comparisons. In *IEEE Trans. Evol. Comput.* 23, 972–986. 2019.