



HAL
open science

Categorical-Continuous Bayesian optimization applied to chemical reactions (short version)

Theo Rabut, Hamamache Kheddouci, Thomas Galeandro-Diamant

► To cite this version:

Theo Rabut, Hamamache Kheddouci, Thomas Galeandro-Diamant. Categorical-Continuous Bayesian optimization applied to chemical reactions (short version). International Conference on Optimization and Learning (OLA2024), Jul 2022, Syracuse, Italy. hal-04739565

HAL Id: hal-04739565

<https://hal.science/hal-04739565v1>

Submitted on 16 Oct 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Categorical-Continuous Bayesian optimization applied to chemical reactions

Theo Rabut¹, Hamamache Kheddouci¹, and Thomas Galeandro-Diamant²

¹ Université de Lyon, Université Lyon 1, LIRIS UMR CNRS 5205, F-69621, Lyon, France

² ChemIntelligence, Lyon, France

Abstract. Chemical reaction optimization is a challenging field for the industry. Its purpose is to experimentally find reaction parameters (e.g. temperature, concentration, pressure) that maximize or minimize a set of objectives (e.g. yield or selectivity of the chemical reaction). These experiments are often expensive and long (up to several days), making the use of modern optimization methods more and more attractive for chemistry scientists.

Recently, Bayesian optimization has been showed to outperform human decision-making for the optimization of chemical reactions [14]. It is well-suited for chemical reaction optimization problems, for which the evaluation is expensive and noisy.

In this paper we address the problem of chemical reaction optimization with continuous and categorical variables. The presence of categorical variables in an optimization problem often increases its difficulty and decreases the performances of the optimization algorithms.

We propose a Bayesian optimization method with the use of a covariance function initially proposed by Ru *et al.* in the COCABO method [12] and specifically designed for categorical and continuous variables. Also, we experimentally compare different methods to optimize the acquisition function. We establish their performances based on the optimization of two simulated chemical reactions involving categorical and continuous reaction parameters.

We show that the proposed Bayesian optimization algorithm finds optimal reaction parameters in fewer experiments than state of the art algorithms on our simulations.

Keywords: Mixed bayesian optimization · chemical reaction optimization · categorical variables

1 Introduction

Every chemical reaction is optimized before being industrialized. The goal is to find, by carrying out experiments, input parameters (e.g. temperature, pressure, residence time, etc.) that yield optimal values for a set of objectives (e.g. maximize the yield, minimize the production of an impurity, etc.).

The pursuit of high-performance optimization methods is driven by the high cost of chemical experiments. The performances of optimization methods applied

to chemical reactions are measured against the quality of the solution (i.e. how close the solution is to the optimization objectives) and how many experiments are needed to find this solution.

One-Variable-At-a-Time (OVAT) and Design of Experiments (DoE) [1, 16] methods are the most used approaches to optimize chemical reactions. The OVAT method iterates by performing experiments and modifying only one parameter at a time. DoE methods consist in planning a series of experiments following a design matrix, running these experiments and building a statistical model (usually linear or polynomial) with the resulting dataset. An optimum is then computed from the model. OVAT and DoE methods tend to need a large number of experiments to be effective. In addition, OVAT can be very slow (because only one variable is changed at a time) and can get stuck in local optima. Simplex-based methods are also sometimes used to optimize chemical reactions [9, 19]. They consist of building a simplex in the search space, then evaluating the objective function at each of the vertices of the simplex and iteratively displacing one vertex at a time following heuristics. Simplex-based methods tend to be easily stuck in local optima [18].

Zhou *et al.* [21] proposed a deep reinforcement learning (DRL) based method to optimize chemical reactions. The authors combined DRL and pre-training to be able to start working with very small amounts of data. This leads to satisfactory results on problems containing only continuous variables but hasn't been tested with categorical variables (without descriptors).

Bayesian optimization (BO) is a powerful approach to optimize problems for which the evaluations are expensive and noisy. It has shown a variety of successful applications [13]. BO concepts are described in figure 1. First, an initialisation is done with a small number of experiments. Then, a surrogate model (e.g. Gaussian process) is trained using these experiments. An acquisition function, that balances the predicted improvement (exploitation strategy) and the uncertainty of the predictions (exploration strategy), is applied to the model. An optimization algorithm is applied to find the maximum of this acquisition function. The set of parameters that gives this maximal value for the acquisition function determines the next experiment (chemical reaction) to run. This experiment is run, its result is added to the dataset, and the algorithm starts a new iteration. The algorithm stops when the objectives are attained or when the experiments budget is spent.

Categorical variables are often present in the optimization of chemical reactions [11]. We can cite as an example the choice of a catalyst or additives, the choice of the solvent or the order of addition of the reactants. Categorical variables have two important particularities. The first one is the non-continuity constraint, since categorical variables are not defined on a continuous space. The second one is the non-ordinality constraint: they can only be compared with the equality operator. For example, with a categorical variable representing a choice between three solvents *water*, *ethanol*, *toluene* asserting that *water* > *toluene* is meaningless.

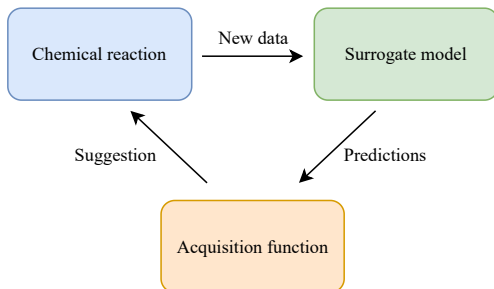


Fig. 1: Simplified Bayesian optimization algorithm applied to chemical reactions.

Mixed-variable optimization can be handled with one-hot encoding: a categorical variable with n categories is encoded as a vector of n corresponding bits, with all bits being equal to 0 except the bit corresponding to the selected category, that is equal to 1. However, in the BO algorithm, treating one-hot dimensions as continuous without any supplementary treatment misleads the acquisition function optimizer and often results in a sub-optimal solution. Indeed, the experiment proposed by the acquisition function optimizer is a real-valued vector and has to be decoded to the closest category. Hence, most of the time, there will be a gap between the experiment suggested by the acquisition function optimizer and the experiment that will actually be performed, leading to a mediocre optimization performance.

The work presented by Garrido-Merchán *et al.* [3] brings an improvement to the basic one-hot encoding approach. During the optimization of the acquisition function, real-valued encoded vectors are transformed to the nearest one-hot vectors before being used as inputs of the model. It follows that the acquisition function optimizer considers real-valued vectors as having the same acquisition values as the associated transformed vectors. Thus, the acquisition optimizer suggests an experiment that can be performed as is, which ensures the convergence to optimal solutions.

Häse *et al.* [4] have developed an augmented Bayesian optimization algorithm called Gryffin that uses a Bayesian neural network as surrogate model. It estimates kernel densities, based on previously evaluated experiments, that are used to approximate the objective function. Gryffin is able to use expert knowledge (descriptors) to guide the optimization, which drastically improves the performances of their method. Its "naive" version doesn't use descriptors, which enabled us to use it in our benchmarks.

COCABO [12] is a Bayesian optimization method designed for mixed-variable optimization. At each iteration, COCABO first selects categories with a multi-armed bandit algorithm and then separately optimizes the numerical variables (after modelling them using a mixed covariance function).

Random forests inherently handle categorical variables and can be used as surrogate model in the Bayesian optimization algorithm [5]. A ready-to-use implementation of this algorithm is provided in a package called SMAC [7].

In this study, we aim at improving the performances (i.e. reducing the number of experiments necessary to reach an optimum) of the Bayesian optimization method for the optimization of chemical reactions with continuous and categorical variables. Our approach is based on Gaussian processes as surrogate models with the COCABO covariance function [12]. We propose different techniques for the optimization of the acquisition function. Next, we compare the different acquisition function optimizer on the optimization of simulated chemical reactions. And finally, we compare our optimization algorithm (using the COCABO covariance function and the highest-performing acquisition function optimizer) with other state-of-the-art algorithms.

2 Problem definition

Our work is applied to problems with a form given by:

$$\text{Minimize } f(\mathbf{z}) \text{ with the smallest possible number of evaluations} \quad (1)$$

where :

- $\mathbf{z} = (\mathbf{x}, \mathbf{h})$
- $\mathbf{x} = x_0, \dots, x_n$ and $x_i \in [A_i, B_i]$ with $A_i, B_i \in \mathbb{R}$
- $\mathbf{h} = h_0, \dots, h_n$ and $h_i \in C_i$ with C_i denotes the categorical space of the i th categorical variable.

This work is restricted to single objective optimization. Moreover, only continuous and categorical variables are used.

The "No-Free Lunch Theorem" [17] stipulates that the performances of every optimization methods are equal when averaged on all possible problems. It implies that in order to increase the performances on a specific optimization problem (e.g. chemical reaction optimization), we must evaluate the optimization method on similar problems without any regards on the performances of unrelated ones. The underlying functions of chemical reactions have some particularities: they are smooth and have few local optima [8, 15]. So, in order to be specific to the chemical reaction optimization problem, we measure the performances of our approach using chemical reaction simulators. We have built these chemical reaction simulators by training machine learning models with publicly available chemical reaction data (see table 1). This benchmarking strategy was initially introduced by Felton *et al.* [2] for measuring performances on chemical reactions with continuous and categorical variables. It allows us to establish optimization performances on chemical reactions without having to run experiments in a chemistry lab.

Table 1: Details of the data used to train the simulators

Reaction type	Number of experiments	Source
Pd-catalysed direct arylation	1728	[14]
Suzuki-Miyaura cross-coupling	4 cases of 96	[11, 2]

3 Propositions

In a first part, we describe the surrogate model including the COCABO kernel and its hyperparameters. In a second part, we present different approaches for the optimization of the acquisition function.

3.1 Gaussian process kernel

We use Gaussian processes (GP) to approximate the underlying functions of chemical reactions. It is the most commonly used model since it can inherently predict both a value and an associated uncertainty. Gaussian processes are mainly defined by their covariance function. Since the underlying functions of chemical reactions are smooth, we use a smooth covariance function, Matérn_{5/2} [10], for the continuous dimensions.

The smoothness of the GP on continuous variables is kept with the use of the one-hot encoding. However, the Euclidian distance used for the calculation of the Matérn_{5/2} kernel is based on all dimensions (continuous and encoded). We believe that, in order to capture complex relationships between categorical and continuous variables, the covariance function should use the Euclidian distance only on continuous variables and incorporate categorical knowledge later in its calculation. The COCABO method [12] uses such a covariance function (see equation 2). It combines two sub-functions: one for continuous variables, K_{cont} , and one for categorical variables, K_{cat} .

$$K(\mathbf{z}, \mathbf{z}') = (1 - \lambda) \times (K_{cont}(\mathbf{x}, \mathbf{x}') \times K_{cat}(\mathbf{h}, \mathbf{h}')) + \lambda \times (K_{cont}(\mathbf{x}, \mathbf{x}') + K_{cat}(\mathbf{h}, \mathbf{h}')) \quad (2)$$

where :

- $\mathbf{z} = (\mathbf{x}, \mathbf{h})$
- \mathbf{x} is the set of continuous variables
- \mathbf{h} is the set of categorical variables

K_{cont} is the Matérn_{5/2} function. It is a standard covariance function for smooth Gaussian processes regressions with continuous inputs. K_{cat} , the kernel for categorical inputs (see equation 3), measures similarity between categorical vectors with the equality operator (which is the only permitted operation for categorical variables).

$$K_{cat}(\mathbf{h}, \mathbf{h}') = \sigma \times \frac{1}{D} \sum_1^D \alpha(h_d, h'_d) \quad (3)$$

where:

- $\alpha(a, b)$ equals 1 if $a = b$ and 0 if $a \neq b$
- D is the number of categorical variables
- σ is the variance hyperparameter.

The proposition made by Ru *et al.* in COCABO [12] revolves around the hyperparameter λ , which is a trade-off between the two terms of the equation 2: the sum and the product of K_{cont} and K_{cat} . Both of these terms capture different relationships between continuous and categorical variables. The sum of the two sub-kernels produces a learning of a single trend on the continuous variables and shift this trend depending on the categories whereas the product is able to produce a learning of complex relationships with highly different trends depending on the categories. The sum is especially necessary when the amount of training data is low (beginning of the optimization) because the product is able to capture knowledge only if the evaluations have categories in common. For example, if two evaluations have the same continuous features but different categorical ones, the product will be equal to 0 which prevent the model to learn even on continuous variables. Nonetheless, the product is essential because, as the optimization goes on, more evaluations are added to the training dataset and a single trend with a simple shift will not be sufficient to model the complexity offered by the data. In other words the sum alone will not be able to capture all the knowledge available to guide the optimization. With the hyperparameter λ , the authors ensure that the relationships that can be captured either by the sum or by the product are taken into account into the covariance $K(\mathbf{z}, \mathbf{z}')$, because λ is tuned during the fitting of the Gaussian process.

In order to avoid underfitting/overfitting the data while training the Gaussian process (tuning its hyperparameters to minimize its negative log marginal likelihood [10]), we confined hyperparameter values within a range. σ_K , $\sigma_{K_{cont}}$ and $\sigma_{K_{cat}}$ were bounded in $[10^{-2}, 20]$ while the lengthscale parameter of K_{cont} and λ were respectively bounded in $[10^{-2}, 20]$ and $[0.1, 0.9]$. We used the L-BFGS optimizer to tune the GP hyperparameters.

3.2 Acquisition function optimization

We chose to use the Expected Improvement (EI) acquisition function because it has shown good results on diverse applications and has a strong theoretical support [20]. The equation of Expected Improvement is given by:

$$\text{EI}(\mathbf{x}) = \mathbb{E}[\max(f(\mathbf{x}) - f(\mathbf{x}^+), 0)] \quad (4)$$

with $f(\mathbf{x}^+)$ the value of the evaluation that have yielded the best result so far. The analytical form of EI is the following:

$$\text{EI}(\mathbf{x}) = \begin{cases} \sigma(\mathbf{x})Z\Phi(Z) + \sigma(\mathbf{x})\phi(Z) & \text{if } \sigma(\mathbf{x}) \neq 0 \\ 0 & \text{if } \sigma(\mathbf{x}) = 0 \end{cases} \quad (5)$$

where

$$Z = \frac{\mu(\mathbf{x}) - f(\mathbf{x}^+) - \xi}{\sigma(\mathbf{x})} \quad (6)$$

$\Phi(Z)$ and $\phi(Z)$ denotes respectively the cumulative distribution function (CDF) and the probability density function (PDF) of the variable Z . Z denotes the predicted improvement divided by the standard deviation (uncertainty) and the parameter ξ determines the weight of the exploration strategy in the equation. This analytical form of EI is cheap to evaluate and can be optimized without sparing on the number of evaluations. Therefore, we propose several approaches for the optimization of the acquisition function with mixed variables.

The first approach (denoted as L-BFGS-OHE) involves a one-hot encoding of the categorical variables and a multi-started gradient descent for the optimization of the acquisition function. However, since the COCABO model do not accept one-hot vectors, one-hot dimensions are systematically decoded before any predictions. In other words, predictions are asked for by the acquisition function optimizer with encoded inputs but they are decoded before they pass through the model. The multi-started gradient descent is performed as follows: 1000 configurations are randomly drawn and the 5 configurations with the highest acquisition function value are kept and a gradient descent (L-BFGS) is performed on each of these 5 configurations.

We also propose an approach based on a "brute-force" optimization of the categorical space and a multi-started gradient descent on the continuous space (see Algorithm 1). First, all the combinations of the categorical parameters are constructed. Then, for each combination, a multi-started gradient descent (previously described) is performed on the continuous parameters. Finally, after determining the maximal acquisition values for each categorical combination, the configuration with the highest acquisition value is suggested as the next experiment. This algorithm reduces the difficulty of the optimization of the acquisition function because instead of dealing with different types of variables (or with supplementary dimensions from the encoding), the acquisition optimizer only works on the continuous dimensions. Still, it can be heavy in terms of computational cost if the number of categories and categorical variables is large.

Algorithm 1 Categorical brute-force and multi-started gradient descent

- 1: Construct all categorical combinations
 - 2: Multi-started gradient descent optimization of continuous parameters for each combination
 - 3: Choose as suggestion the configuration (continuous and categorical) with the highest acquisition
-

Lastly, we implemented an evolutionary algorithm based on ant colonies (ACO) that can handle categorical variables [6]. In our experiments, we used the colony hyperparameters proposed by the authors without any restart allowed.

This algorithm is a multi-agent method inspired by the behaviour of ants. An ant represent an evaluation at a given set of parameters. At each generation, each ant randomly moves towards previously evaluated points with good results (exploitation strategy). The presence of multiple ants in the colony and the randomness of their movements enable the mandatory exploration of the search space. It allows the ants to not only moves around promising areas but also randomly explore areas that may have not been explored so far.

4 Results

For each optimization algorithm, we performed 25 runs of 55 experiments each. At each run, we randomly drew 5 initial evaluations and, for a fairness purpose, these 5 evaluations were used to initialize all the optimizers.

First, we compare the performances of different acquisition optimization techniques (using the COCABO kernel and the EI acquisition function).

The figures 2a and 2b compare the performances of the acquisition optimizers. They correspond respectively to a simulation of the direct arylation reaction which contains 3 categorical variables and 2 continuous ones, and to a simulation of the Suzuki-Miyaura reaction (case 1) which contains 1 categorical variables and 3 continuous ones.

In both cases, ACO performs poorly compared to the two other methods. Still, its performances are closer to the two other optimizers than the random strategy so it will be the subject of further work to exploit the potential of the ACO method.

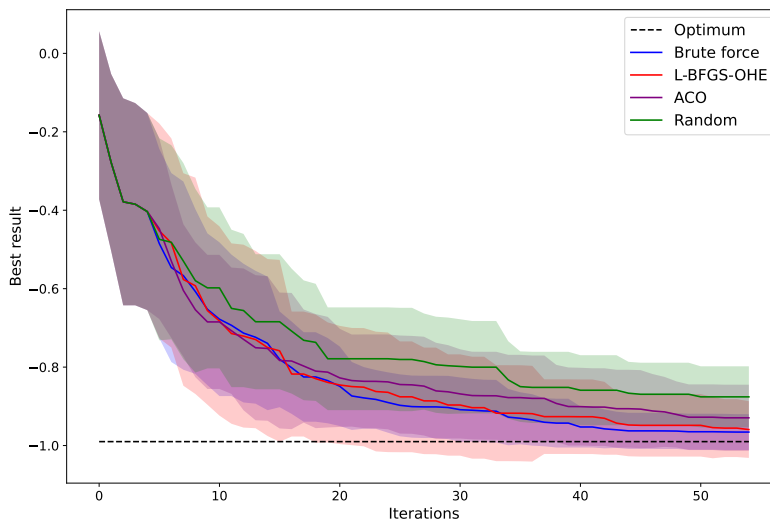
In the figure 2a, the brute-force and the L-BFGS algorithm with one-hot encoded categorical variables give similar results but in the figure 2b, brute-force performs slightly better. Overall, the brute-force approach offers the best performances with the steepest average convergence rate and the lowest standard deviation (filled area).

As consequence of the results presented above, we chose the brute-force approach to be the acquisition function optimizer in the rest of our study.

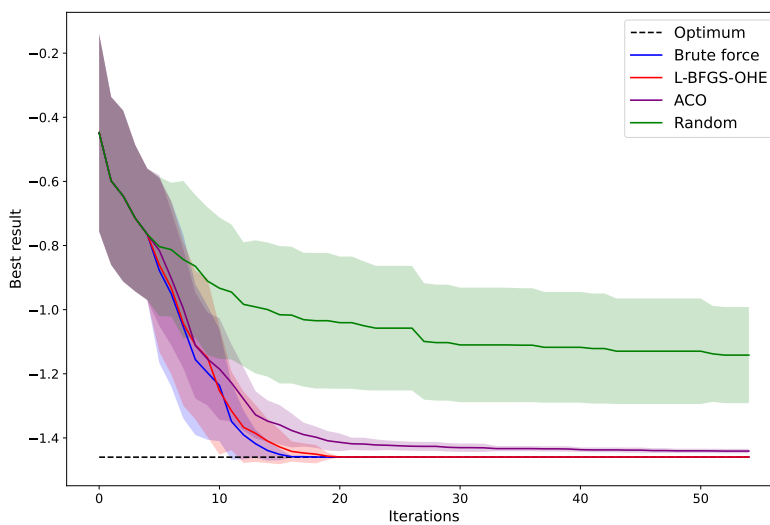
The next results present a comparison between our method (composed by the COCABO kernel, Expected Improvement and the brute-force optimizer), Gryffin [4], COCABO [12], SMAC [5], and the work of Garrido-Merchán *et al.* [3].

We used the "naive" version of Gryffin in its authors' implementation. We used COCABO in its authors' implementation with its default settings and a starting $\lambda = 0.5$. SMAC denotes an optimization algorithm based on Random Forest [5] and the Expected Improvement acquisition function. We used an implementation proposed by Lindauer *et al.* [7].

In Figures 3a and 3b, the Bayesian optimization with the COCABO kernel and the categorical brute-force optimization of the acquisition function gives the best results: it generally converges faster to the optimum than other methods.

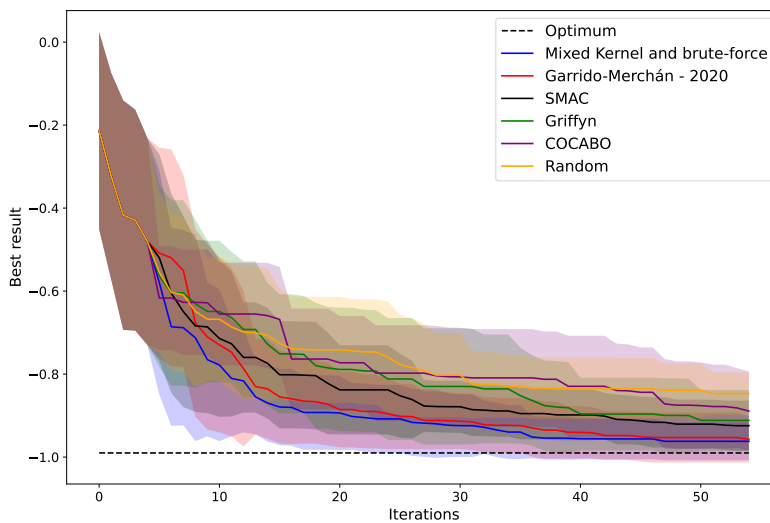


(a) Benchmark function: Pd-catalysed direct arylation simulation

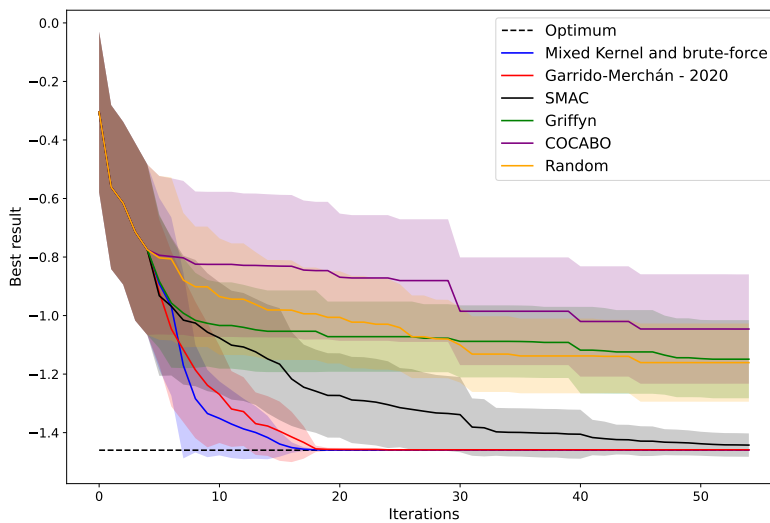


(b) Benchmark function: Suzuki-Miyaura simulation

Fig. 2: Best score evolution on simulations with the use of different acquisition function optimizers (brute-force, ACO, L-BFGS-OHE). A random optimization strategy of each chemical reaction simulation is given.



(a) Benchmark function: Pd-catalysed direct arylation simulation



(b) Benchmark function: Suzuki-Miyaura simulation (case 1)

Fig. 3: Best score evolution on simulations with different optimization methods

SMAC’s performances show us that it can handle categorical variables but, overall, it performs poorly compared to the two Bayesian optimization with Gaussian process as surrogate model.

The COCABO method fails to optimize the Suzuki-Miyaura simulation. The multi-armed bandit (MAB) part of the COCABO method is designed for multiple categorical variables with multiple categories and the Suzuki-Miyaura simulation has only one categorical variable.

Our algorithm (“Mixed kernel and brute-force”) performs slightly better than the work of Garrido-Merchán *et al.*. The main difference between the two methods is the use of different covariance functions. The COCABO kernel is able to capture more complex relationships than a standard Matérn_{5/2} function on a one-hot encoded space.

5 Conclusion

This paper presents a method for the optimization of chemical reactions with mixed variables (continuous and categorical).

We expose a Bayesian optimization algorithm based on a Gaussian process with a covariance function specifically designed for continuous and categorical variables [12]. Also, we evaluate different methods for the optimization of the acquisition function and show that a brute-force approach associated to a multi-started gradient descent performs best. This approach performs globally better than other state-of-the-art methods [4, 3, 12] on two simulated chemical reactions with categorical and continuous inputs.

We are working on further increasing the quality of the model by modifying the covariance function. Also, in order to fully establish the performance of the presented method, later works will imply experimental validation in chemistry labs.

Acknowledgement: This work was supported by the R&D Booster SMAPI project 2020 of the Auvergne-Rhône-Alpes Region.

References

1. Carlson, R., Carlson, J.E.: Design and optimization in organic synthesis. Elsevier (2005)
2. Felton, K., Rittig, J., Lapkin, A.: Summit: Benchmarking machine learning methods for reaction optimisation (2020)
3. Garrido-Merchán, E.C., Hernández-Lobato, D.: Dealing with categorical and integer-valued variables in bayesian optimization with gaussian processes. *Neurocomputing* **380**, 20–35 (2020)
4. Häse, F., Aldeghi, M., Hickman, R.J., Roch, L.M., Aspuru-Guzik, A.: Gryffin: An algorithm for bayesian optimization of categorical variables informed by expert knowledge. *Applied Physics Reviews* **8**(3), 031406 (2021)

5. Hutter, F., Hoos, H.H., Leyton-Brown, K.: Sequential model-based optimization for general algorithm configuration. In: International conference on learning and intelligent optimization. pp. 507–523. Springer (2011)
6. Liao, T., Socha, K., de Oca, M.A.M., Stützle, T., Dorigo, M.: Ant colony optimization for mixed-variable optimization problems. *IEEE Transactions on Evolutionary Computation* **18**(4), 503–518 (2013)
7. Lindauer, M., Eggensperger, K., Feurer, M., Biedenkapp, A., Deng, D., Benjamini, C., Ruhkopf, T., Sass, R., Hutter, F.: Smac3: A versatile bayesian optimization package for hyperparameter optimization. In: ArXiv: 2109.09831 (2021), <https://arxiv.org/abs/2109.09831>
8. Moore, K.W., Pechen, A., Feng, X.J., Dominy, J., Beltrani, V.J., Rabitz, H.: Why is chemical synthesis and property optimization easier than expected? *Physical Chemistry Chemical Physics* **13**(21), 10048–10070 (2011)
9. Morgan, S.L., Deming, S.N.: Simplex optimization of analytical chemical methods. *Analytical chemistry* **46**(9), 1170–1181 (1974)
10. Rasmussen, C.E., Nickisch, H.: Gaussian processes for machine learning (gpml) toolbox. *The Journal of Machine Learning Research* **11**, 3011–3015 (2010)
11. Reizman, B.J., Wang, Y.M., Buchwald, S.L., Jensen, K.F.: Suzuki–miyaura cross-coupling optimization enabled by automated feedback. *Reaction chemistry & engineering* **1**(6), 658–666 (2016)
12. Ru, B., Alvi, A., Nguyen, V., Osborne, M.A., Roberts, S.: Bayesian optimisation over multiple continuous and categorical inputs. In: International Conference on Machine Learning. pp. 8276–8285. PMLR (2020)
13. Shahriari, B., Swersky, K., Wang, Z., Adams, R.P., De Freitas, N.: Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE* **104**(1), 148–175 (2015)
14. Shields, B.J., Stevens, J., Li, J., Parasram, M., Damani, F., Alvarado, J.I.M., Janey, J.M., Adams, R.P., Doyle, A.G.: Bayesian reaction optimization as a tool for chemical synthesis. *Nature* **590**(7844), 89–96 (2021)
15. Tibbetts, K.M., Feng, X.J., Rabitz, H.: Exploring experimental fitness landscapes for chemical synthesis and property optimization. *Physical Chemistry Chemical Physics* **19**(6), 4266–4287 (2017)
16. Weissman, S.A., Anderson, N.G.: Design of experiments (doe) and process optimization. a review of recent publications. *Organic Process Research & Development* **19**(11), 1605–1633 (2015)
17. Wolpert, D.H., Macready, W.G., et al.: No free lunch theorems for search. Tech. rep., Technical Report SFI-TR-95-02-010, Santa Fe Institute (1995)
18. Wright, M.H., et al.: Nelder, mead, and the other simplex method. *Documenta Mathematica* **7**, 271–276 (2010)
19. Xiong, Q., Jutan, A.: Continuous optimization using a dynamic simplex method. *Chemical Engineering Science* **58**(16), 3817–3828 (2003)
20. Zhan, D., Xing, H.: Expected improvement for expensive optimization: a review. *Journal of Global Optimization* **78**(3), 507–544 (2020)
21. Zhou, Z., Li, X., Zare, R.N.: Optimizing chemical reactions with deep reinforcement learning. *ACS central science* **3**(12), 1337–1344 (2017)