Closed-loop automatic gradient design for liquid chromatography using Bayesian optimization

Boelrijk, J.; Ensing, B.; Forré, P.; Pirok, B.W.J.

DOI
10.1016/j.aca.2023.340789

Publication date
2023

Document Version
Final published version

Published in
Analytica Chimica Acta

Citation for published version (APA):
Boelrijk, J., Ensing, B., Forré, P., & Pirok, B. W. J. (2023). Closed-loop automatic gradient design for liquid chromatography using Bayesian optimization. *Analytica Chimica Acta*, *1242*, [340789]. https://doi.org/10.1016/j.aca.2023.340789

Analytica Chimica Acta 1242 (2023) 340789

Contents lists available at ScienceDirect

Analytica Chimica Acta

journal homepage: www.elsevier.com/locate/aca

Closed-loop automatic gradient design for liquid chromatography using Bayesian optimization

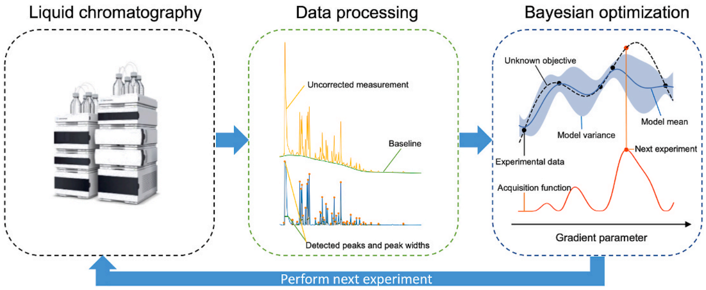
Jim Boelrijk [a,b,*], Bernd Ensing [a,c], Patrick Forré [a,b], Bob W.J. Pirok [a,d,**]

[a] *AI4Science Lab, Informatics Institute, University of Amsterdam, Amsterdam, Science Park 904, 1098 XH, the Netherlands*
[b] *AMLab, Informatics Institute, University of Amsterdam, Amsterdam, Science Park 904, 1098 XH, the Netherlands*
[c] *Computational Chemistry Group, Van 't Hoff Institute for Molecular Sciences, University of Amsterdam, Amsterdam, Science Park 904, 1098 XH, the Netherlands*
[d] *Analytical Chemistry Group, Van 't Hoff Institute for Molecular Sciences, University of Amsterdam, Amsterdam, Science Park 904, 1098 XH, the Netherlands*

HIGHLIGHTS

- A Bayesian optimization algorithm was developed to optimize LC gradient programs.
- The algorithm can operate in both the single- and multi-objective setting.
- The algorithm was successfully applied to the optimization of two complex dye mixtures.
- The performance of single-versus multi-objective optimization was compared.

GRAPHICAL ABSTRACT



ARTICLE INFO

Handling Editor: Prof. L. Buydens

Keywords:
Bayesian optimization
Liquid chromatography
Closed-loop method development
Machine learning
Experimental design

ABSTRACT

Contemporary complex samples require sophisticated methods for full analysis. This work describes the development of a Bayesian optimization algorithm for automated and unsupervised development of gradient programs. The algorithm was tailored to LC using a Gaussian process model with a novel covariance kernel. To facilitate unsupervised learning, the algorithm was designed to interface directly with the chromatographic system. Single-objective and multi-objective Bayesian optimization strategies were investigated for the separation of two complex (n>18, and n>80) dye mixtures. Both approaches found satisfactory optima in under 35 measurements. The multi-objective strategy was found to be powerful and flexible in terms of exploring the Pareto front. The performance difference between the single-objective and multi-objective strategy was further investigated using a retention modeling example. One additional advantage of the multi-objective approach was that it allows for a trade-off to be made between multiple objectives without prior knowledge. In general, the Bayesian optimization strategy was found to be particularly suitable, but not limited to, cases where retention modelling is not possible, although its scalability might be limited in terms of the number of parameters that can be simultaneously optimized.

---

[*] Corresponding author. AI4Science Lab, Informatics Institute, University of Amsterdam, Amsterdam, Science Park 904, 1098 XH, the Netherlands.

[**] Corresponding author. Analytical Chemistry Group, Van 't Hoff Institute for Molecular Sciences, University of Amsterdam, Amsterdam, Science Park 904, 1098 XH, the Netherlands.

E-mail addresses: j.h.m.boelrijk@uva.nl (J. Boelrijk), bob.pirok@uva.nl (B.W.J. Pirok).

https://doi.org/10.1016/j.aca.2023.340789
Received 29 July 2022; Received in revised form 13 December 2022; Accepted 2 January 2023

1. Introduction

Liquid chromatography is a powerful and popular method for the separation of a mixture into its individual constituents. Gradient elution, in which the mobile phase composition is programmed to change over time, is the technique of choice for the separation of complex mixtures exhibiting a wide range of retention factors. Especially, because it leads to a decrease in analysis time and peak compression. The temporal changes to the mobile phase composition are often defined in a multi-linear gradient program, which contains the coordinates (time and modifier concentration) of the transition points between consecutive linear segments, also called the gradient nodes (or gradient parameters). For gradient-elution based methods, one key objective of method development is thus to optimize the parameters of the gradient program (and/or other experimental controls) against an objective function(s), also called chromatographic response function(s) (CRFs) [1].

The search for the optimal combination of the parameters is a combinatorial optimization problem, the difficulty of which increases exponentially with the number of parameters to be optimized. As an example, if there are 10 parameters (just 5 gradient nodes) that can take on 20 values each, the number of combinations is $20^{10}(\sim10^{13})$, an astronomical figure. Therefore, efficient optimization methods that incorporate domain knowledge are required to be able to cope with a large number of parameters.

Developing such methods has been an active field of research over the past decades. One strategy is retention modeling, in which the retention behavior of every analyte in a mixture is described by separate prediction models based on an interpolation of retention times obtained from a number of scanning experiments [2–4]. To construct these models, it is key that analytes can be tracked over multiple experiments and that the predicted experiments (computed *in silico*) match the "wet" experiments well, at least in terms of peak width and retention time [5]. Then this retention model can be used to predict the performance of future experiments that might be optimal, e.g. using a grid search [2], a genetic algorithm [6,7], gradient descent optimization [5] or Bayesian optimization [8]. The above strategy allows separations to be predicted a priori, yet suffers from the dependency of peak tracking and the reliability of the retention model.

Other strategies strictly rely on modeling the objective function directly through"wet" experimental evaluation. In these methods, peak-tracking is not required. Here, pioneering methods utilized simplex optimization [9,10]. Later methods utilized genetic algorithms in a closed-loop fashion for GC-MS [11] and LC-MS/MS [12]. However, for such approaches, as every experiment needs to be physically executed (i.e., "wet"), this approach is more costly and time-consuming. Thus, the first challenge is to efficiently establish complex gradient programs capable of separating contemporary challenging samples in as few "wet" experiments as possible, which we refer to as data efficiency.

Besides this intricate optimization problem, a second challenge lies within the formulation of a numerical optimization goal (i.e., the objective function). While this can be described by a single objective, such as the degree of separation between peaks, method development typically is multi-objective in nature. For example, objectives might include the number of peaks, the separation time, the peak capacity, etc. [11] In this case, one is often interested in Pareto optimal solutions for which any improvement in one objective means deteriorating another. The set of these solutions forms the Pareto front. Provided with this set, decision-makers can select a solution with an objective trade-off according to their needs. Multi-objective problems can be reformulated (scalarized) into a single-objective by assigning different weights to the objectives and (for example) applying a summation. This has been done in various ways in the chromatography literature [7,13–16]. For instance, Berridge et al. [16] combined the resolution between peaks, the number of peaks, the retention time of the last eluted peak, and the retention time of the first eluted peak using three arbitrarily chosen weights. However, it can be non-trivial to obtain an adequate set of weights where every Pareto optimal solution can be reached, especially if the complexity of a sample is unknown a priori. In addition, different weights can lead to different Pareto optimal solutions.

Another approach is to maximize the hypervolume indicator (HVI), which measures the hypervolume (area in two dimensions, or volume in three dimensions) between the Pareto front and a reference point. Maximizing the HVI has been shown to produce excellent coverage of the Pareto front over a range of problems [17–19]. In addition, multi-objective evolutionary algorithms such as PESA-II and NSGA-II have been developed to provide effective approximations of the Pareto frontier. The former has been utilized in the multi-objective optimization of three objectives (number of peaks, run time, and signal/noise ratio) of a human serum sample in GC-MS [11]. As well as in targeted metabolite analysis in LC-MS/MS [12]. Moreover, the latter two works were fully automated using visual scripts that imitate keyboard and mouse commands.

Resolving the above two challenges of data efficiency and balance between (conflicting) optimization objectives, would allow extremely rapid method development. One particularly attractive benefit is that this can happen in a closed-loop fashion, for example, by a system that (1) controls the parameters of the LC machine, (2) performs the measurement, (3) evaluates the objective function(s), and based on this, (4) proposes and performs the next experiment, until some optimization criterion is met, without any human intervention. We earlier investigated the feasibility of Bayesian optimization to address these challenges [8]. Our simulations markedly outperformed grid search and random search approaches, and suggested *in silico* that the technique could be used to optimize gradient programs for both 1D- and 2D-LC. In addition, other works also have consistently shown that Bayesian optimization outperforms evolutionary algorithms (NSGA-II and PESA-II) in terms of sample efficiency [17,20,21]. Therefore, in this work, we developed a novel algorithm based on Bayesian optimization for the closed-loop automated development of gradient-based methods in LC. To achieve this, we designed a unique covariance kernel that elegantly incorporates domain knowledge of the gradient-program shape to improve efficiency in finding a satisfactory optimum. Moreover, our modular strategy is demonstrated for both single-objective and multi-objective Bayesian optimization. The resulting closed-loop platform was used to optimize the separation of two complex dye mixtures. In addition, we assessed the performance between single-objective and multi-objective optimization on a retention modelling example.

2. Theory

2.1. Objective function

The concept of resolution is commonly used to describe the extent of separation between successive peaks in a chromatogram. However, the resolution is usually defined in a way that is applicable only to symmetrical (Gaussian) peaks with similar peak areas. Schoenmakers et al. identified this problem and proposed modifications to the general resolution equation to correct for large variations in peak areas and peak asymmetry [22]. The authors defined peak asymmetry using the asymmetry factor A_s, which for a peak i is given by:

$$A_{s,i} = \frac{a_i}{b_i} \tag{1}$$

where a_i and b_i are widths of the ascending and descending slopes of the peak, relative to the peak apex, respectively. The widths are measured at a height of 0.135 relative to the reference height h_i (4σ for Gaussian peaks). This is illustrated in Fig. 1 for two peaks i and j.

Given the asymmetry of a peak i, the plate count can be defined as:

$$N_i = 16\left(\frac{t_{R,i}}{a_i + b_i}\right)^2 \tag{2}$$

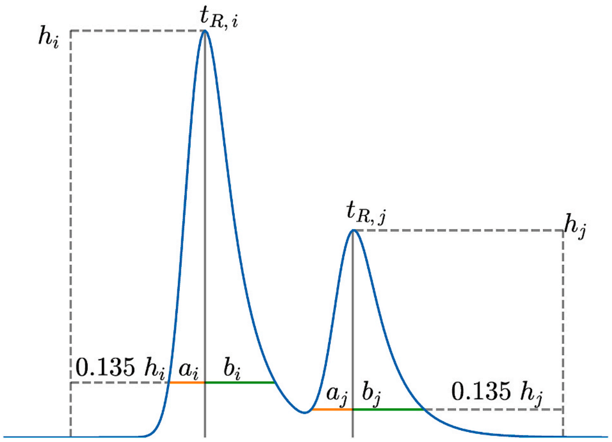


Fig. 1. Illustration of the relevant parameters determining the resolution between two (potentially) asymmetric peaks *i* and *j*. h_i, $t_{R,i}$, a_i, b_i, h_j, $t_{R,j}$, a_j, b_j.

where $t_{R,i}$ is the retention time for peak *i*. By accounting for peak asymmetry, the resolution between two peaks *i* and *j* can then be defined as follows:

$$ {}^{i}Rs_{i,j} = \frac{\left(t_{R,j}-t_{R,i}\right)\left(1+A_{s,i}\right)\left(1+A_{s,j}\right)\sqrt{N_iN_j}}{4A_{s,i}t_{R,i}\left(1+A_{s,j}\right)\sqrt{N_j}+4t_{R,j}\left(1+A_{s,i}\right)\sqrt{N_i}\sqrt{1+{}^{1}/_{2}\ln\left(h_j/h_i\right)}} \tag{3} $$

an important consequence of these modifications is that the resolution for a pair of peaks has two different values, one for each peak. For the second peak (*j*), the corresponding equation is:

$$ {}^{j}Rs_{i,j} = \frac{\left(t_{R,j}-t_{R,i}\right)\left(1+A_{s,i}\right)\left(1+A_{s,j}\right)\sqrt{N_iN_j}}{4A_{s,i}t_{R,i}\left(1+A_{s,j}\right)\sqrt{N_j}\sqrt{1+{}^{1}/_{2}\ln\left(h_j/h_i\right)}+4t_{R,j}\left(1+A_{s,i}\right)\sqrt{N_i}} \tag{4} $$

in this work, we compute values both for Eq. (3) and Eq. (4) and select the lowest value, as the lowest value is the most important indicator of quality of the separation. This value will be abbreviated as $Rs_{i,j}$. Next the resolution is normalized to a value between 0 and 1 using the following desirability function:

$$ d\left(Rs_{i,j}\geq 1\right)=\begin{cases} 0 \ \text{ if } \ Rs_{i,j}\leq 1 \\ \dfrac{Rs_{i,j}}{1.5} \ \text{ if } \ 1\leq Rs_{i,j}\leq 1.5 \\ 1 \ \text{ if } \ Rs_{i,j}\geq 1.5 \end{cases} \tag{5} $$

note that the equation is capped to a resolution of 1.5, and thus higher resolution values are not rewarded more. Also, here a lower limit of the resolution was chosen to be 1 (as stated in $d(Rs_{i,j}\geq 1)$). Finally, for a chromatogram consisting of *P* peaks, the overall separation quality is given by:

$$ O_{Rs}=\sum_{i=1}^{P-1} d\left(Rs_{i,i+1}\right) \tag{6} $$

in this way, the resolution between all neighboring peaks is computed and summed over.

2.2. Parameters of a multilinear gradient program

Fig. 2 illustrates a typical example of a multilinear gradient profile. The gradient nodes (φ_i, t_i) define the coordinates of the transition points between consecutive linear segments, and are depicted by blue dots. Note that the initial and final gradient nodes (depicted by black dots) are kept fixed throughout all experiments at an initial modifier concentration of 0% modifier solvent B for 0.25 min, and a final modifier concentration of 100% B at $t = 20$ min (for this example specifically), respectively. By adding more gradient nodes (effectively adding two parameters per node), more complex gradient programs can be constructed. We only allowed for positive or flat (isocratic) gradient steps by incorporating the following constraints: $\varphi_i \leq \varphi_{i+1}$, and $t_i \leq t_{i+1}$.

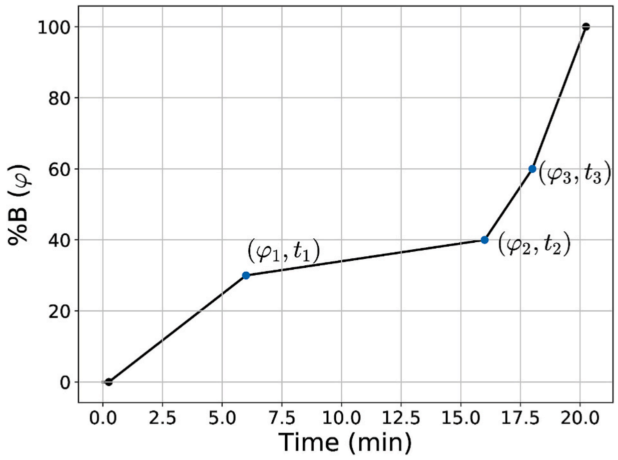


Fig. 2. Example of a multilinear gradient profile. Gradient nodes are depicted by blue dots and define the transition points between consecutive linear segments. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

3. Experimental

3.1. Chemicals

Milli-Q water (18.2 MΩ cm) was obtained from a purification system (Arium 611UV, Sartorius, Germany). Acetonitrile (LC-MS grade) was obtained from Biosolve (Valkenswaard, The Netherlands). triethylamine ($\geq$ 99.5%), and formic acid (reagent grade, $\geq$ 95%) were obtained from Sigma-Aldrich (Darmstadt, Germany). Different dyestuffs and the extracts of the samples from historical objects were obtained from the Cultural Heritage Agency of the Netherlands (RCE, Amsterdam, The Netherlands).

3.2. Instrumental

All experiments in this study were performed using an Agilent 1290 Infinity 2D-LC system (Waldbronn, Germany) configured for one-dimensional operation. The system was comprised of a binary pump (G7120A), an autosampler (G7129B) equipped with a 20 μL injection loop, a thermostatted column compartment (G7116B), and a diode-array detector (DAD, G7117B) with a sampling frequency of 240 Hz equipped with an Agilent Max-Light Cartridge Cell (G4212-60008, 10 mm path length, $V_{\text{det}} = 1.0$ μL. UV–vis spectra were recorded from 190.0 nm to 640.0 nm with 0.5 nm steps. The needle was set to draw at a speed of 100 μL min^{-1} and eject at a speed of 400 μL min^{-1} and to allow 1.2 s of equilibration time. In addition, a needle program with a wait time of 2 min was used. After each measurement, the column was flushed at 100% mobile phase B 1.5 min followed by 1.5 min 0% mobile phase B. During data export, the flow rate was lowered to 0.03 mL min^{-1} to reduce mobile phase use. To protect the column, an Agilent 1290 Infinity in-line filter was used in front of the column. The system was controlled using Agilent OpenLAB CDS Chemstation Edition (rev. C.01.10 [287]). An Agilent Poroshell 120 SB-C18 (USCFH10021, 2.1 × 100 mm i.d., $d_p = 2.7$ μm, max $\Delta P = 1200$ bar) was used. The flow rate was set to 0.65 mL min^{-1}. The injection volume was set to 1.0 μL. The temperature of the column oven was set to 40 °C.

3.3. Sample preparation

The mobile phase consisted of a aqueous 5 mM triethylamine (TEA) buffer, brought to pH = 3.0 with formic acid, and acetonitrile (ACN) in a ratio of 95:5 TEA buffer/ACN (v/v, mobile phase A) and 5:95 TEA buffer/ACN (v/v, mobile phase B).

Sample A was a solution of 18 degraded dyestuffs shown in the Supplementary Materials. Each dye was diluted, in concentrations of approximately 50–60 ppm by weight, in a mixture of ACN and DMSO 1:1 (v/v). Sample B comprised of 80 samples of dyestuffs. The actual composition was more complex because possible impurities and degradation products are unknown. Each sample was diluted into a stock solution, in concentrations of 5000 ppm by weight, in a mixture of ACN and DMSO 1:1 (v/v). A mixture of these 80 dyestuffs was created by mixing all the 5000 ppm stock solutions in equal ratio, resulting in a solution containing all 80 dyes, their contaminants, and degradation products, in a concentration of approximately 60 ppm for each dye. This sample originated from earlier work and a full list of the dyes and their molecular structures are described in the appendix of reference [23]. We then further diluted this mix to approximately 30 ppm in a mixture of ACN and DMSO (1:1) (v/v).

3.4. Retention modeling framework

For the retention modeling examples discussed in Sections 4.1 and 4.5 we use in-house Python code provided in the Supplementary Materials. In this code, we used the linear solvent strength retention model [4], within the framework of multi-linear retention modeling as described in Nikitas et al. [14]. We describe peak widths using the model of Hao and co-workers [24], which corrects for gradient compression effects. For both experiments we use a plate number of 1000. For the experiments in Section 4.1, we uniformly sampled ln k_0 values between 1 and 10 and S values between 8 and 20 for 20 compounds. We used a column dead time of 0.15 min and a dwell time of 0.15 min. For the experiments in Section 4.5, we report retention parameters in the Supplementary Material and use 0.8 min for both the column dead time and dwell time.

3.5. Bayesian optimization algorithm

The Bayesian optimization algorithm was implemented in Python using the BoTorch [25] and GPyTorch [26] packages. The sorting Matérn 5/2 kernel was developed in-house and is compatible with GPyTorch. Details regarding this kernel and how its hyperparameters are estimated can be found in Section S-1.2 of the Supplementary Information. We provide additional theoretical information on Gaussian processes, and the acquisition functions mentioned in Section 3.5.1 and 3.5.2 in Sections S-1.1 and S-1.3 of the Supplementary Information.

3.5.1. Single-objective Bayesian optimization model

For the single-objective Bayesian optimization model described in Section 4.3 we used the expected improvement acquisition function [27]. The acquisition function was optimized using multi-start optimization on 20 starting points drawn from an initial grid of 512 points.

3.5.2. Multi-objective Bayesian optimization model

For the multi-objective Bayesian optimization model described in Section 4.4 we used the q-Expected Hypervolume Improvement acquisition function [17]. The acquisition function was optimized using multi-start optimization with 20 restarts drawn from an initial grid of 512 points.

4. Results and discussion

4.1. Design of the Bayesian optimization framework

Bayesian optimization is an established and efficient machine learning technique for the multivariate optimization of expensive-to-evaluate black-box functions [20]. At its heart, a probabilistic surrogate model of the objective function $f(\mathbf{x})$ (the observable to be optimized) is constructed given it's input parameters $\mathbf{x}$ (e.g. $\mathbf{x} = [\varphi_1, \varphi_2, \varphi_3, t_1, t_2, t_3]$). This surrogate model is then used to guide the optimization process. The model is a prior probability distribution over all possible objective functions, representing our belief about the function's properties. This distribution is often described by a Gaussian process model, in a technique called Gaussian Process Regression (GPR) [28]. The prior distribution is updated with each new measurement to produce a more accurate posterior distribution. The Gaussian process can then provide the mean of this distribution, which is the best estimate of the objective function's form. In addition, the model provides the variance of the distribution, which indicates how certain the model is around its predicted mean. The model will be fitted so that it is compatible with the measurements used to construct it. In the case where measurements are noisy, this information can also be processed by the posterior distribution, so that the posterior distribution only needs to fit the data to a precision dictated by the measurement noise. In this way, the model incorporates both measurement noise and correlations between measurements at different input parameters.

The correlation between any two measurements is characterized by the covariance kernel, which can be designed to reflect domain knowledge and constraints. For instance, in reversed-phase chromatography, gradient programs typically contain isocratic steps or are increasing, so that $\varphi_i \leq \varphi_{i+1}$ for $t_i \leq t_{i+1}$. In addition, values of φ_i should be bounded between a volume fraction of 0 and 1, and t_i should be between 0 and a user-defined maximum measurement time. While the latter constraints are easily set by box constraints (lower and upper bounds on the input parameters), constraining the acquisition function such that $\varphi_i \leq \varphi_{i+1}$ and $t_i \leq t_{i+1}$, is less straightforward. We resolved this by adding an additional sorting operation to the kernel, which sorts all elements $\varphi_{1:D}$ and elements $t_{1:D}$ from low to high. This is described in more detail in Section S-1 of the Supplementary Information. The sorting operation introduces a permutation invariance to the kernel and hence the model, for example $\mathbf{x_1} = [\varphi_1 = 0.4, \varphi_2 = 0.2, \varphi_3 = 0.6, t_1 = 1, t_2 = 3, t_3 = 5]$ and $\mathbf{x_2} = [\varphi_1 = 0.4, \varphi_2 = 0.6, \varphi_3 = 0.2, t_1 = 1, t_2 = 3, t_3 = 5]$, would both be mapped to $\mathbf{x} = [\varphi_1 = 0.2, \varphi_2 = 0.4, \varphi_3 = 0.6, t_1 = 1, t_2 = 3, t_3 = 5]$. This sorting operation thus intrinsically excludes negative gradients, and therefore reduces the input parameter space significantly.

The next experiment to be performed is determined by an acquisition function based on the mean μ and standard deviation σ provided by the surrogate model. A different emphasis can be placed on the importance of the mean and standard deviation in order to allow for a trade-off between exploration and exploitation. The aim of exploration is to suggest regions of the input parameter space where model uncertainty (σ) is high. Whereas exploitation mainly focuses on regions with a high predicted mean. A simple example acquisition function is the upper confidence bound, $\text{UCB}(\mathbf{x}) = \mu(\mathbf{x}) + \beta\sigma(\mathbf{x})$, where β determines the trade-off between exploration and exploitation. Technical details of the Bayesian optimization algorithm used in this work can be found in Section 3, additional theory on Gaussian processes, and the acquisition functions can be found in Sections S-1.1 and S-1.3 of the Supplementary Information. With the above considerations in mind, we designed our Bayesian optimization loop as follows:

1. A number of measurements are performed using randomly selected input parameters.
2. A Gaussian process is fit to the previously performed measurements.
3. An acquisition function is maximized and used to propose the next measurement.

4. The next measurement is performed, assessed, and added to the dataset.
5. Steps 2–4 are repeated until the convergence criteria are met.

This is illustrated qualitatively in Fig. 3 for a dummy retention model consisting of 40 analytes (see Section 3.4 for details). In this synthetic example, the goal is to find the gradient parameters (φ_1 and φ_2) with fixed time points at $t_1 = 0$ min and $t_2 = 20$ min that lead to an optimal resolution score (as in Eq. (6)). Column A of the plots in Fig. 3 resembles the true objective function values (Eq. (6)) computed on a 20x20 grid of values of φ_1, φ_2, also known as the response surface. Note that these surfaces are symmetric with respect to the bottom-left to top-right diagonal, as we have ensured that $\varphi_1 \leq \varphi_2$ by permuting them if this is violated. It can be seen here that there is a clear optimum using a gradient program with $\varphi_1 = 0.2$ and $\varphi_2 = 1$. Column B of Fig. 3 shows the predicted mean of the Gaussian process regression model, based on the observed data points shown in red. Note that, despite the fact that all the observed data points are positioned in the upper left triangle, the Gaussian process regression model models the lower right triangle in a similar fashion, due to the imposed symmetry properties of the kernel. Column C shows the predicted variance of the surrogate model. Note that given the limited number of measurements, the model is not exactly describing the true response surface (Column A), however, even with just 5 data points (Row 1), the model already captures some trends of the true response surface. As more data points are added, the surrogate model begins to more closely resemble the true response function. Column D shows the surface of the acquisition function, where the red point denotes its maximum which is the experiment to be performed next. Note that despite the model not being able to accurately describe the true response function yet, the acquisition function proposes to perform an experiment at the global optimum after just 6 measurements (Fig. 3D2). Given enough measurements, the model should be able to

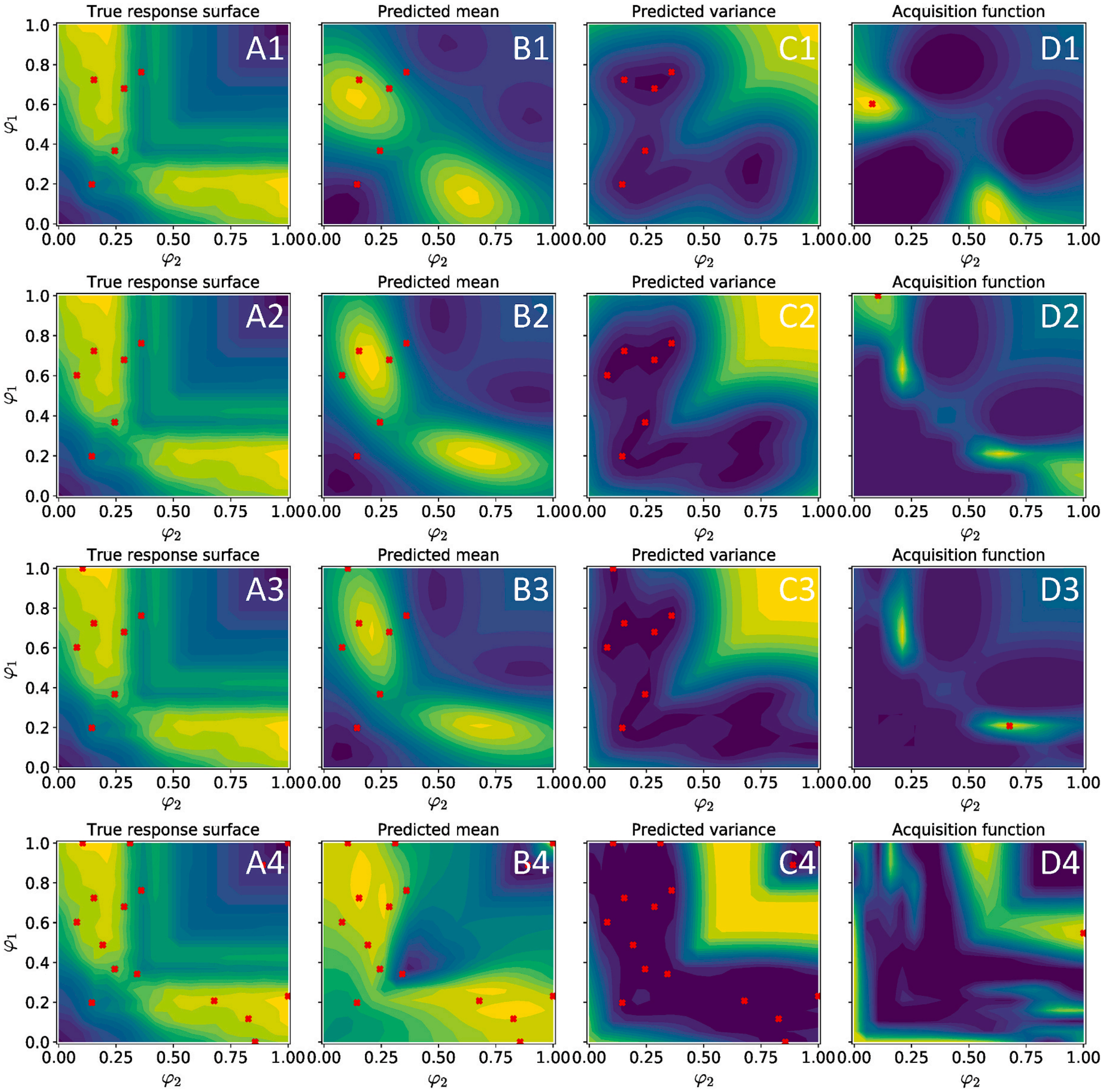


Fig. 3. Two-dimensional example of a Bayesian optimization algorithm with $n = 5$, 6, 7, and 16 measured data points (Row 1, 2, 3 and 4, respectively). For the colormap, yellow indicates a high valued observable, whereas blue indicates a lower-valued observable. Column A shows the true response surface of a toy retention model. Column B shows the predicted mean of the surrogate model, based on the measured data points shown in red. Column C shows the predicted variance of the surrogate model. Column D shows the acquisition function and its maximum, denoted by the red cross, which indicates the experiment to be performed next. As more data points are added, the surrogate model begins to more closely resemble the true response function. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

model the response function accurately.

4.2. Construction of a closed-loop platform to facilitate unsupervised automated measurements

A closed-loop platform that facilitates automation was designed in order to develop and evaluate the Bayesian optimization framework in an unsupervised manner. Since analytical instruments typically comprise sophisticated hardware with built-in safety mechanisms that prevent improper use of experimental machinery, it was opted to interface our algorithm with existing instrument control software (ICS). However, ICS is often the product of several stages of development and continuously adapted to new technology. To prevent the algorithm from requiring adaptation to specific ICS and its procedures, the algorithm was designed to program and activate the ICS, after which a listener function was activated, while the BO algorithm would remain dormant. Once the LC experiment was finished, the ICS was programmed to export measurement data and create a signal for the listener function to reactivate the BO algorithm. As a result, the LC is used as a subordinate in this workflow, controlled by the algorithm, with method parameters and a start signal as input, and raw data and a finalization signal as output. Consequently, our algorithm can support any LC system which is capable to meet these criteria.

Upon reactivation of the BO algorithm, the raw data exported by the ICS was further processed so that a consistent evaluation of the objective function was possible. This comprised a baseline correction using the asymmetric least squares method [29], with an asymmetry factor of 0.002 and a smoothing factor of 10^5, which was found to give stable results. We note that this number might have to be adjusted depending on the problem at hand (sample, detector, etc.). This was then followed by peak detection and computation of peak widths, which was done using modified routines from SciPy [30] using a height threshold of 1000 and a width threshold of 100. We note that we used the peak intensity of the detected peaks as peak height (h_i, See Eqs. (3) and (4)). The peak information was then used to compute the objective function (Eq. (6)). This workflow is illustrated in Fig. 4.

4.3. Single-objective Bayesian optimization

To study the performance of the single-objective Bayesian optimization approach, the algorithm was configured to maximize the objective function described in Eq. (6) for two complex dye mixtures, abbreviated as sample A and B (described in Section 3.3). In both cases, the algorithm was restricted to the development of four-step gradients, which had a predefined $\varphi_{init} = 0\%$ at $t_{init} = 0.25$ min, then progressed to φ_1 at t_1, to φ_2 at t_2, to φ_3 at t_3, to end at a pre-set $\varphi_{final} = 100\%$ at $t_{final} = 20.25$ min. Here we chose 20.25 min, as longer measurement times restricted the number of possible iterations due to mobile phase use, however, this value can be set shorter or longer depending on the practitioners preferences. The algorithm was thus optimizing the parameters $\mathbf{x} = [\varphi_1, \varphi_2, \varphi_3, t_1, t_2, t_3]$. The values of φ_i were restricted between 0% and 100% and the values of t_i were restricted between 0.25 min and 19 min. The algorithm was initialized with 6 initial measurements. To draw parameters for these initial measurements, we opted for a Sobol sampling approach, which is a quasi-random sampling strategy that offers a lower discrepancy (fill the space of possibilities more evenly) than conventional pseudo-random number generators. The algorithm was given an optimization budget of 34 measurements. All raw measurements and baseline corrected measurements can be found in the Supplementary Materials.

Fig. 5 shows the results of the optimization process for sample A. The algorithm is seen to explore various combinations of modifier concentrations and respective time points (Fig. 5B–C), with alternating success according to the objective function score (Fig. 5A1). This is to be expected as the algorithm is balancing between exploration and exploitation and learning more about the response surface on the fly. The best score was obtained at experiment 27, which is shown in Fig. 5D. The number of detected peaks are shown in Fig. 5A2, which is seen to have strong correlation with the objective function described in Eq. (6). However, as the objective function uses the resolution between peak pairs, it is also able to discriminate between measurements with a similar number of detected peaks, which can be seen for measurements 27 and 32, which have a similar number of detected peaks, but the former has a higher objective function score due to more resolved peaks. In addition, incorporating the lower limit of a resolution of 1 in Eq. (6), which strictly cuts off these resolutions to 0, makes the objective function evaluation more stable, as in our setup the width detection typically becomes unreliable when peaks are not well resolved.

Fig. 6 shows the results of the optimization process for sample B. Although this sample is more complex as it contains many more compounds, the algorithm is seen to optimize the separation, where the objective function (Fig. 5A1) obtains its highest value at measurement 18, which is shown in Fig. 6D. It is observed that using these gradient parameters, almost all of the separation space is used. It is also interesting to note that between experiments 10 and 28, most of the time parameters (Fig. 6B–C) are kept fixed or only adapted slightly. Whereas the φ_i parameters do change. It could be that (for this sample) there is some redundancy in the parameters, i.e. using fixed timepoints, the algorithm has enough degrees of freedom by changing the φ_i parameters to optimize the separation.

4.4. Multi-objective Bayesian optimization of complex dye mixture

In many real-world problems, there are several possible conflicting objectives that need to be optimized simultaneously in an efficient manner. In liquid chromatography, this could be the trade-off between

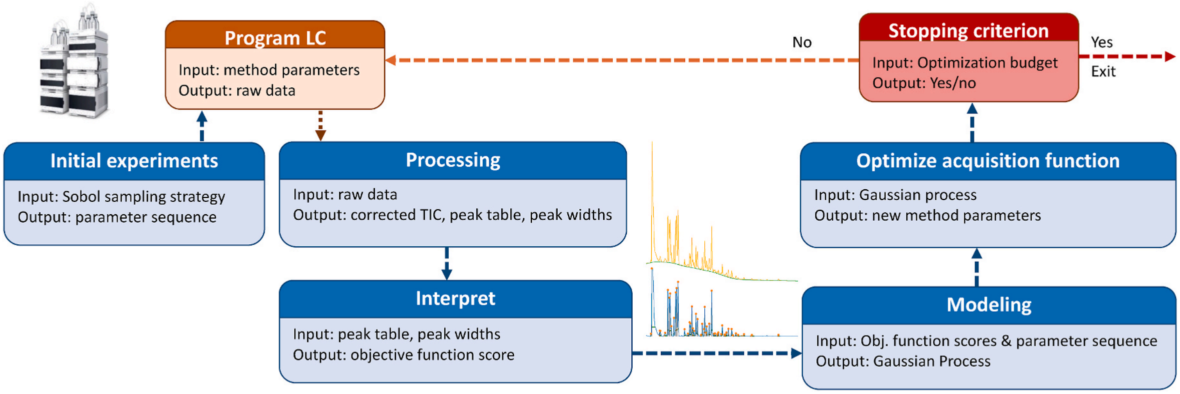


Fig. 4. Schematic overview of the closed-loop Bayesian optimization framework.

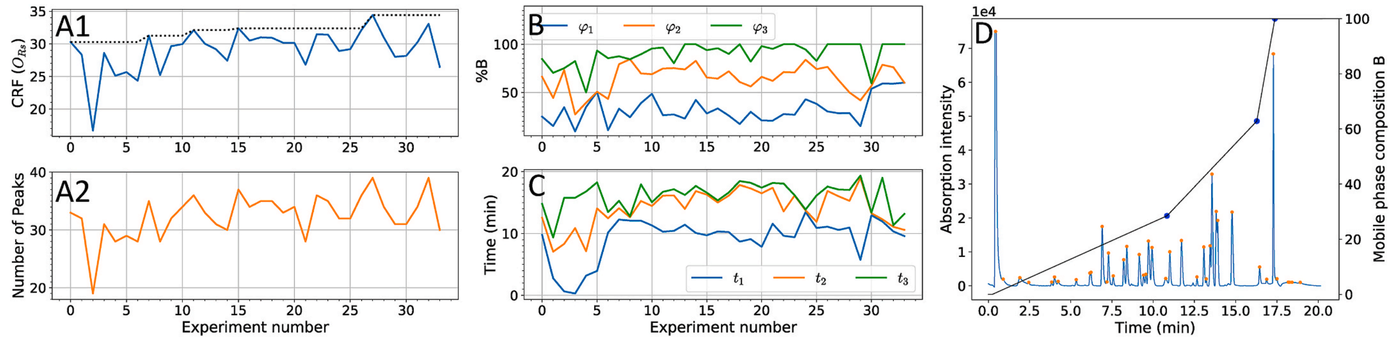
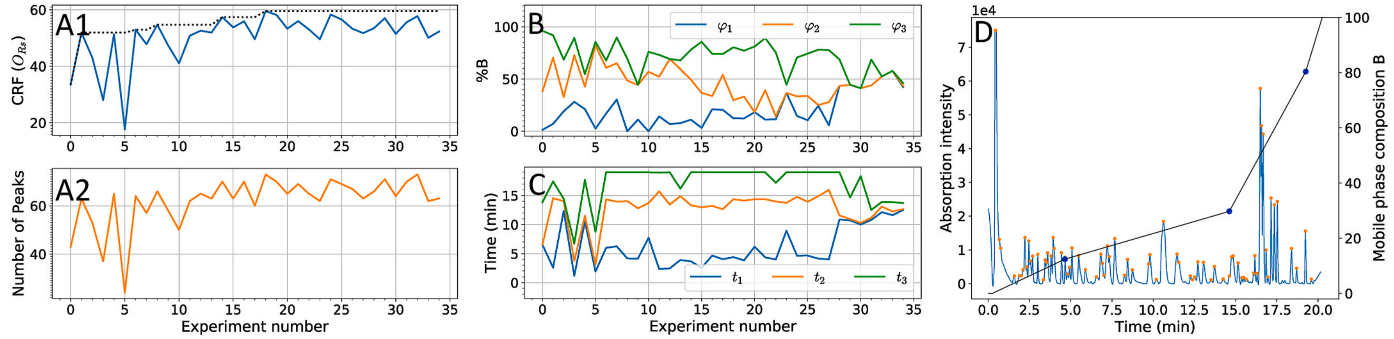


Fig. 5. Overview of results of the Bayesian optimization of sample A. A1) Observed objective function score as a function of experiment number in blue. The black dotted line denotes the best experiment value as a function of the experiment number. A2) The detected number of peaks in orange. B-C) Used values for φ_1, φ_2 and φ_3 (B) and t_1, t_2 and t_3 (C) as a function of experiment number. D) Total intensity chromatogram (in blue) of experiment 27, the measurement with the highest objective function score. Detected peaks are shown in orange. The black line denotes the gradient program. The blue dots denote the optimizable gradient parameters. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)



gr9_3c.pdf - Fig. 6. Overview of results of the Bayesian optimization of sample B. A1) Observed objective function score as a function of experiment number in blue. The black dotted line denotes the best experiment value as a function of the experiment number. A2) The detected number of peaks in orange. B-C) Used values for φ_1, φ_2 and φ_3 (B) and t_1, t_2 and t_3 (C) as a function of experiment number. D) Total intensity chromatogram (in blue) of experiment 18, the measurement with the highest objective function score. Detected peaks are shown in orange. The black line denotes the gradient program. The blue dots denote the optimizable gradient parameters. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

resolution and total measurement time. In this scenario, the practitioner is typically interested in the Pareto front. The hypervolume (i.e.,the area for two objectives, the volume for three objectives, etc.) is a frequently used metric used to assess the quality of a Pareto front [17]. The hypervolume indicator quantifies the hypervolume from a reference point *r* with respect to the Pareto front, as illustrated in Fig. 7. Hence, the reference point dictates what values of each respective objective contribute to the hypervolume. Therefore, the location of the reference point impacts what area of the Pareto front is explored most. In case there is some clear area of interest, the reference point can be kept fixed and be provided by the practitioner based on domain knowledge. However, the reference point can also be adopted on the fly if this knowledge is not there. For instance by the method proposed by Ishibuchi et al. [31] where the reference point is taken to be slightly worse (10%) than the current minimum values of the Pareto front. We use this approach to pick the reference point.

We also investigated the use of multi-objective optimization. For this, several objectives had to be defined, which were then modeled by separate Gaussian process models. Next, the multi-objective expected hypervolume improvement (EHVI) [32] acquisition function was optimized, in order to pick the experiment to perform next. This acquisition function is described in more detail in Section S-1.3. This framework was then used to maximize the objective function described in Eq. (6) in combination with an objective function based on time, which was defined to be the time of the latest eluting peak in a measurement. In

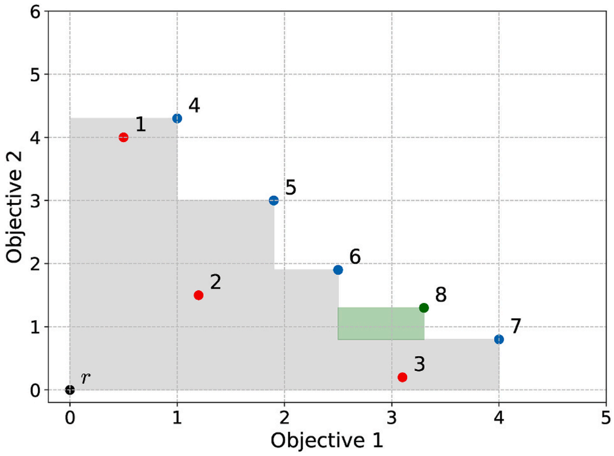


Fig. 7. Visualization of the hypervolume indicator (grey shaded area) from a reference point *r*. The blue points denote measurements on the Pareto front, while the red points denote measurements that are not on the Pareto front. The green point is a datapoint that would increase the hypervolume if measured. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

order to have a robust evaluation of this time, we performed an additional peak detection with a height threshold of 2000 and a width threshold of 100, as otherwise artifacts from the baseline correction were in rare cases wrongly detected as peaks, and hence resulted into a erroneous time score. The peak detection with a height threshold of 1000 was used for the computation of the objective function described in Eq. (6). The algorithm was restricted to optimize the parameters of a four-step gradient ($\mathbf{x} = [\varphi_1, \varphi_2, \varphi_3, t_1, t_2, t_3]$) with similar predefined values for φ_{init}, t_{init}, φ_{final} and t_{final}, and bounds for the optimizable parameters as in Section 4.3. We optimized the same dye mixtures, sample A and B as in the single-objective setting. In addition, the algorithm was initialized with the same 6 initial experiments and was given an optimization budget of 34 measurements. The reference point described in Fig. 7 was chosen to be adapted on the fly and was defined to be 10% worse than the minimum values of the Pareto front. The values of the reference point during each iteration of the algorithm can be found in Supplementary Information Section S-2. All raw measurements and baseline corrected measurements can be found in the Supplementary Materials.

Fig. 8 shows the results of the optimization process of sample A. Fig. 8B–C shows what combinations of modifier concentrations and respective time points have been proposed by the algorithm. Interestingly, compared to the single-objective optimization, the variations in the gradient parameters are larger and remain relatively large throughout the 34 experiments. This is a result of the algorithm trying to identify and explore the full Pareto front, instead of just one Pareto optimal point as in single-objective optimization. The progress of the discovery of the Pareto front can be seen in terms of hypervolume in Fig. 8A and in terms of the Pareto front in Fig. 8D. During the 34 iterations, the algorithm continuously improves the hypervolume and thus improves its estimation of the Pareto front, which seemingly is convexly shaped, thereby finding different sets of method parameters that trade off the measurement time and resolution score.

Fig. 9 shows two measurements (0 in A, and 6 in B) on the Pareto front. Experiment 6 has the highest resolution score but is the longest experiment on the Pareto front. Whereas measurement 0 is 5 min shorter, yet at a cost of a slightly reduced resolution.

In similar fashion, Fig. 10 shows the results of the optimization process of sample B. Where, although the sample has a more complex nature, the algorithm is also able to find several points on a convexly shaped Pareto front, and continues to improve the hypervolume over the 34 experiments, both finding measurements with high resolution scores and measurements with short measurement time. Compared to the Pareto front in sample A, the resolution score deteriorates faster when decreasing the measurement time, due to the more complicated sample composition. Note that for both samples, the experiments that have the highest resolutions scores in the multi-objective setting, have similar scores as the optima found by the single-objective experiments. Measurements 18 and 16 that lie on the Pareto fronts are shown in Fig. 11.

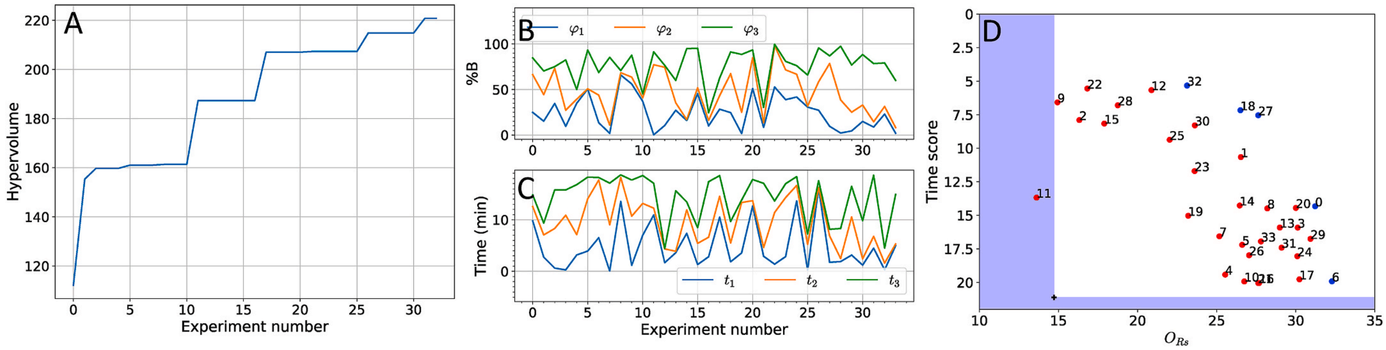


Fig. 8. Overview of results using multi-objective Bayesian optimization on sample A. A) Hypervolume determined from the initial reference point as a function of experiment number. B-C) Used values for φ_1, φ_2, and φ_3 (B) and t_1,t_2, and t_3 (C) as a function of experiment number. D) Pareto front plot. Blue points denote measurements on the Pareto front, whereas red points are dominated by the Pareto front. The black cross denotes the initial reference point from which the hypervolume is determined. The blue shaded area denotes the area that is dominated by the reference point, hence measurements that have either a time score or resolution score (or both) in this area, cannot contribute to the hypervolume. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

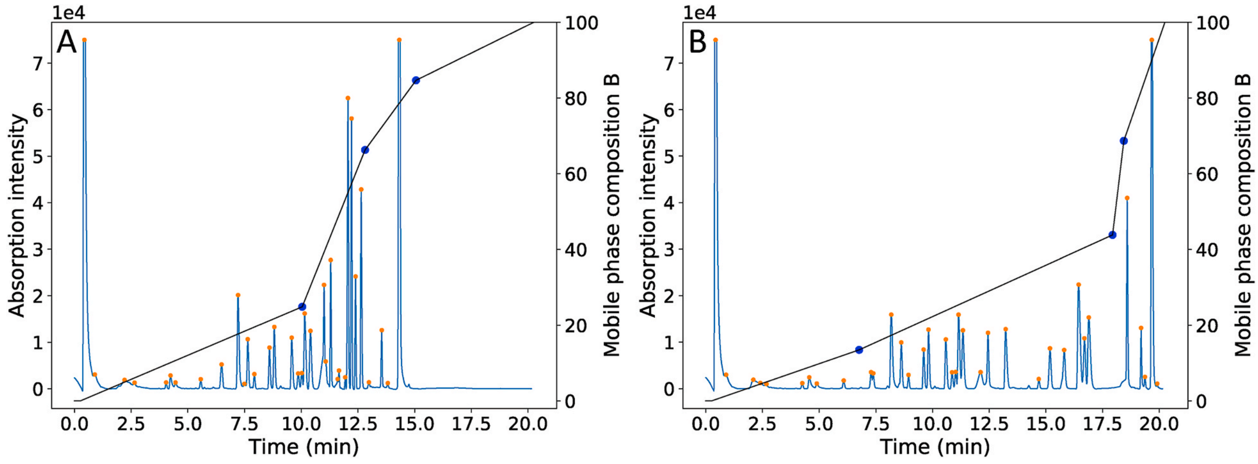


Fig. 9. Total intensity chromatograms of measurement 0 (A), and 6 (B) of sample A, which are measurements on the Pareto front.

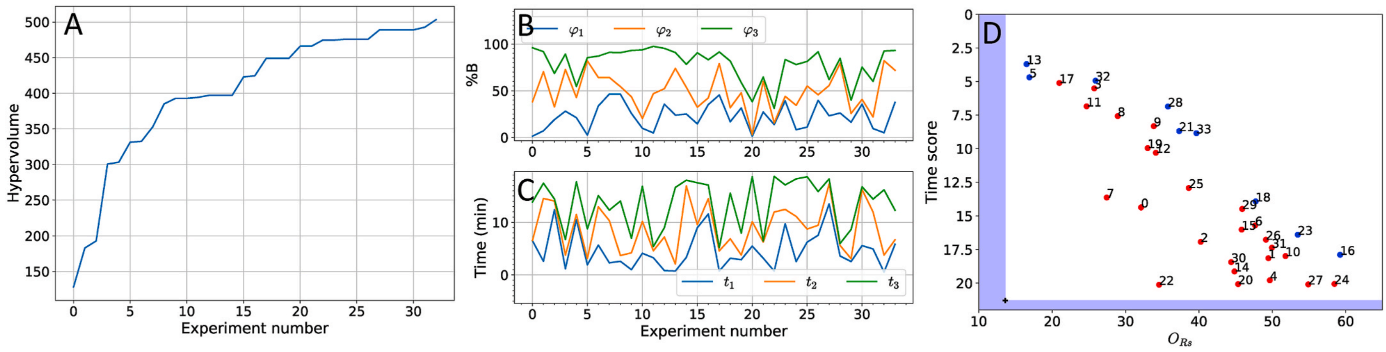


Fig. 10. Overview of results using multi-objective Bayesian optimization on sample B. A) Hypervolume determined from the initial reference point as a function of experiment number. B-C) Used values for φ_1, φ_2, and φ_3 (B) and t_1,t_2, and t_3 (C) as a function of experiment number. D) Pareto front plot. Blue points denote measurements on the Pareto front, whereas red points are dominated by the Pareto front. The black cross denotes the initial reference point from which the hypervolume is determined. The blue shaded area denotes the area that is dominated by the reference point, hence measurements that have either a time score or resolution score (or both) in this area, cannot contribute to the hypervolume. (For interpretation of the references to color in this figure legend, the reader is referred to the Web version of this article.)

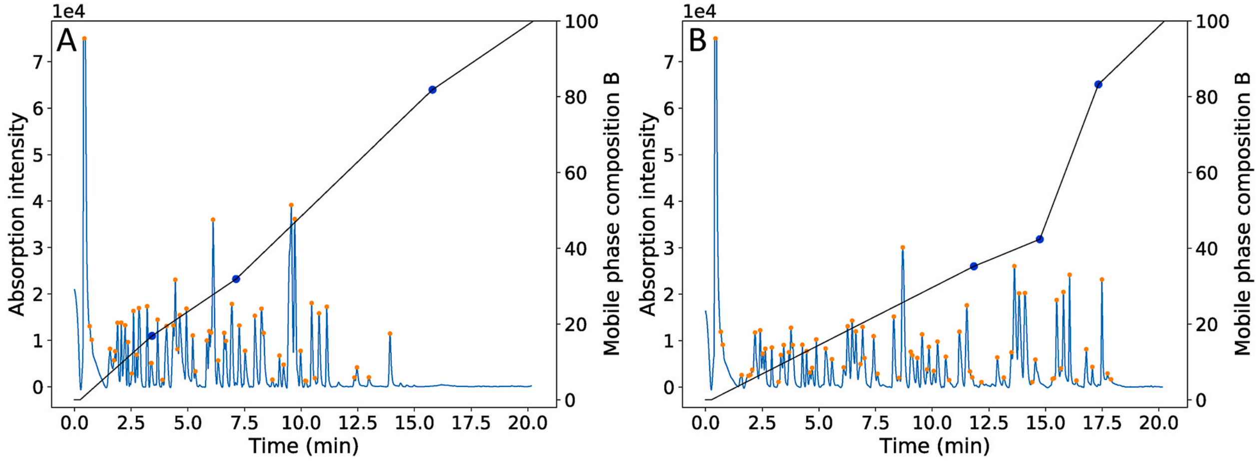


Fig. 11. Total intensity chromatograms of measurement 18 (A), and 16 (B) of sample B, which are measurements on the Pareto front.

4.5. Single-objective versus multi-objective

In this section we further investigate and compare the performance of single- and multi-objective optimization. For this purpose we study a synthetic sample (n = 20), using retention modelling (See Sec. 3.4). As in retention modelling we know the retention time and width of each component, even when not properly resolved, this allows for a comparison study of the BO algorithm that is not corrupted by measurements noise or erroneous peak detection.

When using a linear gradient, this sample has two distributions, which is shown in Fig. 12A and by quickly increasing mobile phase B after elution of the first distribution, a significantly faster method time can be obtained (See Fig. 12B–C).

To test the performance of the single-objective Bayesian optimization we defined the following scalarized objective:

$$O_{scalarized} = \alpha O_{Rs} + \beta O_t, \tag{7}$$

Where O_{Rs} is the earlier defined Eq. (6), O_t is the elution time of the last eluting peak and α and β are weighting factors which determine the importance of each objective. Note that in the determination of O_{Rs}, peaks eluting after 60 min were excluded. For the multi-objective Bayesian optimization, we maximized the hypervolume encompassed by the objectives, where we set the reference point to [0, 60], where the aim was to maximize the resolution score and minimize the measurement time. The algorithms optimized 8 gradient parameters. Both the single-objective and multi-objective algorithms where then ran for 3 trials with different random initial experiments per trial. Here the single-objective algorithm was ran using different values of α and β to compare how this affected the optimization process. The random initial experiments in each trial where the same for both the single- and multi-objective algorithms so that a fair comparison could be made.

Fig. 13A–E shows the score as a function of iterations for each algorithm setting. Here it can be observed that all single-objective (SO) variations and the multi-objective (MO) optimization converged to an optimum after 20 iterations after which only slight improvements where still observed. This was the case for all three trials as indicated by the error bars. Fig. 13F shows the Pareto fronts for each algorithm setting. Here it can be seen, in the single-objective case, that depending on the choice of weighting factors α and β, the algorithms converged to different optima. For instance, for the case $\alpha = 1$, $\beta = 1e{-}5$, which is heavily focused on maximizing the resolution (shown in green), optima in the right corner of the Pareto front were profound, thus mostly ignoring the time component. Likewise for the case $\alpha = 1e{-}3$, $\beta = 1$ (shown in red), most solutions ended up in the top left corner, hence mostly focusing on short method times. When picking a more sensible case $\alpha = 1, \beta = 0.1$ (shown in purple), the algorithm found methods with a high resolution score, but also found relatively fast methods that can do this. Therefore, it is crucial that α and β are defined in such a way that they give methods in which the practitioner is interested. This can be difficult to define a priori when the sample composition is unknown, and

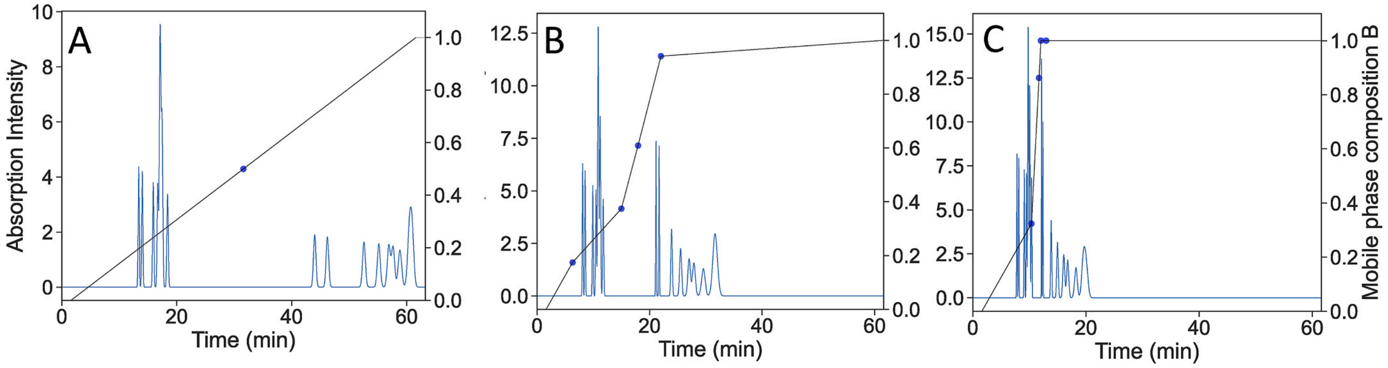


Fig. 12. Chromatograms of synthetic retention modeling sample (n = 20). A) Linear gradient, showing that sample has two distributions of eluting compounds. B) The purple measurement encircled in Fig. 13. C) The orange measurement encircled in Fig. 13.

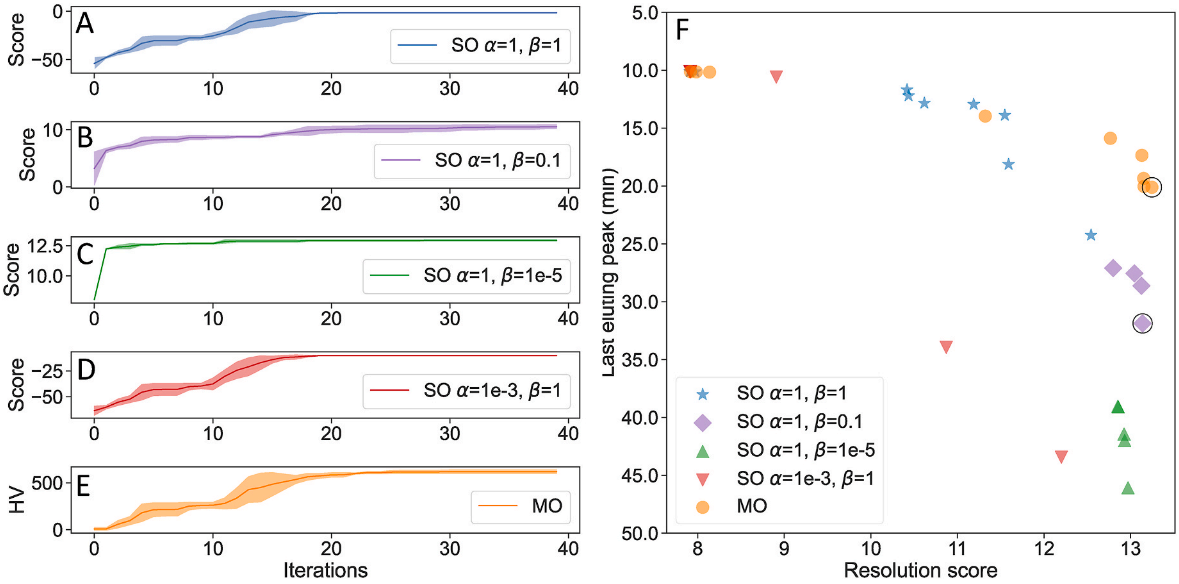


Fig. 13. A-E) Score versus iteration for multi-objective (MO) and single objective Bayesian optimization (SO) variations with different values of α and β. We report the mean and standard error over 3 trials. F) Pareto front of each respective optimization mode.

hence it is not known what values the resolution score can take, nor how long the measurement should be. The multi-objective algorithm on the other hand, found solutions in all of these areas simultaneously, within the same optimization budget, without the need of defining weighting factors. In addition, compared to the setting $\alpha = 1, \beta = 0.1$, it even found methods with similar resolution, yet shorter method times. Two chromatograms for the measurements encircled in Fig. 13F are shown in Fig. 12, where it can be seen that the multi-objective algorithm finds a faster method with similar resolution. Therefore, the use of multi-objective Bayesian optimization seems preferable over single-objective optimization, when dealing with multiple competing objectives.

5. Conclusion

We have developed an algorithm capable of optimizing gradient programs in an unsupervised closed-loop fashion. For our algorithm, we opted for Bayesian optimization given its data efficiency and adapted this for use in liquid chromatography by tailoring a Gaussian process model. To accomplish this, we designed a novel covariance kernel capable of incorporating domain knowledge. The algorithm was configured to directly interface with the LC system and was successfully applied for the optimization of a complex dye mixture. Single-objective and multi-objective Bayesian optimization were investigated. While both approaches resulted in satisfactory separation methods, multi-objective optimization provides the user with a broader range of solutions without any prior knowledge. Relative to retention modeling based approaches, our strategy based on Bayesian optimization does not rely on peak tracking nor retention modeling. As a result, this method can be used for cases where retention modeling is not possible. However, it is known that successful applications of Bayesian optimization are typically limited to problems with less than twenty parameters [33], and generally require more measurements as this dimensionality increases. Hence, the scalability of the method (i.e. the number of parameters to be simultaneously optimized) may be limited. Therefore, as an additional step forward, we envisage a solution that combines the time-efficiency of retention modelling with the data-efficiency of Bayesian optimization. Our future work will be dedicated to this. Furthermore, the applicability to different sample matrices must further be explored, as well as the utilized objective functions. Finally, the use of multi-channel data (e.g. MS) will yield additional information which is expected to significantly improve the robustness of the algorithm and will be investigated.

CRediT authorship contribution statement

Jim Boelrijk: Conceptualization, Investigation, Data curation, Formal analysis, Methodology, Writing – original draft. **Bernd Ensing:** Supervision, Writing – review & editing, Funding acquisition. **Patrick Forré:** Methodology, Supervision, Writing – review & editing. **Bob W.J. Pirok:** Methodology, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data is added to attached files. Code is stored on a Github page referenced in paper. Raw data is available upon request.

Acknowledgements

Special thanks to ChangYong Oh for helpful discussions and insights regarding Bayesian optimization.This work was performed in the context of the Chemometrics and Advanced Separations Team (CAST) within the Centre Analytical Sciences Amsterdam (CASA). The valuable contributions of the CAST members are gratefully acknowledged. The authors would additionally like to thank Pascal Camoiras Gonzalez for his experimental help. Bob Pirok acknowledges the TTW VENI research programme (Project 19173, “Unleashing the Potential of Separation Technology to Achieve Innovation in Research and Society (UPSTAIRS)“), which is financed by the Dutch Research Council (NWO).

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.aca.2023.340789.

Supplementary Materials

Supplementary material associated with this article can be found, in the online version, at DOI:

Additional information regarding the theory of Bayesian optimization and the variations of the reference point can be found in the Supplementary information. Additional data including all the samples, raw measurements and baseline corrected measurements, Example interface code, and code used in Section 4.5 can be found at: https://github.com/Jimbo994/AutoLC-BO.

References

[1] J.T.V. Matos, R.M.B.O. Duarte, A.C. Duarte, Chromatographic response functions in 1D and 2D chromatography as tools for assessing chemical complexity, Trends Anal. Chem. 45 (2013) 14–23, https://doi.org/10.1016/j.trac.2012.12.013.

[2] B.W.J. Pirok, S. Pous-Torres, C. Ortiz-Bolsico, G. Vivó-Truyols, P.J. Schoenmakers, Program for the interpretive optimization of two-dimensional resolution, J. Chromatogr. A 1450 (2016) 29–37, https://doi.org/10.1016/j.chroma.2016.04.061, 10.1016/j.chroma.2016.04.061.

[3] M.J. den Uijl, P.J. Schoenmakers, G.K. Schulte, D.R. Stoll, M.R. van Bommel, B.W. J. Pirok, Measuring and using scanning-gradient data for use in method optimization for liquid chromatography, J. Chromatogr. A 1636 (2021), 461780, https://doi.org/10.1016/j.chroma.2020.461780.

[4] P.J. Schoenmakers, H.A.H. Billiet, R. Tussen, L. De Galan, Gradient selection in reversed-phase liquid chromatography, J. Chromatogr. A 149 (C) (1978) 519–537, https://doi.org/10.1016/S0021-9673(00)81008-0.

[5] T.S. Bos, J. Boelrijk, S.R. Molenaar, B.v. t. Veer, L.E. Niezen, D.V. Herwerden, S. Samanipour, P. Forré, B. Ensing, G.W. Somsen, B.W. Pirok, State-of-the-Art chemometric strategis for fully automated interpretive method development in liquid chromatography, Anal. Chem. (2022) submitted for publication.

[6] B. Huygens, K. Efthymiadis, A. Nowé, G. Desmet, Application of evolutionary algorithms to optimise one- and two-dimensional gradient chromatographic separations, J. Chromatogr. A 1628 (2020), 461435, https://doi.org/10.1016/j.chroma.2020.461435. URL, https://linkinghub.elsevier.com/retrieve/pii/S0021967320307111.

[7] W. Hao, B. Li, Y. Deng, Q. Chen, L. Liu, Q. Shen, Computer aided optimization of multilinear gradient elution in liquid chromatography, J. Chromatogr. A 1635 (2021), 461754, https://doi.org/10.1016/j.chroma.2020.461754. URL, https://linkinghub.elsevier.com/retrieve/pii/S0021967320310281.

[8] J. Boelrijk, B. Pirok, B. Ensing, P. Forré, Bayesian optimization of comprehensive two-dimensional liquid chromatography separations, J. Chromatogr. A 1659 (2021), 462628, https://doi.org/10.1016/J.CHROMA.2021.462628.

[9] J.C. Berridge, Simplex optimization of high-performance liquid chromatographic separations, J. Chromatogr. A 485 (C) (1989) 3–14, https://doi.org/10.1016/S0021-9673(01)89129-9. URL, https://linkinghub.elsevier.com/retrieve/pii/S0021967301891299.

[10] M.W. Watson, P.W. Carr, Simplex algorithm for the optimization of gradient elution high-performance liquid chromatography, Anal. Chem. 51 (11) (1979) 1835–1842, https://doi.org/10.1021/ac50047a052. URL, https://pubs.acs.org/sharingguidelines.

[11] S. O’Hagan, W.B. Dunn, M. Brown, J.D. Knowles, D.B. Kell, Closed-loop, multiobjective optimization of analytical instrumentation: gas chromatography/time-of-flight mass spectrometry of the metabolomes of human serum and of yeast fermentations, Anal. Chem. 77 (1) (2005) 290–303, https://doi.org/10.1021/ac049146x.

[12] J. Bradbury, G. Genta-Jouve, J.W. Allwood, W.B. Dunn, R. Goodacre, J.D. Knowles, S. He, M.R. Viant, MUSCLE: automated multi-objective evolutionary optimization of targeted LC-MS/MS analysis, Bioinformatics 31 (6) (2014) 975–977, https://doi.org/10.1093/BIOINFORMATICS/BTU740. URL, https://europepmc.org/articles/PMC4380028https://europepmc.org/article/PMC/4380028.

[13] E. Tyteca, G. Desmet, A universal comparison study of chromatographic response functions, J. Chromatogr. A 1361 (2014) 178–190, https://doi.org/10.1016/j.chroma.2014.08.014.

[14] P. Nikitas, A. Pappa-Louisi, P. Agrafiotou, Multilinear gradient elution optimisation in reversed-phase liquid chromatography using genetic algorithms, J. Chromatogr. A 1120 (1–2) (2006) 299–307, https://doi.org/10.1016/J.CHROMA.2006.01.005.

[15] J.T.V. Matos, R.M.B.O. Duarte, A.C. Duarte, A generalization of a chromatographic response function for application in non-target one- and two-dimensional chromatography of complex samples, J. Chromatogr. A 1263 (2012) 141–150, https://doi.org/10.1016/j.chroma.2012.09.037.

[16] J.C. Berridge, Unattended optimisation of reversed-phase high-performance liquid chromatographic separations using the modified simplex algorithm, J. Chromatogr. A 244 (1) (1982) 1–14, https://doi.org/10.1016/S0021-9673(00)80117-X.

[17] S. Daulton, M. Balandat, E. Bakshy, Differentiable Expected Hypervolume Improvement for Parallel Multi-Objective Bayesian Optimization, Advances in Neural Information Processing Systems. URL http://arxiv.org/abs/2006.05078.

[18] E. Zitzler, L. Thiele, M. Laumanns, C.M. Fonseca, V.G. Da Fonseca, Performance assessment of multiobjective optimizers: an analysis and review, IEEE Trans. Evol. Comput. 7 (2) (2003) 117–132, https://doi.org/10.1109/TEVC.2003.810758.

[19] I. Couckuyt, D. Deschrijver, T. Dhaene, Fast calculation of multiobjective probability of improvement and expected improvement criteria for Pareto optimization, J. Global Optim. 60 (3) (2014) 575–594, https://doi.org/10.1007/S10898-013-0118-2.

[20] B. Shahriari, K. Swersky, Z. Wang, R.P. Adams, N. De Freitas, Taking the human out of the loop: a review of Bayesian optimization, Proc. IEEE 104 (1) (2016) 148–175, https://doi.org/10.1109/JPROC.2015.2494218.

[21] S. Daulton, D. Eriksson, M. Balandat, E. Bakshy, Multi-objective Bayesian optimization over high-dimensional search spaces, in: The 38th Conference of Uncertainty in Artificial Intelligence, 2022, pp. 1–11. URL, http://arxiv.org/abs/2109.10964.

[22] P.J. Schoenmakers, J.K. Strasters, A. Bartha, Correction of the resolution function for non-ideal peaks, J. Chromatogr. A 458 (C) (1988) 355–370, https://doi.org/10.1016/S0021-9673(00)90578-8.

[23] B. W. J. Pirok, M. J. d. Uijl, G. Moro, S. V. J. Berbers, C. J. M. Croes, M. R. v. Bommel, P. J. Schoenmakers, Characterization of Dye Extracts from Historical Cultural-Heritage Objects Using State-of-the-Art Comprehensive Two-Dimensional Liquid Chromatography and Mass Spectrometry with Active Modulation and Optimized Shifting Gradients, Analytical Chemistry doi:10.1021/ACS.ANALCHEM.8B05469. URL https://pubs.acs.org/doi/abs/10.1021/acs.analchem.8b05469.

[24] W. Hao, K. Wang, B. Yue, Q. Chen, Y. Huang, J. Yu, D. Li, Peak compression in linear gradient elution liquid chromatography, J. Chromatogr. A 1619 (2020), 460908, https://doi.org/10.1016/j.chroma.2020.460908.

[25] M. Balandat, B. Karrer, D. R. Jiang, S. Daulton, B. Letham, A. G. Wilson, E. Bakshy, BoTorch: a framework for efficient monte-carlo Bayesian optimization, Adv. Neural Inf. Process. Syst. 33.

[26] J.R. Gardner, G. Pleiss, D. Bindel, K.Q. Weinberger, A.G. Wilson, Gpytorch: Blackbox matrix-matrix Gaussian process inference with GPU acceleration, in: Advances in Neural Information Processing Systems, vol. 2018, Decem, 2018, pp. 7576–7586. URL, https://gpytorch.ai.

[27] D.R. Jones, A taxonomy of global optimization methods based on response surfaces, J. Global Optim. 21 (4) (2001) 345–383, https://doi.org/10.1023/A:1012771025575. URL, https://link.springer.com/article/10.1023/A:1012771025575.

[28] C.E. Rasmussen, Gaussian Processes in Machine Learning, vol. 3176, Springer Verlag, 2004, https://doi.org/10.1007/978-3-540-28650-9_4. URL, http://www.tuebingen.mpg.de//simcarl, http://link.springer.com/10.1007/978-3-540-28650-9_4.

[29] P.H.C. Eilers, H.F.M. Boelens, Baseline correction with asymmetric least squares smoothing, Leiden Univ. Med. Centre Rep. 1 (1) (2005) 5.

[30] P. Virtanen, R. Gommers, T.E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S.J. van der Walt, M. Brett, J. Wilson, K.J. Millman, N. Mayorov, A.R.J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, I. Polat, Y. Feng, E.W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E.A. Quintero, C.R. Harris, A.M. Archibald, A.H. Ribeiro, F. Pedregosa, P. van Mulbregt, A. Vijaykumar, A.P. Bardelli, A. Rothberg, A. Hilboll, A. Kloeckner, A. Scopatz, A. Lee, A. Rokem, C.N. Woods, C. Fulton, C. Masson, C. Häggström, C. Fitzgerald, D.A. Nicholson, D.R. Hagen, D. V. Pasechnik, E. Olivetti, E. Martin, E. Wieser, F. Silva, F. Lenders, F. Wilhelm, G. Young, G.A. Price, G.L. Ingold, G.E. Allen, G.R. Lee, H. Audren, I. Probst, J. P. Dietrich, J. Silterra, J.T. Webber, J. Slavič, J. Nothman, J. Buchner, J. Kulick, J. L. Schönberger, J.V. de Miranda Cardoso, J. Reimer, J. Harrington, J.L. C. Rodríguez, J. Nunez-Iglesias, J. Kuczynski, K. Tritz, M. Thoma, M. Newville, M. Kümmerer, M. Bolingbroke, M. Tartre, M. Pak, N.J. Smith, N. Nowaczyk, N. Shebanov, O. Pavlyk, P.A. Brodtkorb, P. Lee, R.T. McGibbon, R. Feldbauer, S. Lewis, S. Tygier, S. Sievert, S. Vigna, S. Peterson, S. More, T. Pudlik, T. Oshima, T.J. Pingel, T.P. Robitaille, T. Spura, T.R. Jones, T. Cera, T. Leslie, T. Zito, T. Krauss, U. Upadhyay, Y.O. Halchenko, Y. Vázquez-Baeza, SciPy 1.0: fundamental algorithms for scientific computing in Python, Nat. Methods 17 (3) (2020) 261–272, https://doi.org/10.1038/S41592-019-0686-2.

[31] H. Ishibuchi, R. Imada, Y. Setoguchi, Y. Nojima, How to specify a reference point in hypervolume calculation for fair performance comparison, Evol. Comput. 26 (3) (2018) 411–440, https://doi.org/10.1162/EVCO_a_00226.

[32] K. Yang, M. Emmerich, A. Deutz, T. Bäck, Multi-Objective Bayesian Global Optimization using expected hypervolume improvement gradient, Swarm Evol. Comput. 44 (October 2018) (2019) 945–956, https://doi.org/10.1016/j.swevo.2018.10.007.

[33] D. Eriksson, M. Jankowiak, High-dimensional Bayesian optimization with sparse axis-aligned subspaces, in: 37th Conference on Uncertainty in Artificial Intelligence, UAI, vol. 2021, 2021, pp. 493–503.