







REVIEW OPEN ACCESS

A Guide to Bayesian Optimization in Bioprocess Engineering

Maximilian Siska^{1,2}  | Emma Pajak³  | Katrin Rosenthal^{4,5}  | Antonio del Rio Chanona³  | Eric von Lieres^{1,2}  | Laura M. Helleckes^{3,6} 

¹IBG-1: Biotechnology, Forschungszentrum Jülich, Jülich, Germany | ²Computational Systems Biotechnology, RWTH Aachen University, Aachen, Germany | ³The Sargent Centre for Process Systems Engineering, Department of Chemical Engineering, Imperial College London, London, UK | ⁴School of Science, Constructor University, Bremen, Germany | ⁵Bioprocess Engineering, Faculty of Technology, Bielefeld University, Bielefeld, Germany | ⁶I-X Centre for AI in Science, Imperial College London, London, UK

Correspondence: Laura M. Helleckes (l.helleckes@imperial.ac.uk)

Received: 25 August 2025 | **Revised:** 4 December 2025 | **Accepted:** 8 December 2025

Keywords: Bayesian optimization | bioprocess engineering | Design of Experiments | guidelines

ABSTRACT

Bayesian optimization has become widely popular across various experimental sciences due to its favorable attributes: it can handle noisy data, perform well with relatively small data sets, and provide adaptive suggestions for sequential experimentation. While still in its infancy, Bayesian optimization has recently gained traction in bioprocess engineering. However, experimentation with biological systems is highly complex and the resulting experimental uncertainty requires specific extensions to classical Bayesian optimization. Moreover, current literature often targets readers with a strong statistical background, limiting its accessibility for practitioners. In light of these developments, this review has two aims: first, to provide an intuitive and practical introduction to Bayesian optimization; and second, to outline promising application areas and open algorithmic challenges, thereby highlighting opportunities for future research in machine learning.

1 | Bayesian Optimization—From Decision Theory to Practical Experimental Design in Bioprocess Engineering

Across various experimental sciences, Bayesian optimization (BO) is increasingly recognized as a valuable tool for tackling complex, costly, and time-consuming optimization problems—conditions that are also common in bioprocess engineering. While BO saw a major increase in applications over the past decade, particularly in the context of **machine learning** (Shahriari et al. 2016) (see [Glossary](#)), its roots lie deep in the history of decision theory and experimental design, beginning in the 20th century.

Major components of the modern BO framework were already introduced in the context of sequential experimental design as early as the 1960s. In two seminal papers (Kushner 1962, 1964),

applied mathematician Harold Kushner introduced the optimization of a noisy **objective function** using probabilistic models, specifically what we now call **Gaussian processes (GPs)**, to represent the **uncertainty** about how different inputs affect an experimental outcome. He also proposed heuristic strategies for selecting the next experiment, based on either the probability of improvement or how uncertain the outcome in a promising region is. These concepts laid the foundation for the modern BO approaches we apply today. Notably, the field of geostatistics also promoted the use of GPs for experimental design as early as the 1950s, then promoted under the name of kriging after early work by Krige (1951) and Matheron (1963).

In the following decades, several scientists such as Šaltenis (1971) and Močkus (1975, 1989) formalized and further

Abbreviations: BO, Bayesian optimization; DoE, Design of Experiments; EI, expected improvement; GP, Gaussian process; LLM, large language model; MF-BO, multi-fidelity Bayesian optimization; UCB, upper confidence bound.

This is an open access article under the terms of the [Creative Commons Attribution](#) License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2026 The Author(s). *Biotechnology and Bioengineering* published by Wiley Periodicals LLC.

promoted the concept of BO, for example, introducing *expected improvement* (Section 2.3) as a concept to design the next experiment, which is still frequently applied today. By the 1990s, BO had made the full transition from theory to practice (Jones et al. 1998). However, limited computational power still hindered its widespread adoption. This changed drastically following the landmark study by Snoek et al. (2012), who demonstrated that BO could outperform manual tuning and other automated methods for **hyperparameter** optimization in machine learning. This success catalyzed widespread adoption in machine learning and sparked broader interest across scientific disciplines. During this period, BO was also rediscovered in the context of sequential experimental design (Shahriari et al. 2016; Frazier and Wang 2016; Marchant and Ramos 2012; Negoescu et al. 2011). Notably, materials science and chemical engineering have emerged as important application domains for BO in experimental optimization. Researchers in these domains have leveraged BO to guide discovery of materials (Deshwal et al. 2021; Diwale et al. 2022; Jin and Kumar 2023), tune reaction conditions (Shields et al. 2021; Taylor et al. 2023), and optimize reactor designs (Park et al. 2018; Savage et al. 2023)—challenges that mirror those encountered in biotechnology.

In recent years, BO has gained increasing attention in biotechnology, with applications ranging from drug discovery to protein design and reaction engineering. Large data sets, such as those in clinical drug development, benefit from Bayesian approaches that not only yield statistical insights but also support planning, analysis, and decision-making (Ruberg et al. 2023). Furthermore, BO has efficiently guided iterative robotic experiments in protein engineering (Hu et al. 2023), the selection of production strains (Helleckes, Küsters, et al. 2024), or the optimization of production media (Freier et al. 2016) and reaction conditions (Pandi et al. 2022). However, BO is by no means limited to large data sets and automated experiments; it also excels in small-data settings. In bioprocess engineering, the number of experiments that can be carried out might be limited due to costs, complexity, or the fact that automated laboratory equipment may not be available. In such cases, BO can perform effectively with well under 100 experiments, as demonstrated in the optimization of growth and production media (Yoshida et al. 2023; Narayanan et al. 2025), reaction engineering (Siedentop et al. 2025; Rosa et al. 2022), process engineering (Hernández Rodríguez et al. 2022), and downstream processing (Jäpel and Buyel 2022; Freier and von Lieres 2017; Narayanan et al. 2021).

Thus, BO is becoming a promising tool for bioprocess engineering, offering data-efficient and robust optimization. However, despite several reported case studies, which were previously summarized in Gisperg et al. (2025), BO remains underexplored in this field. On the one hand, the field presents unique challenges, such as complex measurement systems with varying biases and high uncertainty resulting from the use of living organisms or parts thereof. The models and algorithms needed to address these challenges are often not yet included in standard BO software or documentation, creating an additional entry barrier. On the other hand, much of the existing literature is targeted at machine learning researchers and scientists from other domains.

This review aims to comprehensively address these gaps, providing guidance on how to tackle challenging experimental systems in bioprocess engineering. Gisperg et al. (2025) provide

a concise overview of current BO applications in bioprocess engineering; here, we build on that foundation by offering a more detailed and practitioner-oriented guide, including expanded methodological background, practical recommendations for real experimental settings, and a structured outlook on future challenges and opportunities.

Our goals are twofold. First, we offer the bioprocess engineering community an accessible methodological foundation and practical guidelines for addressing complex optimization problems. Readers already familiar with BO may wish to proceed directly to Section 3, where we present these practical guidelines. Second, we highlight the specific characteristics of bioprocess engineering applications that present challenging and underexplored problems to the machine learning community, which are further highlighted in Sections 4–6.

2 | Methodological Background of the Bayesian Optimization Framework

In the field of bioprocess engineering and its related experimental work, we face many challenges that can be tackled as optimization problems, for example, finding the composition of a growth medium that yields the highest growth rate, or choosing between different biocatalysts for a bioprocess to obtain the highest reaction rate. From a mathematical point of view, this is generalized in the field of optimization, where we seek to find the optimum of an objective function (i.e., our optimization target), which we evaluate at certain points by making one or more **observations** (i.e., conducting **experiments**). In an iterative process, we decide where to perform these experiments by applying an **optimization policy** to the experimental data observed so far. This policy can take many forms, ranging from naively picking random points within the **parameter space** to strategies that identify the next experimental conditions with the greatest potential to improve upon current results.

BO is a probabilistic, iterative method for identifying the optimum of a black-box objective function, meaning that we do not know the functional form of the optimization target a priori. In bioprocess engineering, the objective function could be, for example, the titer of a product or the overall cost of a bioprocess. As shown in Figure 1, BO requires four elements:

1. wet-lab (or in silico) experiments (i.e., the data-generating system for the observations);
2. a **surrogate model** describing the black-box objective function;
3. an **acquisition function** (i.e., the mathematical function that assesses the value of potential new experiments according to the optimization policy); and
4. a termination criterion.

In the following, each of these components will be explained in a dedicated subsection.

2.1 | Experiments

The first step in BO is to conduct initial experiments, yielding a set of observations that can be used to fit the first surrogate

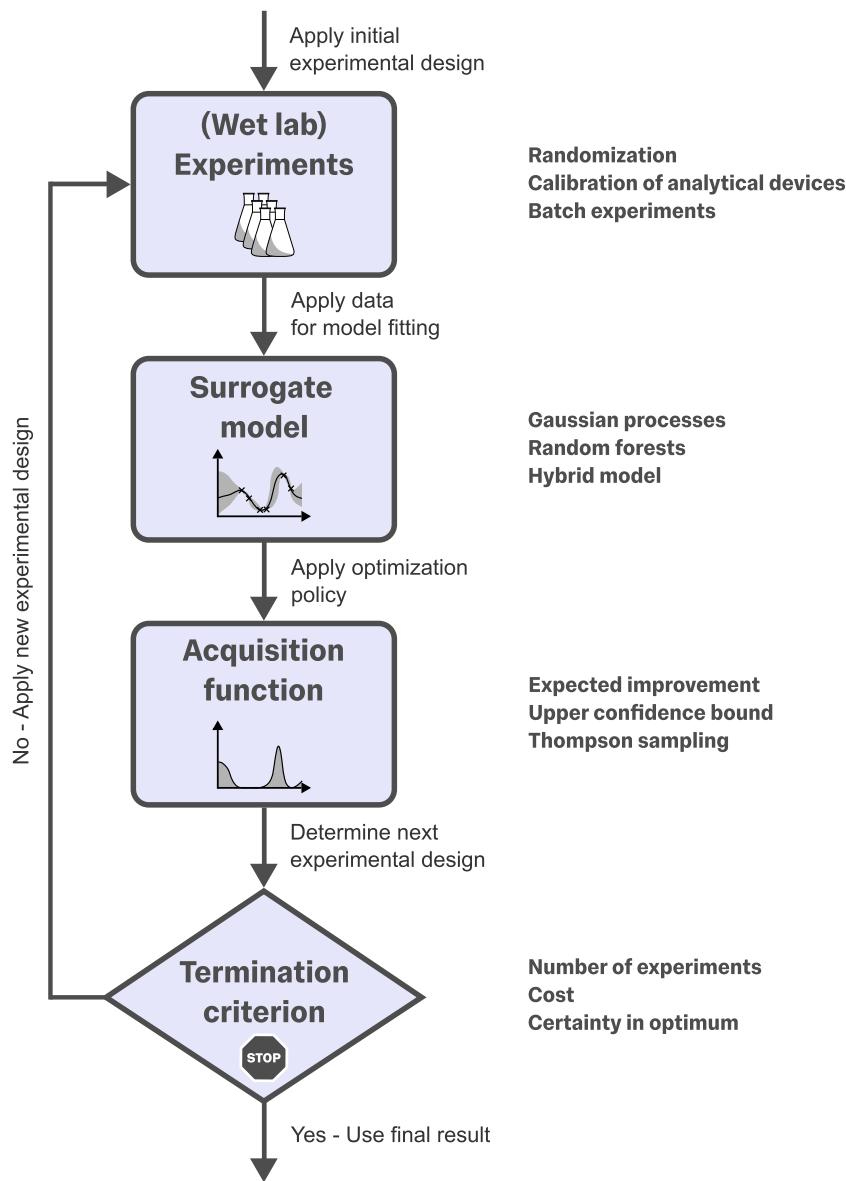


FIGURE 1 | Overview of the four steps in the BO workflow. BO is a sequential, iterative process. Before the iterations begin, initial experiments are performed to obtain foundational observations. This step is typically conducted using space-filling designs such as Sobol sequences or Latin hypercube sampling. For the experiments themselves, important considerations include randomization, calibration of analytical devices to avoid biases, and distributing experiments across multiple batches (Section 2.1). Next, the generated data are used to fit a surrogate model, most commonly a GP (Section 2.2.1), although alternative methods can also be employed, such as decision trees or hybrid models that combine mechanistic and data-driven approaches (Box 2). Based on the surrogate model, the most informative next experimental conditions are selected using an acquisition function, depending on the chosen optimization policy. Common choices include expected improvement, upper confidence bound, or Thompson sampling (Section 2.3). This procedure is then iteratively repeated by applying the selected experimental designs in the next round of experimentation. The iterative procedure is terminated based on a stopping criterion, such as the certainty with which the optimum has been identified, a predefined maximum number of experiments, or overall resource constraints.

model. While the acquisition function (Section 2.3) determines the design of subsequent experiments at later stages of the loop, the initial experiments, sometimes also referred to as excitation design, depend on the specific case study. For example, prior knowledge of the optimization problem might influence the choice of experiments, that is, how to vary the **design variables** initially. In general, space-filling designs have proven beneficial in many cases (Jones et al. 1998). These experimental designs aim to strategically cover the entire parameter space spanned by the design variables, using Sobol sequences or Latin hypercube

sampling. Other typical excitation designs follow established strategies from Design of Experiments (DoE), such as factorial designs.

As with classical DoE, experimental considerations such as randomization of samples, distribution of experiments across **batches**, and minimization of nuisance factors play an important role in obtaining high-quality data sets. This is especially important in bioprocess engineering, where positional biases (e.g., in microtiter plates) and batch-to-batch variability can significantly affect results. These effects are therefore discussed

in more detail in Section 4. Beyond the experimental designs themselves, characteristics of the experimental system, for example, the measurement noise of different analytical devices, can also be taken into account in BO. Many traditional DoE methods assume homoscedastic noise, that is, constant variance across the full range of design variables. However, BO can also accommodate more complex scenarios with **heteroscedastic noise** in the observations, that is, variance that changes with the design variables. To identify the relationship between the measured quantities and the output of the analytical device, calibration can be performed using known reference standards to characterize the response of the devices. Further details on how this information can be used in BO will be discussed in Section 4.

2.2 | Surrogate Model

With the initial set of experiments complete, the next step in the BO workflow (Figure 1) is to fit the surrogate model. A surrogate model is a mathematical approximation that captures how the objective function (e.g., reaction yield or selectivity) changes in response to different design variables (e.g., temperature, pH). In other words, it provides a mapping from the experimental inputs to the system's output. Surrogate models are beneficial when the true objective function is expensive to evaluate, and a mechanistic model is unavailable—both common scenarios in complex bioprocess engineering systems.

For example, fitting a simple mathematical relationship such as a polynomial or response surface to describe how fermentation yield varies with temperature is a straightforward application of surrogate modeling. Importantly, surrogates are deliberately less complex than the physical system they approximate. Rather than aiming for high-fidelity reproduction across the entire parameter space, they should provide sufficiently informative approximations to support informed decisions.

A desirable property of a surrogate model is the ability to act as a universal approximator, meaning it can capture complex input–output relationships even when the true system behavior is unknown. This capability is crucial in bioprocess engineering, where the underlying biological behavior is often unknown or only partially described. Returning to the fermentation example, the true relationship between yield and temperature may be highly nonlinear and poorly understood. Nonetheless, a well-chosen surrogate model with universal approximation capability can still capture the underlying trends and interactions effectively.

While universal approximation is a desirable property, the ability to estimate predictive uncertainty is essential for BO, and not all surrogate models offer this capability. Common surrogate models that do provide uncertainty estimates include GPs, probabilistic decision trees, and Bayesian neural networks. Among these, GPs are particularly popular in BO. They provide uncertainty estimates alongside predictions and operate as nonparametric models that flexibly adapt to data without assuming fixed structural forms. Moreover, they perform well even with limited experimental data compared to other universal approximators such as neural networks. These characteristics make GPs well-suited for applications in bioprocess engineering. In the following, we introduce GPs in more detail.

2.2.1 | Introduction to Gaussian Processes

Suppose we want to model an unknown function $f(\mathbf{x})$, where \mathbf{x} is the vector of design variables. This function could be an enzyme's catalytic activity, which depends on temperature and pH. A GP defines a probability distribution over possible functions that could describe such a dependency, assigning higher likelihood to functions that align with both prior knowledge of the system and observed data. Conceptually, a GP extends the multivariate normal distribution to infinitely many variables, one for each possible input point in the domain, i.e., combinations of our design variables. We provide a primer on the statistical background for GPs, such as multivariate normal distributions, conditioning, and marginalization (see Box 1). Additionally, for further explanation on the link between multivariate normal distributions and GPs, we recommend the following tutorials (Wang 2023; Görtler et al. 2019).

Any finite set of function values drawn from a GP follows a multivariate normal distribution. Thanks to this property, GPs allow closed-form inference: we can compute predictions and uncertainties exactly using linear algebra. This makes them computationally tractable, as the required calculations are efficient and do not involve approximations or iterative fitting.

A GP is fully defined by two components: the mean function, $m(\mathbf{x})$, which gives the expected function value at input \mathbf{x} before observing data, and the covariance function, $k(\mathbf{x}, \mathbf{x}')$, which quantifies the similarity of outputs between inputs \mathbf{x} and \mathbf{x}' . In practice, the covariance function determines the entries of the covariance matrix, Σ (see Box 1).

$$f(\mathbf{x}) \sim \mathcal{GP}(m(\mathbf{x}), k(\mathbf{x}, \mathbf{x}')).$$

Note that the term **kernel** is commonly used in place of covariance function in the literature; hereafter, we adopt the term kernel throughout this work.

Initially, before observing any data, we define a **prior distribution** based on our assumptions (see Box 1). A crucial part of this prior is the choice of kernel, which encodes assumptions about properties such as smoothness, periodicity, or trends in the underlying function. One of the most popular kernels is the squared exponential (or radial basis function—RBF) kernel, defined as:

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2\ell^2}\right) + \sigma_n^2 \delta(\mathbf{x}, \mathbf{x}'),$$

where the hyperparameters are the length scale, ℓ , signal variance, σ_f^2 , and noise variance, σ_n^2 . The length scale, ℓ , controls the rate at which correlations between function values decay with distance in the parameter space, affecting how much each observation informs predictions at nearby locations. Intuitively, it determines how quickly the function is allowed to vary across the parameter space: small values lead to rapid changes over short distances, while large values enforce smoother, more slowly varying functions.

This behavior is illustrated in Figure 2 (top inset), which shows the contrast between short and long length scales. The signal variance, σ_f^2 , sets the overall amplitude of the function,

BOX 1 | Mathematical Primer for Gaussian Processes.

Normal (Gaussian) Distributions . To understand GPs intuitively, it is helpful to begin with their fundamental building block: the normal distribution. A real-valued random variable X follows a normal (Gaussian) distribution if:

$$X \sim \mathcal{N}(\mu, \sigma^2), p(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(x - \mu)^2}{2\sigma^2}\right]$$

where the mean, μ , and the variance $\sigma^2 > 0$ fully characterize the distribution. Here, $p(x)$ is the probability density function of the normal distribution.

In practice, we are often interested in modeling more than one random variable. This leads us to the multivariate normal distribution, or “joint” normal distribution—a generalization of the univariate normal to higher-dimensional spaces. A vector of random variables, $\mathbf{X} = (X_1, \dots, X_k)^T$, follows a multivariate normal distribution if every subset of variables, or any weighted combination of them, is also normally distributed.

We write this multivariate normal distribution as:

$$\mathbf{X} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}),$$

here, $\boldsymbol{\mu}$ is the mean vector, and $\boldsymbol{\Sigma}$ is a symmetric, positive-definite covariance matrix. Each entry Σ_{ij} quantifies how much \mathbf{x}^i and \mathbf{x}^j tend to increase or decrease together, that is, how they co-vary. Two properties make this distribution especially powerful:

- *Marginalization*—Any subset of variables from a multivariate normal distribution is itself normally distributed.
- *Conditioning*—Observing some variables allows us to update our beliefs about the others, based on the known correlation between them, yielding a conditional distribution that is also multivariate normal.

Bayes' Theorem . GPs are Bayesian models, meaning they rely not only on observed data but also incorporate prior beliefs through the kernel and mean function, which together define a prior distribution over possible objective functions in BO. Bayes' theorem provides the governing framework for updating the prior beliefs in light of new data that becomes available through experiments. Let θ denote the model parameters, and D the observed data. Bayes' theorem is given by:

$$p(\theta|D) = \frac{p(\theta) \cdot p(D|\theta)}{p(D)}$$

Here, $p(\theta)$ is the prior distribution, that is, the probability distribution we define for the parameters without having observed any data. $p(D|\theta)$ is the likelihood, which expresses how likely the observed data are under a given choice of parameters. The denominator $p(D)$, known as the evidence, serves as a normalization factor. For continuous variables, it is given by the integral $\int p(\theta) \cdot p(D|\theta) d\theta$. The result,

$p(\theta|D)$, is the posterior distribution, reflecting our updated beliefs after observing the data.

In GPs, these principles are applied when inferring the hyperparameters of the mean and kernel functions, such as the length scale and noise variance (see Section 2.2.1). These hyperparameters can be treated as random variables and updated using observed data, resulting in a posterior distribution over plausible hyperparameter values. While full Bayesian inference is supported in probabilistic programming frameworks such as PyMC (Abril-Pla et al. 2023), most widely used GP and BO libraries (e.g., Balandat et al. 2020; Gardner et al. 2018; Head et al. 2021) rely on point estimates via marginal likelihood maximization instead. The computational challenges of this approximation are discussed in Section 3.2.3. The choice of prior functions for GPs will be further discussed in Sections 3 and 5.

indicating how strongly it is expected to vary around its mean. The noise variance, σ_n^2 , captures uncertainty in the observations themselves, allowing the model to account for measurement noise. When $\sigma_n^2 = 0$, the model assumes noise-free data, whereas increasing this value allows for greater deviation between the modeled function and the observed data—this effect is visualized in Figure 2 (bottom inset).

Finally, different kernels encode different assumptions about the underlying function. The squared exponential kernel implies smooth behavior, while the Matérn kernel introduces a smoothness parameter ν to model rougher functions, and periodic kernels are well-suited to capturing repeating trends. These are among the most commonly used kernels, though many others exist for specialized modeling needs. Section 5.3 explores more advanced kernels.

Once the choices for kernels and hyperparameters are made, the GP can be used for BO. As we collect data, we use it to update our prior beliefs, forming a **posterior distribution** (see Box 1). This updated GP reflects both what we have observed and our remaining uncertainty about the function elsewhere. The ability of GPs to perform such updates stems directly from the properties of the multivariate normal distributions discussed in Box 1: marginalization allows prediction at new points, while conditioning lets us update our model as we observe new data.

Thanks to the conditioning and marginalization properties of multivariate normals, GPs enable:

- *Prediction*: Estimating function values at new, unmeasured points using observed data.
- *Quantifying uncertainty*: Assigning confidence levels to predictions, often visualized by the confidence intervals.
- *Continuous refinement*: In BO, new data points are chosen for observation to update and refine our model. How do we decide which experiment to run next? *This is where acquisition functions come in!*

For readers eager to learn more about GPs, we recommend *Gaussian Processes for Machine Learning* by Rasmussen and Williams (2006).

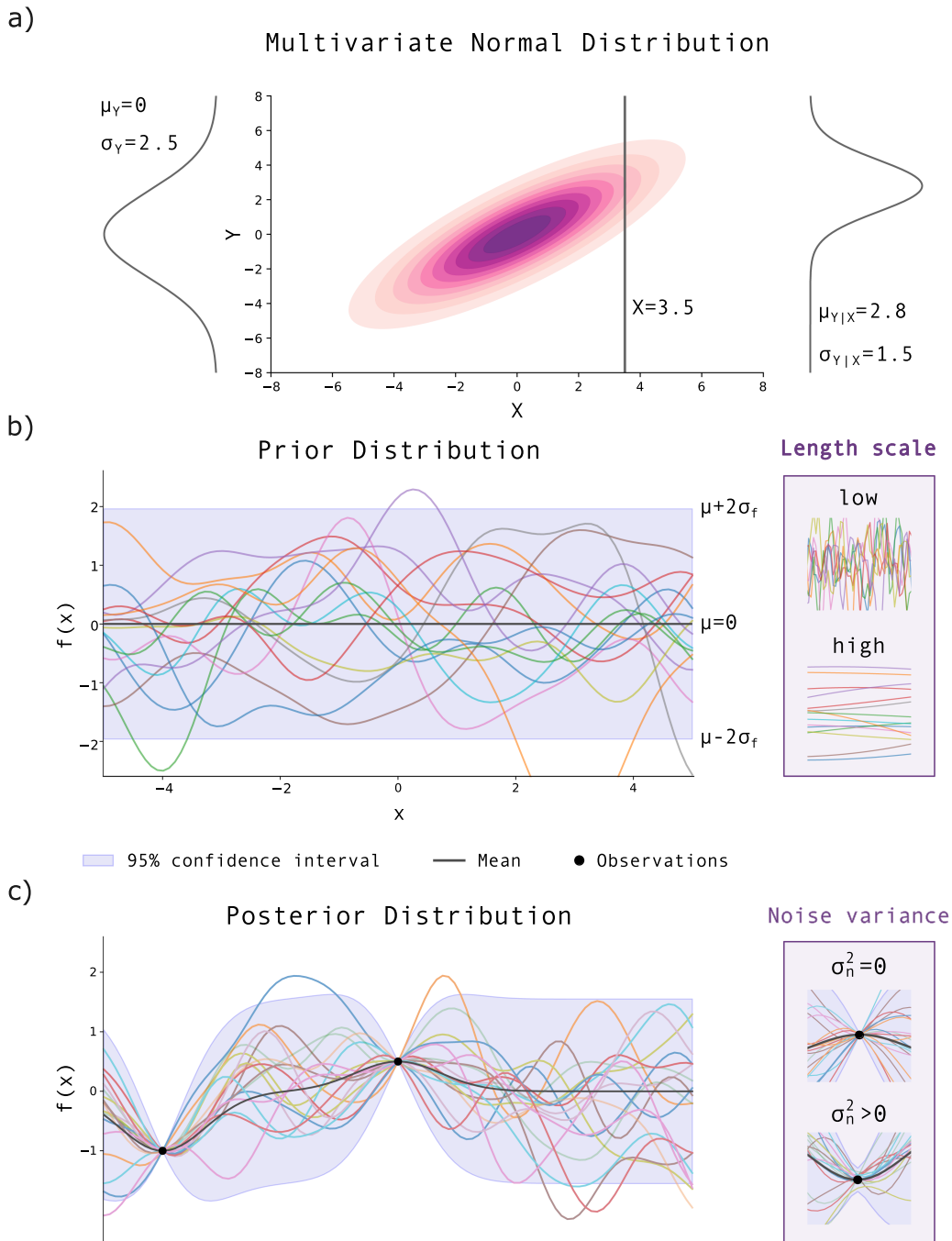


FIGURE 2 | Overview of Gaussian processes: (a) marginal, joint, and conditional distributions of two correlated Gaussian random variables ($\mu_X = \mu_Y = 0$, $\sigma_X = \sigma_Y = 2.5$, $\rho = 0.8$). Left: marginal $p(Y)$ with $\mu_Y = 0$, $\sigma_Y = 2.5$. Center: joint contours with vertical slice at $X = 3.5$. Right: conditional $p(Y|X = 3.5)$ with $\mu_{Y|X} = 2.8$, $\sigma_{Y|X} = 1.5$. The structure in (a) provides the foundation for GPs: while a multivariate normal distribution models joint behavior over a finite set of variables, a GP extends this to an infinite collection of random variables, one for each input, such that any finite subset follows a multivariate normal distribution. (b) Prior distribution: samples from a GP prior with zero mean and a squared exponential kernel. The shaded region indicates the 95% confidence interval. Inset: effect of kernel length scale l , where a lower l yields more rapid variation and a higher l produces smoother functions. (c) Posterior distribution: GP posterior after conditioning on observations (black markers). Inset: higher noise variance σ_n^2 increases predictive uncertainty and flexibility of the posterior.

2.3 | Acquisition Function

Iterative optimization typically proceeds via an optimization policy, that is, a strategy for selecting the next observation(s) based on existing data. In BO, this policy is indirectly implemented through an *acquisition function*, which assigns scores to candidate experiments and estimates their contribution to the optimization process, based on the current data,

surrogate model, and optimization policy. Most acquisition functions balance two key paradigms, *exploration* and *exploitation*. Exploration refers to making observations, i.e., conducting an experiment, where predictive uncertainty is high, aiming to discover potentially promising regions in the parameter space. In contrast, exploitation prioritizes regions with high predicted objective values, aiming to refine the

current best estimate. Effective optimization requires a careful balance of the exploration–exploitation trade-off. While many acquisition functions have been proposed, we focus on three widely used approaches: expected improvement (EI), upper confidence bound (UCB), and Thompson sampling (compare Figure 3):

- *Expected improvement (EI)* quantifies the improvement expected from evaluating a candidate solution compared to the current best-known observation (gray area in Figure 3). It balances exploitation (sampling where the predicted mean is high) and exploration (sampling where uncertainty is large), guiding the search toward promising regions. Formally, EI is defined as:

$$EI(\mathbf{x}; \xi) = (\mu(\mathbf{x}) - f(\mathbf{x}^*) - \xi) \Phi\left(\frac{\mu(\mathbf{x}) - f(\mathbf{x}^*) - \xi}{\sigma(\mathbf{x})}\right) + \sigma(\mathbf{x}) \phi\left(\frac{\mu(\mathbf{x}) - f(\mathbf{x}^*) - \xi}{\sigma(\mathbf{x})}\right)$$

here, μ and σ are the GP predictive mean and standard deviation at point \mathbf{x} , $f(\mathbf{x}^*)$ is the best observed value so far, and ξ is a parameter that encourages exploration when set to a higher value. The term $\Phi(\cdot)$ denotes the cumulative distribution function of the normal distribution and captures

the likelihood of improvement, while $\phi(\cdot)$, the corresponding probability density function, reflects the model’s uncertainty. Together, these terms reward points that are either likely to outperform the current best or are uncertain enough to merit exploration. For more details on EI, we refer to Garnett (2023) and Agnihotri and Batra (2020).

- *Upper confidence bound (UCB)* selects candidates by computing an UCB on the GP posterior mean, given by $\mu(x) + \gamma\sigma(x)$, where γ controls the exploration–exploitation trade-off (a larger γ places greater weight on uncertainty, promoting exploration). In Figure 3, $\gamma = 0.999$, indicating that the acquisition function strongly prioritizes uncertain regions. This upper bound appears as the pink curve overlaid on the GP posterior, and the pink-shaded UCB acquisition function peaks in areas where this bound is highest. The selected candidate (pink star) lies in a region of high uncertainty and moderate predicted value, exemplifying the behavior of UCB in a high- γ regime.
- *Thompson sampling* selects candidates by drawing a **random sample** of a function from the GP posterior (see Box 1) and choosing the point that maximizes it. This approach captures both prediction and uncertainty in a single realization, naturally balancing exploration and exploitation without relying on explicit hyperparameters. In Figure 3, three sample functions are shown in blue, green, and orange,

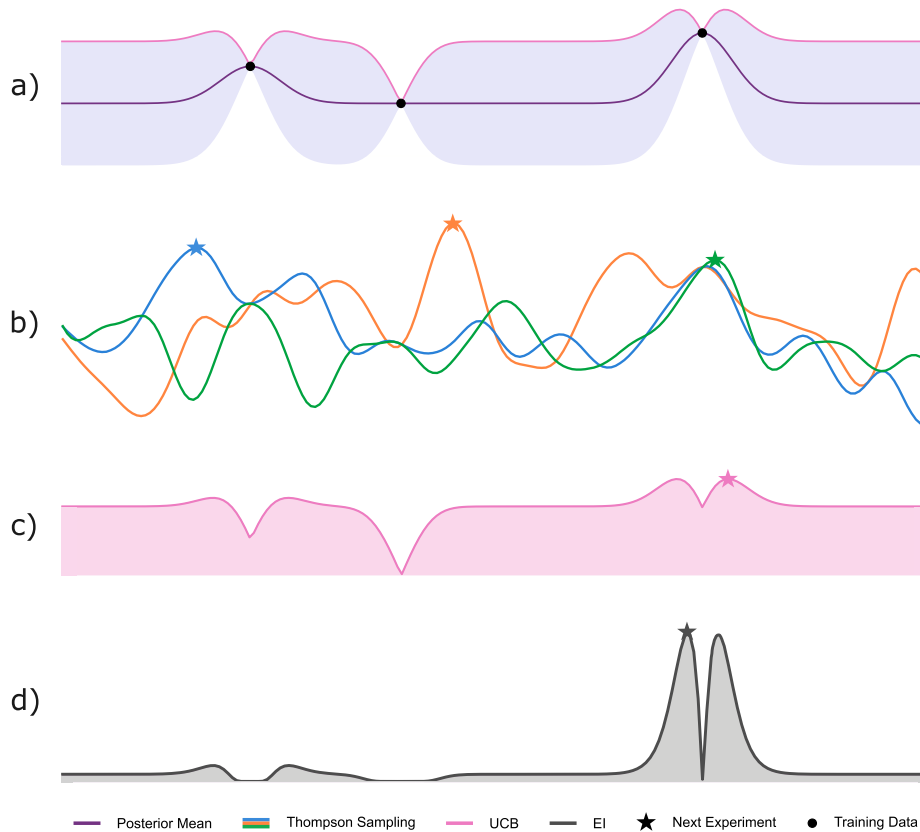


FIGURE 3 | Comparison of BO acquisition functions. (a) The top row shows the GP posterior with confidence intervals and observations. The right-most black circle represents the current highest observation, which is the baseline for EI. The solid pink line visualizes the UCB. The subsequent rows illustrate different acquisition strategies: (b) Thompson sampling (sample functions from the GP posterior, shown in blue, green, and orange), (c) UCB (pink), and (d) EI (gray). Stars mark the next sampling points proposed by each acquisition function.

and orange, along with their respective optima, demonstrating the diversity in candidate selection across random samples.

2.4 | Termination Criterion

The final decision in each optimization loop, as shown in Figure 1, is whether to terminate the process or proceed with another set of experiments. The termination criterion does not need to follow a specific mechanism and is often determined in advance by external decision makers. For example, many projects are constrained by a limited number of experiments or a maximum allowable cost. In other cases, termination may be based on achieving a specific optimization target, such as reaching a desired product titer or attaining a defined level of confidence in having identified the optimum. Alternatively, the decision to terminate may be evaluated dynamically. For instance, either an external agent (e.g., a human decision maker) or the algorithm itself may choose to stop based on the experimental results obtained so far. In bioprocess engineering, the termination criterion should ideally be informed by experimental experts, who can often preallocate an experimental budget and advise on an appropriate target for the objective function, for example, a fraction of the maximum theoretical yield.

3 | Practical Guidelines for Bayesian Optimization in Bioprocess Engineering

With the methodological foundations established in the previous section, including the BO workflow and its core elements such as surrogate models and acquisition functions, we now turn to practical implementation. Although GPs and acquisition functions are now widely supported by open-source software libraries (Balandat et al. 2020; Dürholt et al. 2024; Fitzner et al. 2025; Gardner et al. 2018; Head et al. 2021), their application to the optimization of experimental systems presents additional challenges. In bioprocess engineering, these challenges stem in part from the biological variability and measurement noise inherent to living systems. If not properly addressed, these factors can impair model accuracy and undermine the effectiveness of BO workflows.

In this section, we translate the general methodology of BO with GPs, from defining the parameter space to selecting suitable acquisition functions, into domain-specific best practices. The success of BO in bioprocess engineering hinges in particular on sound decisions made in the initial steps leading up to the surrogate model configuration, as these propagate through the BO workflow. Figure 4 presents a decision tree that highlights these key considerations and, while not fully exhaustive, provides guidance for a broad range of bioprocess applications. It offers a structured overview of the decisions discussed in this section and highlights cases in which departures from the presented guidelines may be warranted.

Following Figure 4, the best practices presented here discuss the selection of design variables and the implications of increasing dimensionality, including when dimensionality reduction may or may not be advisable (Section 3.1.1), the establishment of initial data sets suitable for BO (Section 3.1.2),

and the guided configuration of GPs (Sections 3.2.1–3.2.3). Cases that justify deviations from the presented workflow, such as high-dimensional settings, integrating mechanistic process understanding, or noncontinuous parameter spaces, are treated in the respective sections beyond Section 3. To ground the guidance for the baseline case, we introduce a representative case study from biocatalysis: a crude cell extract of unknown composition that contains multiple enzymes. A detailed characterization of the extract would be costly, but preliminary experiments indicate that it effectively converts a given substrate into a desired product without further processing. Our objective is to maximize the reaction rate by optimizing a single design variable, the pH value. No prior knowledge is assumed regarding the pH optima of the individual enzymes, making this a suitable black-box problem for illustrating BO workflows. We use this example throughout the following sections and conclude with an overview of the sequential optimization in Figure 6.

3.1 | Experiments

Following the workflow outlined in Figure 1, a successful BO study begins with careful consideration of the number of design variables, the strategy for selecting initial experiments, and how these choices affect the overall experimental budget. These early decisions shape the complexity of the parameter space and optimization problem, the structure of the initial data set, and ultimately the representativeness of the surrogate model.

3.1.1 | Defining the Parameter Space

Most bioprocess optimization studies are initialized from existing processes, which may involve laboratory-scale proof-of-concept experiments or previously established process configurations intended for improvement. In such settings, candidate design variables are typically selected based on biological hypotheses, prior experience, or practical constraints such as technical feasibility or material cost. If a reliable historical data set already exists, such as results from prior BO campaigns or comparable covering the same design variables, it should be reused to warm-start the BO workflow with the configuration of the surrogate model (see Figure 4).

While BO is well-suited to handle experimental setups involving many design variables (see Figure 4), it remains subject to the curse of dimensionality. Increasing the number of variables expands the parameter space, thereby exponentially increasing the number of experiments required for process optimization (Shahriari et al. 2016; Greenhill et al. 2020). Users must therefore carefully balance the number of considered design variables with the available experimental resources from the outset. Automation and parallelization of experiments can support workflows with more than ten design variables, but challenges arise long before the computational limits of BO with GPs are reached. These include the growing complexity of interpreting and visualizing BO behavior in high-dimensional spaces (see Section 3.2.3). Such practical considerations, rather than limitations of the GP surrogate itself, often motivate dimensionality reduction in bioprocess engineering.

In classical DoE methodology, it is common practice to adjust dimensionality by reducing the number of design variables

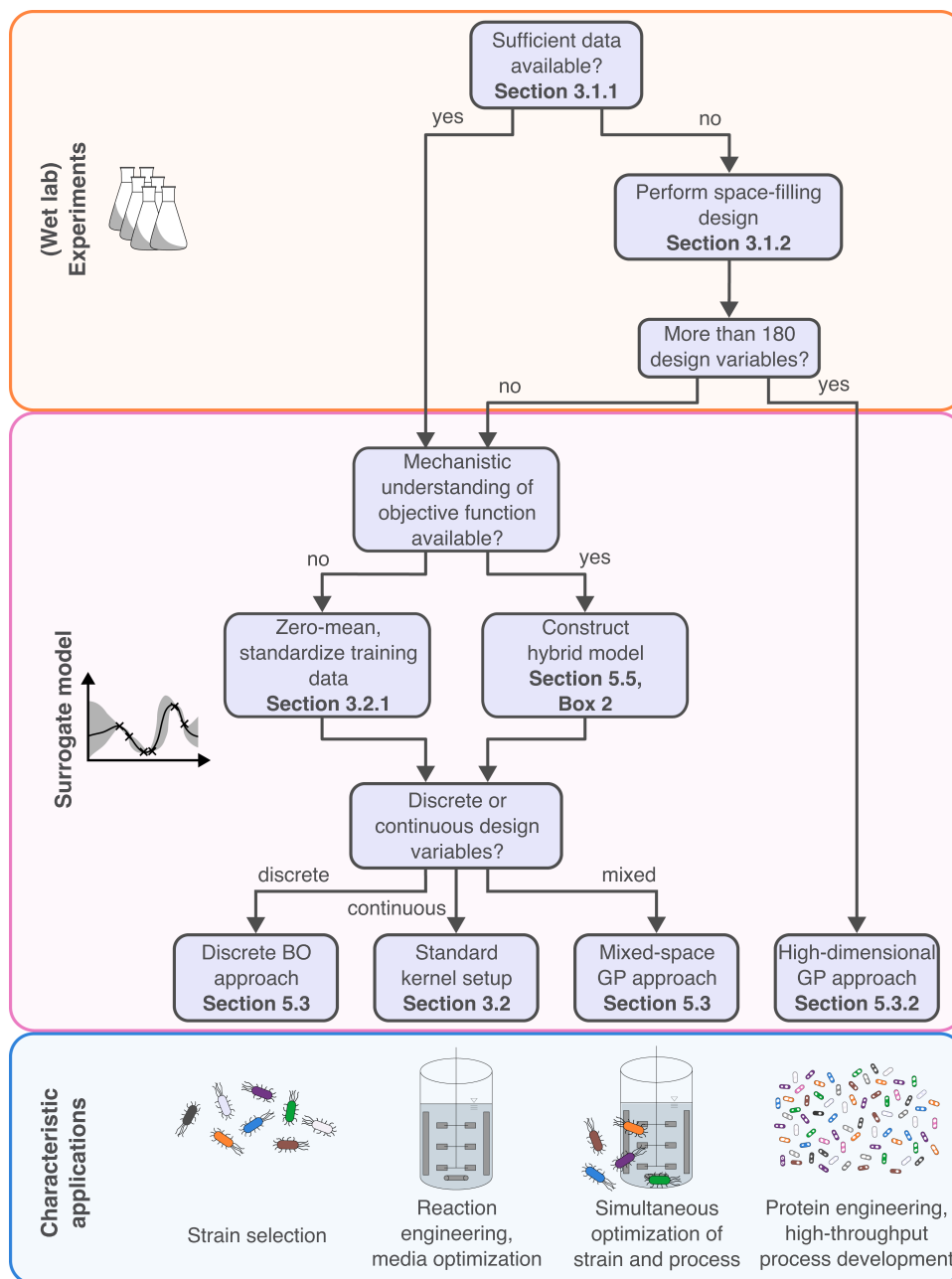


FIGURE 4 | Decision tree for the first two steps of the BO workflow in bioprocess engineering. The figure summarizes the key considerations discussed in Section 3, from parameter space definition up to surrogate model configuration, and guides readers toward an appropriate BO approach based on data availability, dimensionality of the experimental system, mechanistic prior knowledge, and the nature of the design variables. The decision threshold for high-dimensional BO approaches should be interpreted as a rule-of-thumb value, informed by the empirical findings in the biological case study of Hvarfner et al. (2024). Beyond the decisions shown in the tree, batch experimentation is a critical consideration throughout the workflow and is discussed separately in Section 5.1. The bottom row provides representative bioprocess applications, including strain selection and growth media optimization, that commonly align with the depicted BO approaches.

before optimization in high-dimensional settings. Established sensitivity analysis and screening techniques, such as Plackett–Burman, fractional factorial, or one-factor-at-a-time, can help to identify and eliminate factors with negligible influence (Cosenza et al. 2022; Narayanan et al. 2025). Translating this practice to BO by reducing dimensionality prior to model fitting can, in principle, improve model fitting and evaluation, and alleviate visualization challenges (Binois and Wycoff 2021; Greenhill et al. 2020; Shan and Wang 2010; Siedentop et al. 2025). However, these approaches are

inherently local in nature and may miss nonlinear effects or higher-order interactions. As a result, any such reduction carries the inherent risk of overlooking interdependencies between variables, which may subsequently affect process optimization (Schonlau and Welch 2006). Given this limitation, dimensionality reduction should be applied with caution. If variables are excluded prior to BO, a good practice is to reevaluate them at the predicted optimum after BO has been terminated. Naturally, this strategy cannot reveal optima located in entirely different regions of the parameter space.

Despite its prevalence in bioprocess engineering, prescreening and design variable reduction may not be necessary for multi-dimensional BO when the experimental budget is not limiting. Hvarfner et al. (2024) have recently demonstrated that traditional BO performs well in high-dimensional settings, provided one can assume that the underlying system remains bounded in complexity as the dimensionality increases. Under this assumption, dimensionality alone does not necessitate specialized approaches, even in parameter spaces with more than 100 design variables. In line with these findings, we generally recommend switching to specialized high-dimensional approaches only when the number of design variables grows beyond 180. For problems below this threshold, guidance on configuring GPs appropriately is provided in Section 3.2.3. Additional considerations for high-dimensional experimental systems are covered in Section 5.3.

Once the design variables are selected, defining appropriate bounds for each parameter becomes a key step in shaping the optimization space. These bounds are generally informed by available expert knowledge, physical and biological limitations, or economic considerations. Substrate concentrations, for example, may be constrained by solubility limits or material costs. For our case study, this would mean choosing a wide pH range, as there is no prior knowledge regarding the pH optima of the enzymes present in the crude extract. To ensure that the optimum lies within the parameter space, we define bounds ranging from pH 3 to 11.

3.1.2 | Designing Initial Experiments

The quality of the initial data set strongly influences the representativeness of the fitted surrogate model (Georgiou et al. 2025). Insufficient or poorly distributed initial coverage of the parameter space may leave large regions underexplored, which can degrade the quality of the surrogate model, cause unreliable uncertainty estimates, and ultimately misguide the acquisition function in proposing suboptimal experimental conditions (De Luca et al. 2023; Greif et al. 2025). Space-filling designs such as Sobol sequences or Latin hypercube sampling are recommended to achieve adequate coverage of the parameter space. Their importance grows with dimensionality, as the risk of under-sampling regions increases rapidly with the number of design variables.

In lower-dimensional settings, such as our one-dimensional pH optimization case study, equidistant grids supplemented by expert-guided manual experiments are typically a good choice to initialize the GP model (Savage and del Río Chanona 2024). Accordingly, we would perform five equidistant initial experiments in the specified pH range to ensure broad coverage of the parameter space. In addition, one data point near neutral pH would be included, a region that expert knowledge on enzyme activity suggests may be particularly promising.

As a general rule of thumb, the spacing between initial points should roughly match the expected length scale of the GP kernel. In practice, it is not uncommon to allocate approximately half of the total number of experiments to the initial data set (Narayanan et al. 2021; Morschett et al. 2017; Siedentop et al. 2025).

In scenarios where prior experimental data are available, for example, from preliminary screening experiments, these data points can be used for fitting the initial GP model. However,

augmenting such data sets with additional space-filling designs remains advisable to improve coverage of the parameter space, in particular if the experiments were conducted as part of a screening round aiming to reduce dimensionality (see Figure 4). Further experiments should be systematically selected to fill sparsely sampled regions or generated using approaches like the expert-augmented DoE proposed by Savage and del Río Chanona (2024).

3.1.3 | A Case Against Experimental Replicates

Replicated experiments—such as triplicates—are commonly used to obtain precise pointwise estimates within the parameter space. In the context of BO, however, such replication represents a missed opportunity to extract maximal information about the response surface from each experiment (Dai et al. 2023; Kreyling et al. 2018). The inherent uncertainty quantification provided by GP models allows single measurements to meaningfully inform both predictions and their associated uncertainties. This strategy remains robust even in the presence of occasional outliers (Siedentop et al. 2025).

Nonetheless, replicates remain valuable when used selectively to quantify systematic sources of uncertainty, such as batch effects. These considerations are explored in more detail in the discussion on hybrid modeling in Section 4 and Box 2. In line with these considerations, we elect not to perform replicated measurements in the pH case study. As shown in Figure 6, each experimental condition is evaluated only once to maximize information gain.

3.2 | Surrogate Model

Once the initial data set is established, the next critical step in the BO workflow is fitting the surrogate model. As mentioned in Section 2.2, while any probabilistic surrogate model can be used, the most common choice is a GP model. At this stage, users must make informed decisions about the model configuration to ensure that prior knowledge is appropriately incorporated, that the model yields robust predictions, and that it provides reliable uncertainty estimates to guide the acquisition function.

Prominent open-source libraries for BO and GP modeling (Balandat et al. 2020; Dürholt et al. 2024; Fitzner et al. 2025; Gardner et al. 2018; Head et al. 2021) offer reasonable default configurations for general-purpose applications. However, the success of BO in bioprocess engineering relies on adapting these configurations to the specific characteristics and constraints of the domain. We illustrate best practices for this adaptation in general, as well as using the previously introduced case study on pH optimization.

3.2.1 | Mean Function

In most bioprocess engineering scenarios, detailed prior knowledge of the underlying system behavior is limited. In such cases, we recommend using a zero-mean function and standardizing the observations prior to model fitting. This configuration provides a simple, uninformative prior that enables the GP kernel to learn the underlying functional relationship directly from the data without introducing additional

BOX 2 | Extension to Other Probabilistic Models as Surrogates for BO.

GPs remain the most frequently used surrogate models for BO due to their favorable properties described in Section 2.2.1. However, bioprocess engineering case studies often come with prior knowledge on biological parameters, for example, reaction kinetics. As discussed in Section 5, such cases may motivate the use of mechanistic models as mean functions for the GP, or a shift to another probabilistic surrogate model altogether.

In this box, we show an example of a simulated case study from biocatalysis that warrants sufficient prior knowledge to motivate the use of hybrid models (Figure 7). The case study describes the optimization of the reaction temperature of an enzyme, which follows first-order reaction kinetics. We assume that observations of product formation for a single reaction at different temperatures can be measured (Figure 7, left). Experiments are conducted batch-wise in shake flasks (four experiments with different designs per batch), between which the substrate concentration changes due to an undesired batch effect. This can be seen in the deviations between the measured data (dots) and the noise-free reaction curve (solid lines) in Figure 7 (left panel). This example demonstrates more complex experimental conditions and requires the use of BO extensions.

We compare two approaches: (a) the use of a naive GP and (b) a hybrid model, which takes a GP in combination with mechanistic equations for reaction kinetics. Using the probabilistic programming framework PyMC (Abril-Pla et al. 2023), we set up a model that describes the first-order reaction kinetics of the enzyme. For product formation, we can explicitly formulate the reaction as:

$$P_t = S_0 \cdot (1 - e^{(-k \cdot t)}) \quad (1)$$

with P_t as the product concentration at time t , S_0 as the initial substrate concentration, and k as the reaction rate constant.

As shown in the plotted data (Figure 7, left), the reaction rate constant k is temperature-dependent, that is, becoming a function $k(T)$. Moreover, batch-wise experimentation usually implies that reaction media, biocatalyst, and the experimental setup are subjected to small deviations, such as fluctuation of the temperature, inexact processing times, or manual pipetting errors. Here, we simulated small deviations in the initial substrate S_0 to mimic such a batch effect.

The hybrid model uses a latent GP (i.e., one where the predicted outcome cannot directly be measured) to describe the relationship between reaction rate k and the temperature T :

$$k(T) \sim \mathcal{GP}(m(T), k(T, T')). \quad (2)$$

For a given reaction temperature, inserting the respective k back into Equation 1 yields the product concentration. In addition, we can model the initial substrate in Equation 1 as

a local parameter per batch $S_{0, \text{batch}}$. Combined, this results in a hybrid model that yields product concentrations, which can be directly compared to the measured observations.

In contrast, the reaction rate k for a fixed temperature is first extracted by simple regression for the naive GP approach, as it is common in biocatalysis. It is then used to directly train a GP to predict the influence of temperature. The results of this approach and the hybrid model and a naive GP are compared in Figure 7 (right panel), which shows the true function as well as samples from the two GPs to compare the mean and uncertainty predictions.

The comparison shows how the hybrid model learns more efficiently from the seven observations provided per reaction, while the native GP has a larger uncertainty. In light blue, Thompson sampling is visualized as an acquisition function. For the native GP, the deviation of the mean from the true function is larger, which can be best seen around 30° , where the observations of experiments from two different batches deviate. The hybrid model can estimate the initial substrate concentration per batch and thus corrects for this bias. Combined with the information from the reaction kinetics, this leads to an efficient model prediction. Note that while replication of experiments is generally not recommended for BO (Section 3), models quantifying batch effects or positional biases benefit from a few replicates of identical designs to quantify challenging experimental effects.

In conclusion, this example highlights how using a hybrid model, consisting of a GP in combination with mechanistic equations for reaction kinetics, can be more accurate and efficient for optimization problems with limited data availability and existing domain knowledge, which is often the case in biocatalysis and other domains of bioprocess engineering.

assumptions (von den Eichen et al. 2022; Siedentop et al. 2025; Lapiere et al. 2025). We follow this approach in the pH optimization case study.

In contrast, scenarios that involve strong mechanistic prior knowledge, such as systems governed by Monod or Michaelis-Menten kinetics, may justify the use of nonzero mean functions. The same applies to systems that exhibit known asymmetries due to external constraints, such as a sharp decline in productivity beyond a specific temperature caused by enzyme denaturation. Parametric mean functions or constant offsets can be incorporated to capture these trends. Approaches for incorporating mechanistic insight into GP models are discussed in Section 5.5. More broadly, incorporating models as mean functions is an instance of hybrid modeling. An alternative that also falls into this category is dedicated probabilistic process models. Such models not only enable the incorporation of mechanistic insights but can also infer and compensate for systematic biases in the experimental system (Helleckes et al. 2023; Helleckes, Küsters, et al. 2024). This approach to hybrid modeling is thoroughly illustrated with a second case study in Box 2.

3.2.2 | Kernel

In contrast to the mean function, selecting an appropriate kernel requires more nuanced consideration in bioprocess engineering applications. Smooth or locally periodic kernels may be suitable, depending on the experimental context and the anticipated behavior of the underlying process. When strong prior knowledge is available, specialized kernels can be employed to encode structural patterns in the response surface. For instance, locally periodic kernels are particularly suitable for processes exhibiting time-resolved cyclic dynamics (Ortmann et al. 2019), such as pulsed substrate feeding in fed-batch fermentations. By embedding periodicity into the kernel, they enable more confident extrapolation even when only a few cycles are observed. Despite their potential, such kernels remain underutilized in bioprocess engineering, and implementation guidelines remain scarce.

In most other cases, bioprocess engineering applications are well represented by smoothing kernels. The squared exponential kernel and the more general Matérn kernel family are particularly well-suited. Their smoothing characteristics generally align well with typical expectations for a wide range of bioprocess responses (Siedentop et al. 2025; von den Eichen et al. 2022; Rosa et al. 2022). We therefore recommend adopting these kernels in the absence of strong prior knowledge that would favor more specialized alternatives. We recommend that practitioners evaluate multiple kernels during initial model fitting. Visualizing random sample functions drawn from the posterior distribution under different kernel choices provides valuable insight into how each kernel encodes prior assumptions about the behavior of the system. For example, sample functions from the squared exponential kernel typically exhibit very smooth, gently varying behavior, while Matérn kernels produce rougher sample functions. Comparing these qualitative behaviors with prior expectations about the system can guide the selection of an appropriate kernel, which in many bioprocess applications is the squared exponential kernel. In our case study, since enzymatic activity is expected to vary smoothly with pH, we follow this rationale and select the squared exponential kernel.

3.2.3 | Hyperparameter Priors

Once the mean function and kernel are selected, the critical task of configuring hyperparameter priors arises. For the widely used squared exponential kernel, these hyperparameters are the length scale ℓ , signal variance σ_f , and noise variance σ_n (Section 2.3). As discussed in Box 1, these hyperparameters are inferred following Bayes' theorem, which combines the specified priors with the information provided by the observed data. The choice of these priors can strongly influence the resulting posterior distribution, and thus the model fit (Hvarfner et al. 2024), especially in low-data regimes.

In practice, hyperparameters are typically estimated by maximizing the marginal likelihood of the GP. However, the marginal likelihood surface is often highly nonconvex and may exhibit multiple local optima (Rasmussen and Williams 2006). In low-data regimes, the problem becomes ill-posed, as numerous configurations can explain the data equally well. The lack of identifiability means that different hyperparameter configurations can explain the data equally well, which

complicates the numerical optimization process. It is advisable to employ multi-start or other global optimization strategies for hyperparameter fitting to mitigate these issues. Constraining the search space to physically plausible hyperparameter ranges (e.g., ruling out unrealistically small length scales) can further stabilize optimization.

Figure 5 illustrates this challenge using our case study. Two distinct GP fits are shown for the same five initial data points, differing only in their respective hyperparameter priors. Given that hyperparameter priors alone can determine whether a model is well-fitted, over-fitted, or under-fitted, we strongly advise practitioners to carefully assess their prior settings and inspect the resulting GP fits. Visualizations of model predictions, uncertainty bands, and posterior samples are essential for diagnosing poor fits that may not be apparent from numerical statistics alone.

To improve GP model fits in scenarios with limited prior knowledge, we offer the following recommendations. For length scale priors, we recommend encoding how rapidly the underlying system behavior is expected to vary across the defined parameter space. A generally promising approach is to set the length scale in the range of one-quarter to one-half of the total parameter space width. As dimensionality grows, it is advisable to scale the length scale prior at a rate of $\ell \propto \sqrt{d}$, where d is the number of design variables, provided this aligns with prior domain knowledge. Such an adjustment improves BO performance in higher dimensions by offsetting the larger average distances between data points (Hvarfner et al. 2024).

Signal variance and noise variance often exhibit a trade-off during model fitting. The GP can attribute observed variability either to underlying system behavior or to measurement noise (Rasmussen and Williams 2006). As illustrated in Figure 5, poor prior specification can cause the model to misattribute this variability. To avoid such scenarios, we recommend avoiding informative priors for signal variance. In contrast, the prior for the noise variance should be informed by prior knowledge of the measurement system, such as precision estimates from previous calibration studies or historical experimental data.

In the pH case study, we applied these recommendations by setting the prior for the length scale to one-third of the total parameter range. Additionally, we selected a prior for the noise variance based on an estimated measurement uncertainty of 5%. For the detailed implementation and an interactive Jupyter notebook to explore how different hyperparameter priors affect model fit, we refer to the accompanying GitHub repository (Siska et al. 2025).

3.2.4 | Isotropic and Anisotropic Length Scales

In multidimensional parameter spaces, practitioners may choose between isotropic kernels, which use a single length scale shared across all design variables, and anisotropic kernels, often referred to as automatic relevance determination kernels, which assign separate length scales to each dimension. Isotropic kernels reflect the assumption that all design variables influence the underlying function at similar scales. In contrast, anisotropic kernels enable the model to capture variable-specific sensitivities and directional effects. Liang et al. (2021) report that anisotropic kernels tend to outperform isotropic

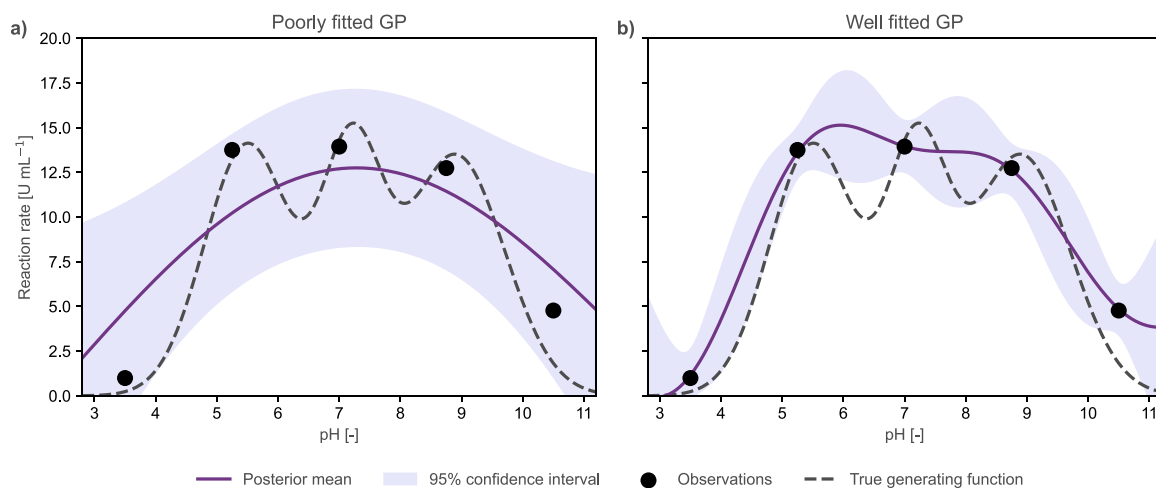


FIGURE 5 | Comparison of well- and poorly-fitted GPs. Both panels display GP models fitted to the five initial data points from the pH optimization case study. The true relationship between pH and reaction rate (dashed curve) remains unknown. Subplot (a) illustrates the effect of inappropriate hyperparameter priors. Here, the length scale prior is centered at the full parameter space width, and the noise variance prior assumes 50% measurement noise. This configuration leads to underfitting. The model attributes most of the variance to noise, and the prediction uncertainty remains high even near the observations. In contrast, Subplot (b) shows a well-fitted GP that provides a plausible interpretation of the observations (black dots), attributing most of the observed variation to the underlying function. Here, we center the length scale prior at one-third of the parameter space and encode in the noise variance prior the belief that our observations carry 10% measurement noise. As a result, the predicted mean (violet curve) closely follows the data, and the uncertainty band narrows around the observed points.

ones when the design variables differ substantially in nature or scale. For example, consider extending the pH case study with a second design variable: temperature, which typically exhibits a single broad optimum. Using a single (isotropic) length scale for both pH and temperature could restrict the ability of the model to capture the sharper pH dependency illustrated in Figure 6. While this added flexibility of anisotropic kernels enhances model expressiveness, it also increases the risk of overfitting (Greif et al. 2025), especially in data-scarce scenarios.

3.3 | Acquisition Function

The acquisition function is the next core component in the BO workflow. It proposes new experiments by leveraging both the predictive mean and uncertainty estimates of the GP to balance exploration and exploitation. This section focuses on practical considerations for the commonly used acquisition functions, as well as typical pitfalls when managing the exploration–exploitation trade-off. We restrict our discussion to sequential BO, where one experiment is proposed at a time. Batch BO strategies, which are highly relevant in bioprocess engineering for parallel experimentation, are discussed in Section 5.

3.3.1 | Balancing Exploration and Exploitation

EI and UCB are among the most widely used acquisition functions in the literature. They are readily available in open-source libraries (Balandat et al. 2020; Dürholt et al. 2024; Gardner et al. 2018; Head et al. 2021), easy to interpret, effective in low-dimensional problems, and often yield good performance with default settings. However, both rely on user-defined hyperparameters to adjust the trade-off between exploration and exploitation.

Selecting these hyperparameters can substantially influence optimization outcomes, particularly during early iterations when the surrogate model is fitted to limited and potentially unrepresentative data. Overly exploitative strategies increase the risk of premature convergence to local optima. Georgiou et al. (2025) report that exploitative acquisition functions are especially prone to failure when the initial data set is sparse or clustered. This issue is particularly relevant in bioprocess engineering, where experiments are costly and the resulting data are often noisy and scarce. Nonetheless, EI and UCB remain well-suited options, provided that the initial data are of sufficient quality and hyperparameters are judiciously selected.

Optimizing acquisition functions is nontrivial, as they are often highly nonconvex and exhibit flat landscapes across large regions of the parameter space (Binois and Wycoff 2021). These characteristics can lead to suboptimal experiment proposals. Efforts to mitigate these issues include the development of logEI, a reformulation of EI that improves numerical performance and has been shown to result in significantly improved optimization outcomes (Ament et al. 2023). Additionally, practitioners of BO are advised to apply multi-start strategies and stochastic evolutionary algorithms to improve the robustness of acquisition function optimization (Georgiou et al. 2025; Vincent and Jidesh 2023).

Figure 6 illustrates the application of EI to the pH case study. Starting with the initial data set in Iteration 1, EI selects exploratory experiments that expand coverage of the parameter space, thereby reducing predictive uncertainty. As model uncertainty decreases, EI progressively shifts toward more exploitative proposals. By Iteration 6, the algorithm converges to the global optimum, having effectively learned the shape of the objective function near its peak and identified the optimal pH for the crude extract at 7.2.

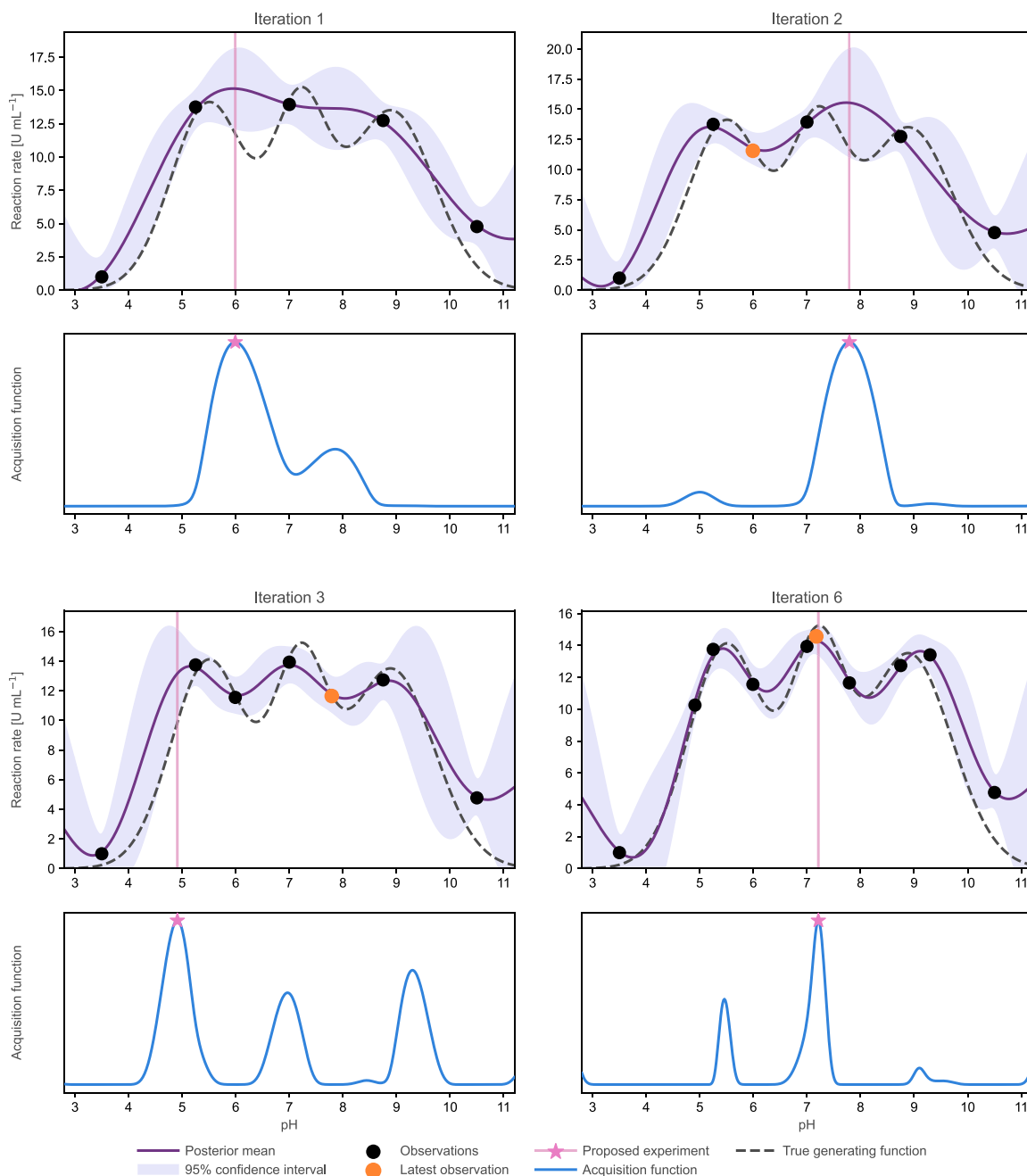


FIGURE 6 | Progression of GP model and acquisition function during BO of the pH case study. Panels illustrate the evolution of the GP model throughout multiple iterations (Iterations 1, 2, 3, and 6) of the BO workflow. For each iteration, the upper panels display the GP model fitted to the available observations. The data point proposed in the previous iteration is highlighted in orange. The lower panels show the corresponding EI acquisition function. The pink line marks the location of the next proposed experiment, corresponding to the maximum of the EI. No replicated experiments are performed. As the optimization proceeds, the GP model incrementally improves its approximation of the system behavior, while the EI acquisition function dynamically adjusts to balance exploration and exploitation.

3.3.2 | Hyperparameter-Free Alternative

Thompson sampling offers a practical alternative to traditional, hyperparameter-dependent acquisition functions. By randomly sampling functions from the GP posterior, it naturally balances exploration and exploitation and fosters broader exploration of the parameter space. This stochasticity makes it particularly robust in early BO iterations when data are sparse and uncertainty is high. More advanced information-theoretic acquisition functions, such as Predictive Entropy Search (Hernández-

Lobato et al. 2014), share conceptual ground with Thompson sampling, as they also rely on sampling from the GP posterior rather than using dedicated hyperparameters. These acquisition functions propose new experiments by estimating which would most reduce the uncertainty about the location of the global optimum. This property makes these acquisition functions especially well-suited for batch BO (see Section 5). However, their high computational cost has thus far limited their use in bioprocess engineering applications.

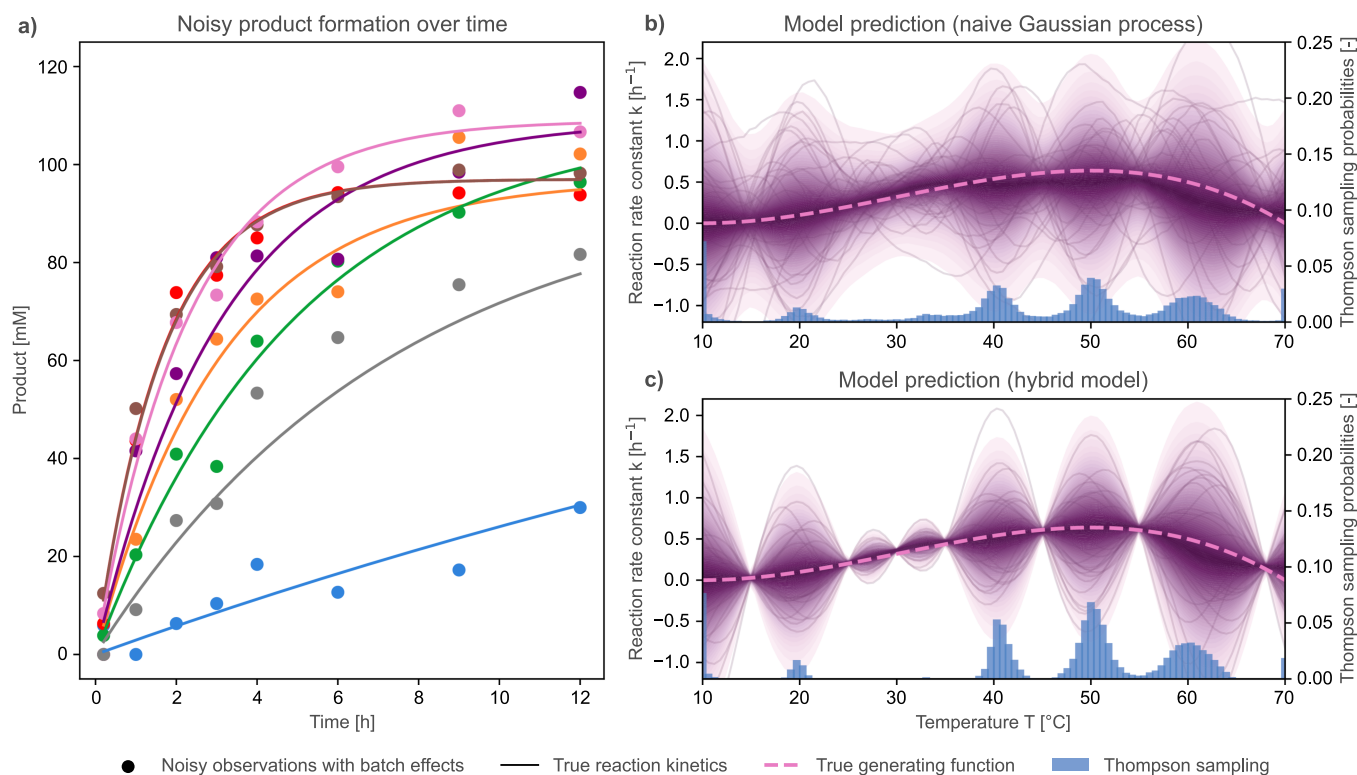


FIGURE 7 | Biocatalysis case study with batch effects, comparing a naive GP and a hybrid model as surrogates. This case study describes the optimization of the reaction temperature of an uncharacterized enzyme. (a) The time series data of product formation for a single reaction at different temperatures can be measured. Experiments are conducted batch-wisely in shake flasks (four experiments with different designs per batch), between which the substrate concentration changes due to an undesired batch effect. This example demonstrates more complex and thus realistic experimental conditions. In the left panel, the noisy observation of product concentrations over time can be seen, influenced by the respective batch effect. The following temperatures were used to generate *in silico* data sets: 15° (blue), 25° (green), 30° (orange, purple), 35° (pink), 45° (red), 55° (brown), and 68° (gray). The right two panels (b, c) compare the fitted surrogate model (purple), as well as the distribution of probability when repeatedly applying Thompson sampling as optimization policy on the data set of the current iteration of BO (blue bars). The purple lines visualize random samples from the GPs, thus indicating the uncertainty around the mean prediction. A GP hybrid model that is taking a random batch effect and the first-order mass action into account (c) captures the true relationship (pink dashed line) more efficiently than a naive GP model (b). The latter was trained on reaction rates that were extracted from the time series data in (a) via regression, which is a standard method in biocatalysis.

Thompson sampling, too, can become computationally demanding in multidimensional settings, due to the sampling from the GP posterior. In particular, drawing samples over a discretized grid in the parameter space becomes increasingly prohibitive as dimensionality grows, with the computational cost scaling cubically with the grid resolution (Rasmussen and Williams 2006). Several strategies have been proposed to mitigate this burden, such as using more efficient sampling schemes or GP approximations based on random Fourier features (Rahimi and Recht 2007; Hernández-Lobato et al. 2014). While such techniques can reduce runtime, they often compromise fidelity by failing to accurately capture the true GP posterior in high-dimensional spaces (Ton et al. 2018; Wesel and Batselier 2021). As experimental systems become more complex and parameter spaces expand, continued advances in scalable surrogate modeling and acquisition strategies will be essential for enabling practical and reliable BO.

Beyond the acquisition functions discussed here, Section 5 covers more sophisticated strategies tailored to accommodate complex experimental realities such as parallel experiments, multistep planning, and conducting experiments across scales. However, these methods introduce substantial computational

and modeling overhead. Such advanced methods should thus only be adopted when the structure or constraints of the experimental system justify their additional cost, for example, when long experiment turnaround times make the improved sample efficiency worth the computational overhead.

4 | Challenges of Applications in Bioprocess Engineering

While the previous section presented practical guidelines based on a base case, this section focuses on more specific, yet commonly encountered challenges and additional considerations in the application of BO to bioprocess engineering.

4.1 | Handling Experimental Biases in Bioprocess Engineering

Following the general discussion, we now focus on several specific challenges arising from experimentation in bioprocess engineering. In particular, we highlight *batch effects*, *positional biases*, and the *calibration of measurement devices*.

Batch effects refer to systematic biases between experimental batches or samples, caused by factors unrelated to the design variables (Leek et al. 2010). Examples include variations in results when experiments are conducted on different days or by different experimenters. In bioprocess engineering, batch effects are common due to the complexity of experimental systems, measurement technologies, and the inherent variability of living organisms (Goh et al. 2017; Han and Li 2022). Addressing batch effects in BO requires extending the standard GPs framework, as illustrated by an empirical example in Box 2.

Another common intricacy is positional bias, which frequently arises when working with microtiter plates (Carau et al. 2017). These biases may result from temperature gradients (Burt et al. 1979), pipetting errors (Pandya et al. 2010), the fabrication of microtiter plates (Kricka et al. 1980), or the plate reader (measurement device) (Harrison and Hammock 1988). In classical DoE, batch effects and positional biases are typically addressed through randomization and blocking (Montgomery 2020), techniques that are also widely applied in bioprocess engineering (Roselle et al. 2016). However, these effects can alternatively be incorporated directly into the model. Such model-based strategies are less commonly applied in BO for bioprocess engineering, likely because most available software packages do not support them by default. Nonetheless, recent studies such as von den Eichen et al. (2022) and Helleckes et al. (2023) provide useful examples. In practice, it is worth investigating these experimental effects prior to applying BO. It is advised to either eliminate them to be able to use the basic practical guidelines outlined in Section 3, or to model them explicitly as illustrated in Box 2 for batch effects.

Finally, many objective functions in bioprocess engineering cannot be measured directly, for instance, titers inferred from fluorescence or absorbance assays. In such situations, an accurate calibration of the measurement device is essential to convert the raw measurement signals into the desired target quantity. This can be achieved with the help of calibration models, which in practice are often applied before the converted data (e.g., an enzyme activity or a titer) is used in BO. In this context, the critical role of a well-suited calibration model should be emphasized. Incorrect assumptions, such as linearity or neglecting a signal drift over time, can strongly bias the surrogate model and mislead the selection of subsequent experiments. Using likelihood-based calibration models, experienced users can directly incorporate these assumptions into the surrogate model (Helleckes et al. 2022).

These examples illustrate that the intricacies of bioprocess engineering require continuous fine-tuning of the surrogate model and tailored selection of priors and acquisition functions, as discussed further in Section 5.

4.2 | Perspectives on User-Friendliness and Greater Adoption of Bayesian Optimization

In contrast to traditional DoE, which typically follows a well-defined and structured design, setting up BO involves a series of decisions that may be challenging for nonexperts. These include selecting the surrogate model, acquisition function, initial sampling strategy, and stopping criteria, along with potentially ongoing adjustments during implementation, making BO more flexible but also more complex. The development of bioprocess-specific

frameworks or guidelines, such as this paper, could lower the entry barrier and facilitate a broader adoption within the field.

Another major barrier is the limited availability of user-friendly software tools. Whereas traditional DoE is supported by various commercial software packages with graphical user interfaces (e.g., JMP by the SAS Institute or MODDE by Sartorius), most BO applications in bioprocess engineering have been realized using software that requires intermediate to advanced programming and machine learning expertise. Some studies have developed custom BO software tools for bioprocess applications using MATLAB (Freier et al. 2016; Freier and Von Lieres 2017) or Python (Thompson et al. 2023). However, most implementations rely on well-established general-purpose BO libraries in Python. The most frequently used libraries include:

- BoTorch and GPyTorch (e.g., Claes et al. 2024; Cosenza et al. 2022, 2023);
- scikit-learn and scikit-optimize (e.g., Jäpel and Buyel 2022; Pandi et al. 2022; Narayanan et al. 2025); and
- GPYOpt (e.g., Kanda et al. 2022; Yoshida et al. 2023).

In addition, several user-friendly packages have been recently developed in the chemical industry, including BoFire (Dürholt et al. 2024), created by a consortium including BASF, Evonik, and Boehringer Ingelheim, BayBe (Fitzner et al. 2025) by Merck KGaA, and obsidian (Stone et al. 2025) by Merck & Co. Although these packages offer extensive options for surrogate models and acquisition functions, thus greatly simplifying the development of BO workflows, they still require intermediate programming skills. Moreover, their documentation and application programming interfaces are often designed for the machine learning community, posing an additional challenge for users without that background.

Finally, bioprocess engineers often come from backgrounds in the natural sciences or biochemical engineering, which may not include training in BO methods, the associated mathematics, or programming. Targeted training programs can play an essential role in driving the adoption of these methods. Emerging degree programs that combine biotechnology or natural sciences with computer science can improve such interdisciplinary competencies. In addition, to support broader application of BO, user-friendly tools and domain-relevant examples are crucial. We hope that the guidelines provided in Section 3 will contribute to the development of decision guidelines and their implementation in user-friendly BO tools tailored to the needs of bioprocess engineering.

5 | Advanced Bayesian Optimization for Bioprocess Engineering

As illustrated in previous sections, BO has emerged as a framework that can guide experimental design in biotechnology. After outlining general recommendations and current challenges for applying BO in bioprocess engineering, we now turn to advanced BO extensions. We believe that these extensions can be leveraged by domain scientists in the future to overcome the rigid, stage-by-stage paradigm of bioprocess development. Rather than optimizing strain selection, process conditions, and scale-up in isolation, holistic bioprocess development must acknowledge their

interdependencies. BO extensions, such as batch BO, multi-fidelity BO, and mixed-variable BO, can unify the traditional stages into one coherent optimization loop, paving the way for fully integrated bioprocess development.

5.1 | Batch Bayesian Optimization in the Era of High-Throughput

Traditional BO is a sequential strategy: at each iteration, a surrogate model (often a GP) is updated using observations, and a single new experiment is selected by maximizing an acquisition function. This paradigm, however, does not adequately reflect the experimental realities of bioprocess engineering. In practice, experiment preparation (e.g., preculture, strain construction) and analytics (e.g., chromatography or mass spectrometry) often dominate timelines, yet can be performed in parallel. Running a batch of four experiments simultaneously, for example, in parallelized bioreactor systems, requires no or only marginally more time than a single experiment, reducing the overhead significantly. Automated high-throughput platforms further enable parallel experimentation at scale. These experimental realities motivate the use of *batch BO* (Dai et al. 2023; González and Zavala 2023).

Batch BO extends the traditional framework by selecting a set of q points (experiments) to evaluate in parallel at each iteration. Selecting a batch of q points introduces additional complexity beyond standard BO, as the points must collectively balance exploration and exploitation. Common strategies include:

- *Greedy or sequential conditioning*: Points are selected one at a time, each conditioned on the previous selections within the batch (Gonzalez et al. 2016).
- *Posterior sampling (Thompson sampling)*: Points are selected by drawing random samples of functions from the GP posterior and selecting their maximizers (see Section 2.3). The inherent randomness of independent draws controls the diversity within the batch (Kandasamy et al. 2018). Determinantal point processes can extend on this concept by ensuring that selected points provide complementary information (Nava et al. 2022).
- *Joint acquisition maximization*: The acquisition function, whether it is EI, UCB, or predictive entropy search (Hernández-Lobato et al. 2014), is optimized jointly over the q points. Though this approach can be the most principled, it can be computationally demanding to the point of intractability as q increases (Wilson et al. 2018).

Beyond batch BO, recent work highlights that traditional acquisition functions such as EI and UCB can limit performance because they are inherently **myopic**, meaning they focus only on selecting the next experiment without considering how that choice affects future rounds of experiments. Whereas batch BO proposes several experiments to run simultaneously, non-myopic acquisition functions optimize a sequence of proposals and therefore improve planning across iterations rather than within a single batch. Rollout-based approaches demonstrate that even a short lookahead horizon of up to four steps can outperform purely myopic policies by mitigating premature exploitation (Yue and Kontar 2020). Reinforcement-learning-based methods further extend these ideas by learning near-

optimal multistep policies that outperform standard one-step BO across a range of settings (Cheon et al. 2024). As high-throughput bioprocess systems increasingly rely on multiround optimization cycles, combining batch execution with non-myopic acquisition logic represents a promising direction for future BO workflows. Importantly, nonmyopic reasoning is complementary to batch BO: while batch selection accelerates experimentation within a round, nonmyopic strategies guide how rounds should proceed over time. As high-throughput bioprocess platforms increasingly rely on iterative optimization cycles, combining batch execution with nonmyopic acquisition logic offers a promising path forward.

The integration of batch BO into bioprocess engineering workflows represents a natural progression given the rise of high-throughput experimentation systems. Batch BO provides a practical framework for leveraging high-throughput experimental capabilities, enabling faster, more data-efficient optimization in bioprocess development. However, some modeling challenges continue to hinder its seamless adoption.

Miniaturization and automated, reproducible experimental workflows expose systematic biases, such as batch and positional effects, as described in Section 4. The prevalence of at-line measurement devices (e.g., photometers) and their associated assays demands calibration models. Addressing these complexities often requires dedicated hybrid models that extend the standard GP framework with mechanistic components (see Box 2). These models must be iteratively refined in tandem with prior assumptions and experimental workflows, forming a time-consuming cycle that limits accelerated bioprocess development. To unlock the full potential of (batch) BO in high-throughput experimentation, the modeling community must shift from bespoke process models to shared libraries of reusable model components that capture systematic biases and establish formal knowledge-transfer practices (e.g., shared documentation, code repositories) to share methodologies and insights across processes.

5.2 | Multi-Fidelity Bayesian Optimization for Bioprocess Scale-Up

Experiments in bioprocess engineering can be conducted at varying levels of fidelity (accuracy of the experimental system). For example, small-scale microtiter plates can rapidly generate large amounts of data at relatively low cost and low fidelity, while pilot-scale bioreactor runs provide higher-fidelity data that more accurately reflects industrial conditions; however, they are time-consuming, costly, and low-throughput. Traditional BO frameworks do not distinguish between such data sources. In contrast, multi-fidelity Bayesian optimization (MF-BO) is explicitly designed to integrate and balance information from these different levels of fidelity.

Specifically, MF-BO models both the objective function (e.g., yield, productivity, titer) and the fidelity level of each data source. It treats experimental outcomes as being drawn from a hierarchy of fidelities, where:

- *Low-fidelity sources* (e.g., small-scale, simplified models, in silico simulations) are cheap and fast to query but provide only approximate insights into real-world performance.

- *High-fidelity sources* (e.g., pilot plant runs, advanced simulations) are accurate but expensive and slow to obtain.

By jointly learning from both types of data, MF-BO can efficiently explore the parameter space, using low-fidelity data to guide the search while reserving high-fidelity evaluations for only the most promising candidates (Savage et al. 2024). Importantly, MF-BO can facilitate holistic bioprocess development across scales. It has the potential to integrate traditionally sequential bioprocess development stages, namely the optimization of process conditions and scale-up, and unite small-scale screening with large-scale validation.

5.2.1 | Bioprocess Scale-Up

Scaling bioprocesses involves navigating a landscape where conditions that perform well at small scales do not always translate directly to larger scales due to factors like mass transfer, heat transfer, and mixing limitations. MF-BO provides a structured, data-driven way to traverse scales efficiently by rapid exploration at small scales (e.g., microtiter plates, small bioreactors) to map broad trends and identify promising regions of the parameter space. High-fidelity experiments are targeted only for conditions that appear optimal or highly informative (e.g., pilot-scale bioreactors), conserving valuable time and resources. At every iteration, MF-BO algorithms select not only the experimental conditions, but also the scale (fidelity) at which these experiments are to be run. When used for scale-up, MF-BO thereby balances cost and accuracy by ensuring that the budget and experimental effort are concentrated where they will have the greatest impact on process development.

5.2.2 | Multi-Fidelity Bayesian Optimization Algorithms

MF-BO frameworks are designed to capture both the dependency between fidelities and between design variables (Forrester et al. 2007). Different MF-BO methods capture the fidelity dependencies differently. In general, their acquisition functions consist of different terms, including a penalty for cost (e.g., time, budget), another penalty for loss of information from using a lower (cheaper) fidelity from a higher (more expensive) one, and a final term for exploration–exploitation trade-off. The latter is also used in traditional BO to make decisions; in this case, however, the decisions comprise the fidelity as well as optimal experimental conditions. It is also possible to combine batch and multi-fidelity BO to create a framework that screens over multiple fidelities in high-throughput (Folch et al. 2023; Moss et al. 2021; Martens et al. 2025). In this way, multi-fidelity BO offers a framework for bridging the gap between small-scale screening and pilot-plant validation, supporting faster, more cost-effective bioprocess development.

5.3 | Handling Discrete Variables in Bayesian Optimization

Bioprocess engineering usually entails the optimization of both biocatalyst and process conditions, for example, the selection of an enzyme out of a candidate library while optimizing the reaction temperature and pH. Most of the solutions discussed so far address the optimization of continuous design variables, whereas catalyst or strain selection are inherently discrete

problems. Such joined optimization problems are referred to as **mixed-variable problems**, which handle both discrete and continuous parameters.

5.3.1 | Pragmatic Approaches for Noncontinuous Problems in Bioprocess Engineering

Mixed-variable domains comprise continuous and discrete variables, with **categorical variables** constituting a specific subclass of the latter. The simplest way to handle categorical problems is one-hot encoding, which entails creating binary dummy variables that encode which data belongs to which category in the discrete space. Although simple to implement, this method requires additional dummy variables that take values of 0 or 1 for each category, which quickly becomes inefficient for larger sets of catalysts, strains, or media components.

Generally, discrete variables can be treated as continuous variables and solutions are then rounded to the next integer to obtain the selected candidates, a heuristic that, however, introduces discontinuities that complicate the optimization of the acquisition function (Daulton et al. 2022). Over the past years, several other modifications of kernels for discrete variables have been proposed, such as transformations on the discrete parameters (Garrido-Merchán and Hernández-Lobato 2020) or splitting continuous and discrete design variables in different kernels and using sums or products of both to combine them into a valid mutual kernel (Ru et al. 2020). Modern libraries such as BoTorch provide various options, including kernels that can capture the correlation of discrete variables, for example, the intrinsic coregionalization model (Bonilla et al. 2007).

5.3.2 | Tailored Solutions for High-Dimensional Parameter Spaces

Many bioprocess applications involve *high-dimensional* categorical or mixed-variable parameter spaces. Strain libraries or promoter-plasmid combinations can comprise hundreds of variants, and catalyst families or media formulations may involve large sets of discrete alternatives. In such settings, a single global surrogate often becomes unreliable because of multimodality and the combinatorial growth of the discrete parameter space. Recent methodological developments address these challenges through complementary strategies that balance local and global search. Trust-region approaches restrict modeling to adaptively sized neighborhoods around promising regions, avoiding the computational burden of fitting global surrogates over the entire discrete space (Wan et al. 2021). Alternatively, subspace embedding methods enable optimization in lower-dimensional target spaces by partitioning and iteratively refining the parameter space (Papenmeier et al. 2023). Beyond these structural approaches, specialized kernel formulations can encode relationships between categorical variables, and when variants carry meaningful structure such as sequences or molecular features, embedding-based methods become particularly advantageous.

This latter case, where variants carry meaningful structure, is particularly common in bioprocess applications, where the integration of large databases spanning diverse biological

candidates drives high-dimensional parameter spaces. When these structures can be expressed, it is advantageous to expand on the above approaches by employing embedding or deep-kernel-based methods that learn similarities between candidates rather than treating each as an independent category. Strategies for this are discussed in Section 5.3.3.

Even with the increasing prevalence of automated experimentation, very high-dimensional continuous parameter spaces remain uncommon in bioprocess engineering, and typical workflows remain within the practical limits noted by Hvarfner et al. (2024). Should an application nonetheless exceed these bounds, scalable high-dimensional BO methods such as TuRBO (Eriksson et al. 2019) or SAASBO (Eriksson and Jankowiak 2021) provide viable alternatives, though their discussion lies outside the scope of this guide.

5.3.3 | Advanced Methods for Encoding Biocatalyst Information

Beyond the selection of discrete candidates, details about the biocatalyst, for example, genetic information on a cell line or the three-dimensional structure of an enzyme, can be encoded into the kernel to enhance the learning process. Deep kernel learning (A. G. Wilson et al. 2016) extends BO by combining the flexibility of deep neural networks with the uncertainty quantification of GPs. While traditional BO models with simple kernels work well in low-dimensional structured problems, they often struggle in complex, high-dimensional domains common in bioprocess engineering, such as the amino acid sequence of a protein, or molecular structures. Deep kernels address this by transforming inputs through a neural network into a latent space where a standard GP kernel can operate, enabling BO to handle challenging parameter spaces more effectively (Gómez-Bombarelli et al. 2018).

In bioprocess engineering, deep kernels show promise in applications like protein engineering and synthetic biology. For example, (variational) autoencoders, machine learning models that are specifically suited to encode complex information to an embedding or “fingerprint,” can map protein sequences into a compact latent space that captures key properties like stability or binding affinity. BO can then efficiently search this space for optimal designs (Romero et al. 2013). These techniques of latent BO are already extensively used in the realm of chemical engineering (e.g., Moss et al. 2020; Griffiths and Hernández-Lobato 2020), thus showing potential for the encoding of biological properties in future applications.

Beyond deep kernel learning, neural networks following the transformer architecture are of interest. Transformers became famous in large language models (LLMs) such as ChatGPT, and are increasingly applied in the era of protein language models to encode protein sequences. The most famous example is AlphaFold (Jumper et al. 2021), which can be used to predict protein folding from sequence data. Similar to autoencoders, protein language models are efficient in generating embeddings out of protein sequences. In addition, graph neural networks are frequently applied to represent small molecules as molecular fingerprints (Duvenaud et al. 2015; Kwon et al. 2022). Protein language models and graph neural networks thus represent further options to encode biocatalyst information in the kernel for future applications.

Beyond these representation learning approaches, recent work has explored leveraging LLMs directly as part of the kernel. For instance, GOLLuM (Ranković et al. 2025) introduces a deep-kernel framework in which design points are first embedded using an LLM and the embedding space is then jointly optimized with the GP marginal likelihood. Such LLM-based kernels offer a flexible way to incorporate heterogeneous or textual bioprocess descriptors (e.g., strain annotations, media descriptions) into BO without manually constructing composite kernels.

5.4 | Transfer Learning for Bayesian Optimization in Bioprocess Development

In bioprocess engineering, researchers often work with families of related systems—such as optimizing media formulations across different microbial strains or fine-tuning enzyme activity across a panel of homologous proteins. In these scenarios, data from past experiments on related systems can provide valuable information to accelerate optimization of a new system.

Transfer learning for BO offers a way to accelerate experimental campaigns by leveraging knowledge from related tasks. A key approach for transfer learning in BO is the use of multioutput GPs, also known as multitask GPs or cokriging. Multioutput GPs model correlations not only between inputs and outputs but also across different, related tasks (e.g., strains, scales, or enzyme variants). By capturing these relationships, multioutput GPs allow prior data to improve predictions and guide the selection of new experiments for the current task.

Typical advantages of transfer learning in bioprocess development include:

- *Leveraging historical data:* make the most of prior experimental campaigns by reutilizing them in new tasks.
- *Supporting generalization across related systems:* facilitates process development when moving between strains, enzymes, or scales.

Interestingly, transfer learning and multi-fidelity are closely related, as multi-fidelity can be seen as a particular case of transfer learning where the learning is transferred across scales. Early case studies explore options for transfer learning in BO for bioprocess engineering, for example, work by Hutter et al. (2021), who explored the encoding of different pharmaceutical products with embeddings, or Helleckes, Wirnsperger, et al. (2024), who investigated meta learning of prior distributions for GP parameters across cell lines. However, most advanced options of transfer learning between processes and different biocatalysts remain underexplored and pose an interesting challenge to the machine learning community.

5.5 | Incorporating Models as Mean Functions in Bayesian Optimization

Similarly to transfer learning, further incorporation of domain knowledge can enhance the traditional BO workflow, for example, through the use of empirical models as mean functions or the application of hybrid models as surrogates. In many bioprocess engineering applications, prior knowledge about a

process is available in the form of mechanistic models, empirical correlations, or simulation frameworks (e.g., kinetic models, bioreactor dynamics, pathway simulations). A natural extension of BO is to incorporate such models directly into the surrogate by using them as the *mean function* of the GPs.

In this setup, if $m(\mathbf{x})$ is the output of a mechanistic or empirical model at input \mathbf{x} , the GP is applied to the residual r :

$$y(\mathbf{x}) = m(\mathbf{x}) + r(\mathbf{x}), r(\mathbf{x}) \sim \mathcal{GP}(0, k(\mathbf{x}, \mathbf{x}')).$$

While not yet standard in off-the-shelf BO frameworks, the integration of models as mean functions is an easy way to combine domain knowledge with data-driven design in bioprocess engineering.

Beyond the use of mechanistic models as GPs mean functions, more complicated hybrid model structures are available. Here, the GPs are replaced altogether by a probabilistic model. Such an extension is discussed in Box 2.

5.6 | Further BO Extensions

Other extensions to BO relevant in bioprocess engineering include preferential BO (using user preference over different Options A and B) (Gonzalez et al. 2017; Coutinho et al. 2024), human-in-the-loop (using human expert feedback in the decision making) BO (Savage et al. 2024; Biswas et al. 2024), multiobjective (using more than one optimization target) BO (Hernandez-Lobato et al. 2016), safe BO (using safety constraints, e.g., avoiding unsafe experimental setups) (Petsagkourakis et al. 2021; Paulson et al. 2022), and BO under transition constraints (where experimental conditions cannot be changed arbitrarily between iterations, which may become relevant for continuous processes where parameters like flow rate or pH must be adjusted gradually) (Folch et al. 2024). Finally, the latest trends in LLMs also extend to BO, specifically LLM-in-the-loop BO, where natural-language feedback or prior knowledge provided by an LLM guides early-stage exploration (Kobalczuk et al. 2025; Liu et al. 2024). These methods illustrate promising future directions for integrating foundation models into BO-driven bioprocess development.

6 | Reproducibility and Data Deposition

BO operates at the interface between wet-lab experimentation and computational modeling. As such, rigorous data organization and metadata collection are required across both domains to ensure full reproducibility and scientific validity of experimental and analytical workflows, including research software.

6.1 | Considerations for Experimental Workflows

In bioprocess engineering, experimental workflows are often highly diverse and tailored to specific processes, posing particular challenges for standardized metadata collection. For instance, consider manually conducted biocatalysis experiments involving enzymes that are produced and purified in-house, where each measurement is costly, time-consuming, and yields very limited data. In such scenarios, reproducibility depends on

meticulous record-keeping. Essential information includes enzyme source and purification protocols, preparation and storage conditions of stock solutions, and operational parameters such as temperature profiles and time delays during batch processing (Tipton et al. 2014; Malzacher et al. 2024). Fully automated laboratory platforms ease reproducibility through scripted protocols and consistent experimental execution. To fully realize these benefits, users must also manage the additional layers of metadata, including experiment scripts, proprietary software configurations, and detailed version control of hardware and software components (Jessop-Fabre and Sonnenschein 2019).

Recent advances in metadata standards promote comprehensive contextual documentation. Initiatives such as EnzymeML (Lauterbach et al. 2023), STRENDA (Tipton et al. 2014; Malzacher et al. 2024), and SBML (Hucka et al. 2003; Keating et al. 2020) provide structured frameworks for data exchange and continue to evolve in response to the increasing complexity of modern laboratory practice. These standards capture essential experimental details, such as reaction vessel types (e.g., microtiter plates, shake flasks), reaction conditions (e.g., temperature, agitation speed, substrate concentrations), and procedural steps. However, as automated laboratory platforms become increasingly prevalent, existing standards have struggled to keep pace with the complexity and volume of metadata they generate. To bridge this gap, Mione et al. (2025) have recently proposed a labeled property graph-based framework that integrates with laboratory workflow management systems to enable the automated collection of both experimental and its associated metadata. A key recommendation is to proactively record rich, machine-readable metadata that goes beyond the current requirements of prevailing standards. Once uniquely stored, this information can later be reused and integrated into emerging standards via automated conversion tools. Exhaustive data capture should be considered an investment in long-term reproducibility.

6.2 | Considerations for Computational Workflows

Proper version control and documentation are essential to ensure that algorithmic decisions remain traceable over time. According to the FAIR principles, both data and software should be findable, accessible, interoperable, and reusable (Wilkinson et al. 2016). Importantly, these requirements also apply in proprietary contexts, even when the data or software are not open or freely available.

On the computational side, the open-source software community routinely employs platforms such as GitLab or GitHub to support sustainable software development and reproducible computational science. In BO workflows, git-based version control should likewise track every script, configuration file, and structured data reference, from initial experiment designs through hyperparameter choices, to guarantee full traceability of decision policies and algorithmic performance (Sandve et al. 2013). This is especially critical for GPs, where choices of length scale and noise variance priors often undergo revision between BO iterations.

Hybrid process models in BO (e.g., Box 2) further underscore the necessity to integrate computational and experimental

metadata. Details such as well positions in microtiter plates or batch identifiers for cryogenic cultures must be versioned alongside model configurations to enable detection of systematic biases, including batch or positional effects (Helleckes et al. 2023). Electronic laboratory notebooks can streamline this by both documenting and aggregating essential data in machine-readable formats (Dunie 2017; Potthoff et al. 2019). During hybrid process model development, the iterative feedback between modeling and experimental workflow drives updates to both. All of these human-in-the-loop decisions must be captured.

Importantly, metadata must capture every intervention in the BO workflow, including override or adaptation of algorithmic output, to provide a comprehensive, transparent history of input data, evolving model configurations, and corresponding outputs across successive BO iterations. A holistic approach not only enhances reproducibility but also facilitates retrospective analysis, rigorous model validation, and reproducible evaluation of hyperparameter priors and their influence on model performance. Ultimately, thorough documentation and systematic data deposition are indispensable for ensuring the scientific robustness and translational potential of BO, not only in bioprocess engineering.

7 | Concluding Remarks

BO has garnered interest from scientists across disciplines, not only for optimizing experimental setups, reactions, and materials, but also as a tool for hyperparameter tuning in the context of machine learning workflows. In bioprocess engineering, real-world applications of BO remain nascent, posing several inspiring challenges for the coming years.

From a technical perspective, more experience is required to identify demands and define standards for selecting surrogate models, prior distributions, and suitable acquisition functions. Additionally, some optimization problems require customized algorithms capable of navigating parameter spaces with complex design variables, such as genetic information. While this review presents recommendations and best practices for bioprocess engineering, we hope that these will be tested, evaluated, and further developed by domain scientists in the upcoming years. At the same time, it aims to guide machine learning practitioners in understanding the unique demands of bioprocess engineering and in developing new algorithms and methods tailored to those challenges.

More broadly, overcoming traditional experimental paradigms—such as triplicate designs, which are neither required nor beneficial for BO—remains a practical challenge. Nonetheless, rethinking experimental designs is essential to unlock the full potential of BO, as demonstrated by early pioneering applications. In addition, current developments toward FAIR data and software documentation, all increasingly discussed in the context of artificial intelligence, will greatly facilitate the use of BO.

Overall, we foresee BO evolving from a technically demanding method applied by specialists and in isolated case studies to a standard tool for experimental design in areas such as biocatalysis, strain screening, and bioreactor development. Achieving this transition will require both empowering and sensitizing

domain scientists, as well as familiarizing machine learning engineers with the unique challenges of bioprocess engineering and experimental design. As such, it is important for the field to develop best practices for addressing the specific intricacies of biological systems and their associated measurement setups in the coming years.

Glossary

| | |
|-----------------------|--|
| Acquisition function | A mathematical function that guides the optimization process by quantifying the expected utility of evaluating a particular point, considering both the predicted value and uncertainty. |
| Batch | A group of experiments that are conducted in parallel or in sequence before the resulting observations are collectively analyzed and new experiments are proposed. |
| Categorical variable | A type of discrete variable that takes values from a finite set of categories rather than numerical values. Examples in bioprocess engineering include the choice of strain, or media formulation. |
| Design variable | A controllable parameter or input variable in an experimental design that can be adjusted to optimize the outcome. In Bayesian optimization, these are the variables over which the optimization is performed and which form the parameter space (also referred to as search space or design space). |
| Experiment | The execution of an experimental procedure (wet lab or in silico) where specific design variable values are selected to obtain observations of system outputs. A single experiment may yield a single or multiple observations. For example, one may perform a one-time endpoint measurement of a single key performance indicator or record a time series of multiple values made at regular intervals. |
| Gaussian process | A probabilistic model that defines a distribution over functions, specified by a mean function and a covariance (kernel) function, commonly used as a surrogate model in Bayesian optimization. |
| Heteroscedastic noise | Heteroscedastic noise denotes that the variance of observation noise throughout the parameter space is not constant. A typical example in bioprocess engineering is the turbidity-based optical measurement of biomass concentration, where the observation noise increases for higher concentrations of biomass. |
| Hyperparameters | Specific parameters that control the behavior of an algorithm or model architecture, but are not learned from the data during training. In Bayesian optimization, these include structural elements like the choice of covariance function in a Gaussian process or pre-defined parameters like the length scale, |

| | | | |
|-----------------------------|--|-----------------------|---|
| | which controls how strongly the surrogate model varies between observations. In contrast to model parameters, which are learned from training data, hyperparameters are specified separately before the training process. | Optimization policy | A strategy that determines how to select the next point to evaluate in the optimization process, typically by balancing exploration of uncertain regions with exploitation of promising areas. |
| Kernel | A kernel is a function that defines how similar or correlated two inputs are, typically based on their distance in input space. In GPs, the kernel governs how much information at one input point influences the prediction at another. Also referred to as a covariance function, the kernel is used to construct the covariance matrix that encodes prior assumptions about the smoothness, periodicity, or other structural properties of the function being modeled. This matrix underpins both the generation of sample functions from the GP prior and the conditioning process used to make predictions after observing data. Hence, the kernel plays a central role in shaping the GP's behavior. | Parameter space | The parameter space is the range of all possible input values (e.g., temperature, pH, concentration) in which the algorithm searches for the best result. It defines the limits within which the optimization process explores and evaluates different parameter combinations of the design variables. |
| | | Posterior probability | Updated probability distribution after combining the observed experimental data with the prior beliefs according to Bayes' theorem. |
| | | Prior probability | Probability distribution in Bayes' theorem that describes the initial beliefs about a parameter before observing any data. In the context of Bayesian optimization, the distribution, also referred to as a prior, defines assumptions about the unknown objective function. These are subsequently updated with the experimental data, resulting in the posterior distribution. |
| Machine learning | A subdomain within Artificial Intelligence that focuses on algorithms that learn from (large) data sets to make predictions or decisions. In contrast to expert systems, which emulate human decision making by using fixed, preprogrammed rules of reasoning, Machine Learning focuses on improving model performance by directly learning from data. Moreover, Machine Learning emphasizes the mathematical and statistical foundations of learning algorithms. | Random sampling | The process of generating random instances from a probability distribution, particularly important in Bayesian optimization for estimating uncertainty and exploring the parameter space. |
| | | Surrogate model | A model approximating a system or function through the mapping of input data to its corresponding output. Surrogate models find application when evaluating the true system is costly, and the system is not fully understood. In Bayesian optimization, they approximate the true objective function and are iteratively updated with new data to guide the optimization process. |
| Mixed-variable problem | An optimization problem where the parameter space consists of both continuous (e.g., temperature, concentrations) and discrete design variables (e.g., sampling or feeding times). Discrete variables may be categorical, such as the type of an organism or substrate. | | |
| Myopic acquisition function | Acquisition functions that consider only the immediate expected utility of a proposed experiment. Common examples include expected improvement and upper confidence bound. In contrast, nonmyopic acquisition functions take into account how a proposed experiment will influence future Bayesian optimization iterations. | Uncertainty | Uncertainty represents the potential error or bias in a statistical estimate or measurement. Regarding (surrogate) models, uncertainty expresses the lack of knowledge in predictions, typically quantified through probability distributions or credible intervals in Bayesian optimization. Uncertainty in model parameters may arise from measurement noise or biases in the technical setup of an experiment, as well as from biological variability. |
| Objective function | A mathematical function that quantifies the performance or quality of a system configuration, which is to be optimized (maximized or minimized) during the optimization process. The objective function maps the design variables to a scalar value that represents the system performance, such as product yield, conversion efficiency, or process throughput. | | |
| Observation | The result of inferring a system output or key performance indicator based on one experimental measurement. Each observation is subject to noise arising from both inherent system variability and measurement uncertainties. An experiment can yield several observations. | | |

Author Contributions

Maximilian Siska: conceptualization, methodology, software, visualization, writing – original draft, writing – review and editing. **Emma Pajak:** software, visualization, writing – original draft, writing – review and editing. **Katrin Rosenthal:** methodology, writing – original draft, writing – review and editing. **Antonio del Rio Chanona:** methodology, writing – original draft, writing – review and editing. **Eric von Lieres:** conceptualization, methodology, supervision, writing – original draft, writing – review and editing. **Laura M. Helleckes:** conceptualization, methodology, project administration, software, supervision, visualization, writing – original draft, writing – review and editing.

Acknowledgments

M.S. performed this work as part of the Helmholtz School for Data Science in Life, Earth and Energy (HDS-LEE) and received funding from the Helmholtz Association of German Research Centres. E.P. is supported by BASF SE and the EPSRC Centre for Doctoral Training in Next Generation Synthesis & Reaction Technology (EP/S023232/1). K.R. thanks the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation)—535167833 for financial support. L.M.H. is supported by an Eric and Wendy Schmidt AI in Science Postdoctoral Fellowship, a Schmidt Sciences program.

Data Availability Statement

Accompanying code, notebooks, and figures are published in an accompanying GitHub repository (Siska et al. 2025). The data that support the findings of this study are openly available in BO_Empirical_Examples at https://github.com/lhelleckes/BO_Empirical_Examples.

References

- Abril-Pla, O., V. Andreani, C. Carroll, et al. 2023. “PyMC: A Modern, and Comprehensive Probabilistic Programming Framework in Python.” *PeerJ Computer Science* 9: e1516.
- Agnihotri, A., and N. Batra. 2020. “Exploring Bayesian Optimization.” *Distill* 5, no. 5. <https://doi.org/10.23915/distill.00026>.
- Ament, S., S. Daulton, D. Eriksson, M. Balandat, and E. Bakshy. 2023. “Unexpected Improvements to Expected Improvement for Bayesian Optimization.” arXiv. <https://doi.org/10.48550/arXiv.2310.20708>.
- Balandat, M., B. Karrer, D. Jiang, et al. 2020. “BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization.” In *Advances in Neural Information Processing Systems*, 33, 21524–21538. Curran Associates, Inc.
- Binois, M., and N. Wycoff. 2021. “A Survey on High-Dimensional Gaussian Process Modeling With Application to Bayesian Optimization.” arXiv. <https://doi.org/10.48550/arXiv.2111.05040>.
- Biswas, A., Y. Liu, N. Creange, et al. 2024. “A Dynamic Bayesian Optimized Active Recommender System for Curiosity-Driven Partially Human-in-the-Loop Automated Experiments.” *npj Computational Materials* 10, no. 1: 29.
- Bonilla, E. V., K. Chai, and C. Williams. 2007. “Multi-Task Gaussian Process Prediction.” In *Advances in Neural Information Processing Systems*, Vol. 20, 153–160. Curran Associates, Inc.
- Burt, S. M., T. J. N. Carter, and L. J. Kricka. 1979. “Thermal Characteristics of Microtitre Plates Used in Immunological Assays.” *Journal of Immunological Methods* 31, no. 3: 231–236.
- Caraus, I., B. Mazouze, R. Nadon, and V. Makarenkov. 2017. “Detecting and Removing Multiplicative Spatial Bias in High-Throughput Screening Technologies.” *Bioinformatics* 33, no. 20: 3258–3267.
- Cheon, M., J. H. Lee, D.-Y. Koh, and C. Tsay. 2024. “EARL-BO: Reinforcement Learning for Multi-Step Lookahead, High-Dimensional Bayesian Optimization.” arXiv. <https://doi.org/10.48550/arXiv.2411.00171>.
- Claes, E., T. Heck, K. Coddens, M. Sannaert, J. Schrooten, and J. Verwaeren. 2024. “Bayesian Cell Therapy Process Optimization.” *Biotechnology and Bioengineering* 121, no. 5: 1569–1582.
- Cosenza, Z., R. Astudillo, P. I. Frazier, K. Baar, and D. E. Block. 2022. “Multi-Information Source Bayesian Optimization of Culture Media for Cellular Agriculture.” *Biotechnology and Bioengineering* 119, no. 9: 2447–2458.
- Cosenza, Z., D. E. Block, K. Baar, and X. Chen. 2023. “Multi-Objective Bayesian Algorithm Automatically Discovers Low-Cost High-Growth Serum-Free Media for Cellular Agriculture Application.” *Engineering in Life Sciences* 23, no. 8: e2300005.
- Coutinho, J. P., I. Castillo, and M. S. Reis. 2024. “Human-in-the-Loop Controller Tuning Using Preferential Bayesian Optimization.” *IFAC-PapersOnLine* 58, no. 14: 13–18.
- Dai, Z., Q. P. Nguyen, S. S. Tay, et al. 2023. “Batch Bayesian Optimization for Replicable Experimental Design.” arXiv. <https://doi.org/10.48550/arXiv.2311.01195>.
- Daulton, S., X. Wan, D. Eriksson, M. Balandat, M. A. Osborne, and E. Bakshy. 2022. “Bayesian Optimization Over Discrete and Mixed Spaces via Probabilistic Reparameterization.” arXiv. <https://doi.org/10.48550/arXiv.2210.10199>.
- De Luca, R., G. Costa, H. Narayanan, et al. 2023. “Comparison of Strategies for Iterative Model-Based Upstream Bioprocess Development With Single and Parallel Reactor Set-Ups.” *Biochemical Engineering Journal* 191: 108813.
- Deshwal, A., C. M. Simon, and J. R. Doppa. 2021. “Bayesian Optimization of Nanoporous Materials.” *Molecular Systems Design & Engineering* 6, no. 12: 1066–1086.
- Diwale, S., M. K. Eisner, C. Carpenter, W. Sun, G. C. Rutledge, and R. D. Braatz. 2022. “Bayesian Optimization for Material Discovery Processes With Noise.” *Molecular Systems Design & Engineering* 7, no. 6: 622–636.
- Dunie, M. 2017. “The Importance of Research Data Management: The Value of Electronic Laboratory Notebooks in the Management of Data Integrity and Data Availability.” *Information Services & Use* 37, no. 3: 355–359.
- Dürholt, J. P., T. S. Asche, J. Kleinekorte, et al. 2024. “BoFire: Bayesian Optimization Framework Intended for Real Experiments.” arXiv. <https://doi.org/10.48550/arXiv.2408.05040>.
- Duvenaud, D. K., D. Maclaurin, J. Iparraguirre, et al. 2015. “Convolutional Networks on Graphs for Learning Molecular Fingerprints.” In *Advances in Neural Information Processing Systems*, Vol. 28, 2224–2232. Curran Associates, Inc.
- Eriksson, D., and M. Jankowiak. 2021. “High-Dimensional Bayesian Optimization With Sparse Axis-Aligned Subspaces.” arXiv. <https://doi.org/10.48550/arXiv.2103.00349>.
- Eriksson, D., M. Pearce, J. R. Gardner, R. Turner, and M. Poloczek. 2019. “Scalable Global Optimization via Local Bayesian Optimization.” arXiv. <https://doi.org/10.48550/arXiv.1910.01739>.
- Fitzner, M., A. Šošić, A. V. Hopp, et al. 2025. “BayBE: A Bayesian Back End for Experimental Planning in the Low-to-no-Data Regime.” *Digital Discovery* 8: 1991–2000.
- Folch, J. P., R. M. Lee, B. Shafei, et al. 2023. “Combining Multi-Fidelity Modelling and Asynchronous Batch Bayesian Optimization.” *Computers & Chemical Engineering* 172: 108194.
- Folch, J. P., C. Tsay, R. M. Lee, et al. 2024. “Transition Constrained Bayesian Optimization via Markov Decision Processes.” arXiv. <https://doi.org/10.48550/arXiv.2402.08406>.
- Forrester, A. I., A. Söbester, and A. J. Keane. 2007. “Multi-Fidelity Optimization via Surrogate Modelling.” *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* 463, no. 2088: 3251–3269.
- Frazier, P. I., and J. Wang. 2016. “Bayesian Optimization for Materials Design.” In *Information Science for Materials Discovery and Design*, edited by T. Lookman, F. J. Alexander, and K. Rajan, Vol. 225, 45–75. Springer International Publishing.
- Freier, L., J. Hemmerich, K. Schöler, W. Wiechert, M. Oldiges, and E. Von Lieres. 2016. “Framework for Kriging-Based Iterative Experimental Analysis and Design: Optimization of Secretory Protein

- Production in *Corynebacterium glutamicum*." *Engineering in Life Sciences* 16, no. 6: 538–549.
- Freier, L., and E. Von Lieres. 2017. "Multi-Objective Global Optimization (MOGO): Algorithm and Case Study in Gradient Elution Chromatography." *Biotechnology Journal* 12, no. 7: 1600613.
- Gardner, J. R., G. Pleiss, D. Bindel, K. Q. Weinberger, and A. G. Wilson. 2018. "GPYtorch: Blackbox Matrix-Matrix Gaussian Process Inference With GPU Acceleration." arXiv. <https://doi.org/10.48550/arXiv.1809.11165>.
- Garnett, R. 2023. *Bayesian Optimization*, 1st ed. Cambridge University Press.
- Garrido-Merchán, E. C., and D. Hernández-Lobato. 2020. "Dealing With Categorical and Integer-Valued Variables in Bayesian Optimization With Gaussian Processes." *Neurocomputing* 380: 20–35.
- Georgiou, A., D. Jungen, L. Kaven, et al. 2025. "Deterministic Global Optimization of the Acquisition Function in Bayesian Optimization: To Do or Not To Do?" arXiv. <https://doi.org/10.48550/arXiv.2503.03625>.
- Gisberg, F., R. Klausser, M. Elshazly, J. Kopp, E. P. Brichtová, and O. Spadiut. 2025. "Bayesian Optimization in Bioprocess Engineering—Where Do We Stand Today?" *Biotechnology and Bioengineering* 122, no. 6: 1313–1325.
- Goh, W. W. B., W. Wang, and L. Wong. 2017. "Why Batch Effects Matter in Omics Data, and How to Avoid Them." *Trends in Biotechnology* 35, no. 6: 498–507.
- Gómez-Bombarelli, R., J. N. Wei, D. Duvenaud, et al. 2018. "Automatic Chemical Design Using a Data-Driven Continuous Representation of Molecules." *ACS Central Science* 4, no. 2: 268–276.
- Gonzalez, J., Z. Dai, A. Damianou, and N. D. Lawrence. 2017. "Preferential Bayesian Optimization." arXiv. <https://doi.org/10.48550/arXiv.1704.03651>.
- Gonzalez, J., Z. Dai, P. Hennig, and N. Lawrence. 2016. "Batch Bayesian Optimization via Local Penalization." In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, 648–657. PMLR.
- González, L. D., and V. M. Zavala. 2023. "New Paradigms for Exploiting Parallel Experiments in Bayesian Optimization." *Computers & Chemical Engineering* 170: 108110.
- Görtler, J., R. Kehlbeck, and O. Deussen. 2019. "A Visual Exploration of Gaussian Processes." *Distill* 4, no. 4. <https://doi.org/10.23915/distill.00017>.
- Greenhill, S., S. Rana, S. Gupta, P. Vellanki, and S. Venkatesh. 2020. "Bayesian Optimization for Adaptive Experimental Design: A Review." *IEEE Access* 8: 13937–13948.
- Greif, L., N. Hübschle, A. Kimmig, S. Kreuzwieser, A. Martenne, and J. Ovcharova. 2025. "Structured Sampling Strategies in Bayesian Optimization: Evaluation in Mathematical and Real-World Scenarios." *Journal of Intelligent Manufacturing*: 1–31.
- Griffiths, R.-R., and J. M. Hernández-Lobato. 2020. "Constrained Bayesian Optimization for Automatic Chemical Design Using Variational Autoencoders." *Chemical Science* 11, no. 2: 577–586.
- Han, W., and L. Li. 2022. "Evaluating and Minimizing Batch Effects in Metabolomics." *Mass Spectrometry Reviews* 41, no. 3: 421–442.
- Harrison, R. O., and B. D. Hammock. 1988. "Location Dependent Biases in Automatic 96-Well Microplate Readers." *Journal of Association of Official Analytical Chemists* 71, no. 5: 981–987.
- Head, T., M. Kumar, H. Nahrstaedt, G. Louppe, and I. Shcherbatyi. 2021. *Scikit-Optimize/Scikit-Optimize*. Zenodo.
- Helleckes, L. M., K. Küsters, C. Wagner, et al. 2024. "High-Throughput Screening of Catalytically Active Inclusion Bodies Using Laboratory Automation and Bayesian Optimization." *Microbial Cell Factories* 23, no. 1: 67.
- Helleckes, L. M., C. Müller, T. Griesbach, et al. 2023. "Explore or Exploit? A Model-Based Screening Strategy for PETase Secretion by *Corynebacterium glutamicum*." *Biotechnology and Bioengineering* 120, no. 1: 139–153.
- Helleckes, L. M., M. Osthege, W. Wiechert, E. von Lieres, and M. Oldiges. 2022. "Bayesian Calibration, Process Modeling and Uncertainty Quantification in Biotechnology." *PLOS Computational Biology* 18, no. 3: e1009223.
- Helleckes, L. M., C. Wirmsperger, J. Polak, G. Guillén-Gosálbez, A. Butté, and M. Von Stosch. 2024. "Novel Calibration Design Improves Knowledge Transfer Across Products for the Characterization of Pharmaceutical Bioprocesses." *Biotechnology Journal* 19, no. 7: 2400080.
- Hernandez-Lobato, D., J. Hernandez-Lobato, A. Shah, and R. Adams. 2016. "Predictive Entropy Search for Multi-Objective Bayesian Optimization." In *Proceedings of the 33rd International Conference on Machine Learning*, 1492–1501. PMLR.
- Hernández-Lobato, J. M., M. W. Hoffman, and Z. Ghahramani. 2014. "Predictive Entropy Search for Efficient Global Optimization of Black-Box Functions." arXiv. <https://doi.org/10.48550/arXiv.1406.2541>.
- Hernández Rodríguez, T., A. Sekulic, M. Lange-Hegermann, and B. Frahm. 2022. "Designing Robust Biotechnological Processes Regarding Variabilities Using Multi-Objective Optimization Applied to a Biopharmaceutical Seed Train Design." *Processes* 10, no. 5: 883.
- Hu, R., L. Fu, Y. Chen, J. Chen, Y. Qiao, and T. Si. 2023. "Protein Engineering via Bayesian Optimization-Guided Evolutionary Algorithm and Robotic Experiments." *Briefings in Bioinformatics* 24, no. 1: bbac570.
- Hucka, M., A. Finney, H. M. Sauro, et al. 2003. "The Systems Biology Markup Language (SBML): A Medium for Representation and Exchange of Biochemical Network Models." *Bioinformatics* 19, no. 4: 524–531.
- Hutter, C., M. Von Stosch, M. N. Cruz Bournazou, and A. Butté. 2021. "Knowledge Transfer Across Cell Lines Using Hybrid Gaussian Process Models With Entity Embedding Vectors." *Biotechnology and Bioengineering* 118, no. 11: 4389–4401.
- Hvarfner, C., E. O. Hellsten, and L. Nardi. 2024. "Vanilla Bayesian Optimization Performs Great in High Dimensions." In *Proceedings of the 41st International Conference on Machine Learning*, Proceedings of Machine Learning Research, 20793–20817. PMLR.
- Jäpel, R. C., and J. F. Buyel. 2022. "Bayesian Optimization Using Multiple Directional Objective Functions Allows the Rapid Inverse Fitting of Parameters for Chromatography Simulations." *Journal of Chromatography A* 1679: 463408.
- Jessop-Fabre, M. M., and N. Sonnenschein. 2019. "Improving Reproducibility in Synthetic Biology." *Frontiers in Bioengineering and Biotechnology* 7: 18.
- Jin, Y., and P. V. Kumar. 2023. "Bayesian Optimisation for Efficient Material Discovery: A Mini Review." *Nanoscale* 15, no. 26: 10975–10984.
- Jones, D. R., M. Schonlau, and W. J. Welch. 1998. "Efficient Global Optimization of Expensive Black-Box Functions." *Journal of Global Optimization* 13, no. 4: 455–492.
- Jumper, J., R. Evans, A. Pritzel, et al. 2021. "Highly Accurate Protein Structure Prediction With AlphaFold." *Nature* 596, no. 7873: 583–589.
- Kanda, G. N., T. Tsuzuki, M. Terada, et al. 2022. "Robotic Search for Optimal Cell Culture in Regenerative Medicine." *eLife* 11: e77007.
- Kandasamy, K., A. Krishnamurthy, J. Schneider, and B. Poczos. 2018. "Parallelised Bayesian Optimisation via Thompson Sampling." In *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, 133–142. PMLR.

- Keating, S. M., D. Waltemath, M. König, et al. 2020. "SBML Level 3: An Extensible Format for the Exchange and Reuse of Biological Models." *Molecular Systems Biology* 16, no. 8: e9110.
- Kobalczyk, K., Z. J. Lin, B. Letham, Z. Zhao, M. Balandat, and E. Bakshy. 2025. "LILo: Bayesian Optimization With Interactive Natural Language Feedback." arXiv. <https://doi.org/10.48550/arXiv.2510.17671>.
- Kreyling, J., A. H. Schweiger, M. Bahn, et al. 2018. "To Replicate, or Not to Replicate - That Is the Question: How to Tackle Nonlinear Responses in Ecological Experiments." *Ecology Letters* 21, no. 11: 1629–1638.
- Kricka, L. J., T. J. Carter, S. M. Burt, et al. 1980. "Variability in the Adsorption Properties of Microtitre Plates Used as Solid Supports in Enzyme Immunoassay." *Clinical Chemistry* 26, no. 6: 741–744.
- Krige, D. G. 1951. "A Statistical Approach to Some Mine Valuation and Allied Problems on the Witwatersrand." PhD thesis, University of the Witwatersrand.
- Kushner, H. J. 1962. "A Versatile Stochastic Model of a Function of Unknown and Time Varying Form." *Journal of Mathematical Analysis and Applications* 5, no. 1: 150–167.
- Kushner, H. J. 1964. "A New Method of Locating the Maximum Point of an Arbitrary Multipeak Curve in the Presence of Noise." *Journal of Basic Engineering* 86, no. 1: 97–106.
- Kwon, Y., D. Lee, J. W. Kim, Y.-S. Choi, and S. Kim. 2022. "Exploring Optimal Reaction Conditions Guided by Graph Neural Networks and Bayesian Optimization." *ACS Omega* 7, no. 49: 44939–44950.
- Lapierre, F. M., P. Mattaliano, D. Raith, M. Castillo-Cota, J. Bermeitinger, and R. Huber. 2025. "Multi-Cycle High-Throughput Growth Media Optimization Using Batch Bayesian Optimization." *Journal of Chemical Technology & Biotechnology* 100, no. 8: 1525–1746.
- Lauterbach, S., H. Dienhart, J. Range, et al. 2023. "EnzymeML: Seamless Data Flow and Modeling of Enzymatic Data." *Nature Methods* 20: 400–402.
- Leek, J. T., R. B. Scharpf, H. C. Bravo, et al. 2010. "Tackling the Widespread and Critical Impact of Batch Effects in High-Throughput Data." *Nature Reviews Genetics* 11, no. 10: 733–739.
- Liang, Q., A. E. Gongora, Z. Ren, et al. 2021. "Benchmarking the Performance of Bayesian Optimization Across Multiple Experimental Materials Science Domains." *npj Computational Materials* 7, no. 1: 1–10.
- Liu, T., N. Astorga, N. Seedat, and M. van der Schaar. 2024. "Large Language Models to Enhance Bayesian Optimization." arXiv. <https://doi.org/10.48550/arXiv.2402.03921>.
- Malzacher, S., D. Meißner, J. Range, et al. 2024. "The STRENDA Biocatalysis Guidelines for Cataloguing Metadata." *Nature Catalysis* 7, no. 12: 1245–1249.
- Marchant, R., and F. Ramos. 2012. "Bayesian Optimisation for Intelligent Environmental Monitoring." In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2242–2249. IEEE.
- Martens, A., M. Neufang, A. Butté, M. von Stosch, A. d. R. Chanona, and L. M. Helleckes. 2025. "Holistic Bioprocess Development Across Scales Using Multi-Fidelity Batch Bayesian Optimization." arXiv. <https://doi.org/10.48550/arXiv.2508.10970>.
- Matheron, G. 1963. "Principles of Geostatistics." *Economic Geology* 58, no. 8: 1246–1266.
- Mione, F. M., M. F. Luna, L. Kaspersetz, P. Neubauer, E. C. Martinez, and M. N. Cruz Bournazou. 2025. "A Property Graph Schema for Automated Metadata Capture, Reproducibility and Knowledge Discovery in High-Throughput Bioprocess Development." *Digital Discovery* 4: 2401–2422.
- Moćkus, J. 1975. "On Bayesian Methods for Seeking the Extremum." In *Optimization Techniques IFIP Technical Conference Novosibirsk, July 1–7, 1974*, edited by G. Goos, J. Hartmanis, P. Brinch Hansen, et al., Vol. 27, 400–404. Springer.
- Moćkus, J. 1989. *Bayesian Approach to Global Optimization: Theory and Applications*. Mathematics and Its Applications, Vol. 37. Springer.
- Montgomery, D. C. 2020. *Design and Analysis of Experiments*, 10th ed. Wiley.
- Morschett, H., L. Freier, J. Rohde, W. Wiechert, E. Von Lieres, and M. Oldiges. 2017. "A Framework for Accelerated Phototrophic Bioprocess Development: Integration of Parallelized Microscale Cultivation, Laboratory Automation and Kriging-Assisted Experimental Design." *Biotechnology for Biofuels* 10, no. 1: 26.
- Moss, H., D. Leslie, D. Beck, J. González, and P. Rayson. 2020. "BOSS: Bayesian Optimization Over String Spaces." In *Advances in Neural Information Processing Systems*, Vol. 33, 15476–15486. Curran Associates, Inc.
- Moss, H. B., D. S. Leslie, J. Gonzalez, and P. Rayson. 2021. "GIBBON: General-Purpose Information-Based Bayesian Optimisation." *Journal of Machine Learning Research* 22, no. 235: 1–49.
- Narayanan, H., F. Dingfelder, I. Condado Morales, et al. 2021. "Design of Biopharmaceutical Formulations Accelerated by Machine Learning." *Molecular Pharmaceutics* 18, no. 10: 3843–3853.
- Narayanan, H., J. A. Hinckley, R. Barry, et al. 2025. "Accelerating Cell Culture Media Development Using Bayesian Optimization-Based Iterative Experimental Design." *Nature Communications* 16, no. 1: 6055.
- Nava, E., M. Mutný, and A. Krause. 2022. "Diversified Sampling for Batched Bayesian Optimization With Determinantal Point Processes." arXiv. <https://doi.org/10.48550/arXiv.2110.11665>.
- Negoescu, D. M., P. I. Frazier, and W. B. Powell. 2011. "The Knowledge-Gradient Algorithm for Sequencing Experiments in Drug Discovery." *INFORMS Journal on Computing* 23, no. 3: 346–363.
- Ortmann, L., D. Shi, E. Dassau, F. J. Doyle, B. J. Misgeld, and S. Leonhardt. 2019. "Automated Insulin Delivery for Type 1 Diabetes Mellitus Patients Using Gaussian Process-Based Model Predictive Control." In *2019 American Control Conference (ACC)*, 4118–4123. IEEE.
- Pandi, A., C. Diehl, A. Yazdizadeh Kharrazi, et al. 2022. "A Versatile Active Learning Workflow for Optimization of Genetic and Metabolic Networks." *Nature Communications* 13, no. 1: 3876.
- Pandya, K., C. A. Ray, L. Brunner, J. Wang, J. W. Lee, and B. DeSilva. 2010. "Strategies to Minimize Variability and Bias Associated With Manual Pipetting in Ligand Binding Assays to Assure Data Quality of Protein Therapeutic Quantification." *Journal of Pharmaceutical and Biomedical Analysis* 53, no. 3: 623–630.
- Papenmeier, L., L. Nardi, and M. Poloczek. 2023. "Bounce: Reliable High-Dimensional Bayesian Optimization for Combinatorial and Mixed Spaces." arXiv. <https://doi.org/10.48550/arXiv.2307.00618>.
- Park, S., J. Na, M. Kim, and J. M. Lee. 2018. "Multi-Objective Bayesian Optimization of Chemical Reactor Design Using Computational Fluid Dynamics." *Computers & Chemical Engineering* 119: 25–37.
- Paulson, J. A., G. Makrygiorgos, and A. Mesbah. 2022. "Adversarially Robust Bayesian Optimization for Efficient AUTO-TUNING of Generic Control Structures Under Uncertainty." *AIChE Journal* 68, no. 6: e17591.
- Petsagkourakis, P., B. Chachuat, and E. Antonio Del Rio-Chanona. 2021. "Safe Real-Time Optimization Using Multi-Fidelity Gaussian Processes." In *2021 60th IEEE Conference on Decision and Control (CDC)*, 6734–6741. IEEE.
- Potthoff, J., P. Tremouilhac, P. Hodapp, B. Neumair, S. Bräse, and N. Jung. 2019. "Procedures for Systematic Capture and Management of Analytical Data in Academia." *Analytica Chimica Acta: X* 1: 100007.
- Rahimi, A., and B. Recht. 2007. "Random Features for Large-Scale Kernel Machines." *Advances in Neural Information Processing Systems* 20: 1177–1184.
- Ranković, B., R.-R. Griffiths, and P. Schwaller. 2025. "Large Language Models as Uncertainty-Calibrated Optimizers for Experimental Discovery." arXiv. <https://doi.org/10.48550/arXiv.2504.06265>.

- Rasmussen, C. E., and C. K. I. Williams. 2006. "Gaussian Processes for Machine Learning." in *Adaptive Computation and Machine Learning*. MIT Press.
- Romero, P. A., A. Krause, and F. H. Arnold. 2013. "Navigating the Protein Fitness Landscape With Gaussian Processes." *Proceedings of the National Academy of Sciences* 110, no. 3: E193–E201. <https://doi.org/10.1073/pnas.1215251110>.
- Rosa, S. S., D. Nunes, L. Antunes, D. M. F. Prazeres, M. P. C. Marques, and A. M. Azevedo. 2022. "Maximizing mRNA Vaccine Production With Bayesian Optimization." *Biotechnology and Bioengineering* 119, no. 11: 3127–3139.
- Roselle, C., T. Verch, and M. Shank-Retzlaff. 2016. "Mitigation of Microtiter Plate Positioning Effects Using a Block Randomization Scheme." *Analytical and Bioanalytical Chemistry* 408, no. 15: 3969–3979.
- Ru, B., A. S. Alvi, V. Nguyen, M. A. Osborne, and S. J. Roberts. 2020. "Bayesian Optimisation Over Multiple Continuous and Categorical Inputs." arXiv. <https://doi.org/10.48550/arXiv.1906.08878>.
- Ruberg, S. J., F. Beckers, R. Hemmings, et al. 2023. "Application of Bayesian Approaches in Drug Development: Starting a Virtuous Cycle." *Nature Reviews Drug Discovery* 22, no. 3: 235–250.
- Šaltenis, V. 1971. "On a Method of Multi-Extremal Optimization." *Automatics and Computers (Avtomatika i Vychislatel'naya Tekhnika)* 3: 33–38.
- Sandve, G. K., A. Nekrutenko, J. Taylor, and E. Hovig. 2013. "Ten Simple Rules for Reproducible Computational Research." *PLoS Computational Biology* 9, no. 10: e1003285.
- Savage, T., N. Basha, J. McDonough, J. Krassowski, O. Matar, and E. A. del Rio Chanona. 2024. "Machine Learning-Assisted Discovery of Flow Reactor Designs." *Nature Chemical Engineering* 1, no. 8: 522–531.
- Savage, T., N. Basha, J. McDonough, O. K. Matar, and E. A. del Rio Chanona. 2023. "Multi-Fidelity Data-Driven Design and Analysis of Reactor and Tube Simulations." *Computers & Chemical Engineering* 179: 108410.
- Savage, T., and E. A. del Rio Chanona. 2024. "Human-Algorithm Collaborative Bayesian Optimization for Engineering Systems." *Computers & Chemical Engineering* 189: 108810.
- Schonlau, M., and W. J. Welch. 2006. "Screening the Input Variables to a Computer Model Via Analysis of Variance and Visualization." In *Screening*, edited by A. Dean and S. Lewis, 308–327. Springer-Verlag.
- Shahriari, B., K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas. 2016. "Taking the Human Out of the Loop: A Review of Bayesian Optimization." *Proceedings of the IEEE* 104, no. 1: 148–175.
- Shan, S., and G. G. Wang. 2010. "Survey of Modeling and Optimization Strategies to Solve High-Dimensional Design Problems With Computationally-Expensive Black-Box Functions." *Structural and Multidisciplinary Optimization* 41, no. 2: 219–241.
- Shields, B. J., J. Stevens, J. Li, et al. 2021. "Bayesian Reaction Optimization as a Tool for Chemical Synthesis." *Nature* 590, no. 7844: 89–96.
- Siedentop, R., M. Siska, J. Hermes, S. Lütz, E. von Lieres, and K. Rosenthal. 2025. "Avoiding Replicates in Biocatalysis Experiments: Machine Learning for Enzyme Cascade Optimization." *ChemCatChem* 17, no. 1: e202400777.
- Siska, M., L. Helleckes, and E. Pajak. 2025. Lhelleckes/BO_Empirical_Examples: V0.1.0. Zenodo. <https://doi.org/10.5281/zenodo.16755807>.
- Snoek, J., H. Larochelle, and R. P. Adams. 2012. "Practical Bayesian Optimization of Machine Learning Algorithms." In *Advances in Neural Information Processing Systems*, Vol. 25, 2951–2959. Curran Associates, Inc.
- Stone, K., Y. Xu, X. Li, A. Vikram, M. Christensen, and K. Felton. 2025. *Obsidian: Algorithmic Process Optimization and AI Experiment Design*. Merck & Co. Inc.
- Taylor, C. J., K. C. Felton, D. Wigh, et al. 2023. "Accelerated Chemical Reaction Optimization Using Multi-Task Learning." *ACS Central Science* 9, no. 5: 957–968.
- Thompson, J. C., V. M. Zavala, and O. S. Venturelli. 2023. "Integrating a Tailored Recurrent Neural Network With Bayesian Experimental Design to Optimize Microbial Community Functions." *PLOS Computational Biology* 19, no. 9: e1011436.
- Tipton, K. F., R. N. Armstrong, B. M. Bakker, et al. 2014. "Standards for Reporting Enzyme Data: The STREND Consortium: What It Aims to Do and Why It Should Be Helpful." *Perspectives in Science* 1, no. 1–6: 131–137.
- Ton, J.-F., S. Flaxman, D. Sejdinovic, and S. Bhatt. 2018. "Spatial Mapping With Gaussian Processes and Nonstationary Fourier Features." *Spatial Statistics* 28: 59–78.
- Vincent, A. M., and P. Jidesh. 2023. "An Improved Hyperparameter Optimization Framework for AutoML Systems Using Evolutionary Algorithms." *Scientific Reports* 13, no. 1: 4737. <https://doi.org/10.1038/s41598-023-32027-3>.
- von den Eichen, N., M. Osthege, M. Dölle, et al. 2022. "Control of Parallelized Bioreactors II: Probabilistic Quantification of Carboxylic Acid Reductase Activity for Bioprocess Optimization." *Bioprocess and Biosystems Engineering* 45, no. 12: 1939–1954.
- Wan, X., V. Nguyen, H. Ha, B. Ru, C. Lu, and M. A. Osborne. 2021. "Think Global and Act Local: Bayesian Optimisation Over High-Dimensional Categorical and Mixed Search Spaces." arXiv. <https://doi.org/10.48550/arXiv.2102.07188>.
- Wang, J. 2023. "An Intuitive Tutorial to Gaussian Process Regression." *Computing in Science & Engineering* 25, no. 4: 4–11.
- Wesel, F., and K. Batselier. 2021. "Large-Scale Learning With Fourier Features and Tensor Decompositions." arXiv. <https://doi.org/10.48550/arXiv.2109.01545>.
- Wilkinson, M. D., M. Dumontier, I. J. Aalbersberg, et al. 2016. "The FAIR Guiding Principles for Scientific Data Management and Stewardship." *Scientific Data* 3, no. 1: 160018.
- Wilson, A. G., Z. Hu, R. Salakhutdinov, and E. P. Xing. 2016. "Deep Kernel Learning." In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, 370–378. PMLR.
- Wilson, J. T., F. Hutter, and M. P. Deisenroth. 2018. "Maximizing Acquisition Functions for Bayesian Optimization." arXiv. <https://doi.org/10.48550/arXiv.1805.10196>.
- Yoshida, K., K. Watanabe, T.-Y. Chiou, and M. Konishi. 2023. "High Throughput Optimization of Medium Composition for *Escherichia coli* Protein Expression Using Deep Learning and Bayesian Optimization." *Journal of Bioscience and Bioengineering* 135, no. 2: 127–133.
- Yue, X., and R. A. Kontar. 2020. "Why Non-Myopic Bayesian Optimization Is Promising and How Far Should We Look-Ahead? A Study via Rollout." In *Proceedings of Machine Learning Research*, 2808–2818. MLResearchPress.