# Bayesian Optimisation vs. Input Uncertainty Reduction

JUAN UNGREDDA, MICHAEL PEARCE, and JUERGEN BRANKE, University of Warwick

Simulators often require calibration inputs estimated from real-world data, and the estimate can significantly affect simulation output. Particularly when performing simulation optimisation to find an optimal solution, the uncertainty in the inputs significantly affects the quality of the found solution. One remedy is to search for the solution that has the best performance on average over the uncertain range of inputs yielding an optimal compromise solution. We consider the more general setting where a user may choose between either running simulations or querying an external data source, improving the input estimate and enabling the search for a more targeted, less compromised solution. We explicitly examine the trade-off between simulation and real data collection to find the optimal solution of the simulator with the true inputs. Using a value of information procedure, we propose a novel unified simulation optimisation procedure called *Bayesian Information Collection and Optimisation* that, in each iteration, automatically determines which of the two actions (running simulations or data collection) is more beneficial. We theoretically prove convergence in the infinite budget limit and perform numerical experiments demonstrating that the proposed algorithm is able to automatically determine an appropriate balance between optimisation and data collection.

CCS Concepts: • **Computing methodologies** → **Simulation evaluation**; **Gaussian processes**; *Continuous space search;*

Additional Key Words and Phrases: Input uncertainty, simulation optimisation, Bayesian optimisation

**ACM Reference format:**
Juan Ungredda, Michael Pearce, and Juergen Branke. 2022. Bayesian Optimisation vs. Input Uncertainty Reduction. *ACM Trans. Model. Comput. Simul.* 32, 3, Article 17 (July 2022), 26 pages.
https://doi.org/10.1145/3510380

## 1 INTRODUCTION

Simulation optimisation is the problem of identifying the best solution, when solution qualities can only be estimated via sampling—that is, running a computationally expensive simulation and obtaining a stochastic output value. In many cases, the simulation model has additional parameters that need to be set, such as the mean arrival rate of customers or the mean and variance of the demand distribution.

In reality, such input parameters are either chosen by expert opinion or set to values estimated from historical data. If the chosen values for the input parameters differ significantly from the true

J. Ungredda received funding from EPSRC through grant EP/L015374/1.

Authors' address: J. Ungredda, University of Warwick, Complexity Science Centre, Coventry, CV4 7AL, UK; email: j.ungredda@warwick.ac.uk; M. Pearce, University of Warwick, Mathematics for Real-World Systems, Coventry, CV4 7AL, UK; email: M.A.L.Pearce@warwick.ac.uk; J. Branke, University of Warwick, Warwick Business School, Coventry, CV4 7AL, UK; email: Juergen.Branke@wbs.ac.uk.

1049-3301/2022/07-ART17
https://doi.org/10.1145/3510380

parameters, the solution found by optimising the simulation model may be far from optimal in the real world. This problem is generally known as simulation optimisation with input uncertainty and has received much attention in recent years [Lam et al. 2016]. Much work has focused on explicitly modeling the uncertainty of the input parameters and seeking a robust solution that performs well on average (or worst case) over this distribution.

In this article, we extend our previous work on **Bayesian optimisation (BO)** aiming to identify the solution with the best expected performance given the input uncertainty [Pearce and Branke 2017]. In particular, we assume that the user has access to real-world data that can help to inform the parameters required by the simulator. Given finite resources to spend on simulation and/or data collection, an algorithm must carefully determine which of the two possible actions to perform.

Devoting too much effort to data collection may not leave sufficient resources for optimisation and an algorithm would return a sub-optimal solution to an accurate simulator. However, devoting too little effort to data collection may lead to learning a good *compromise* solution that performs well on average across a variety of possible input parameters but may be sub-optimal under the true input parameters. In this work, we propose a BO algorithm that can intelligently trade off simulation and data collection.

This applies to simulation optimisation problems where additional external input data can be collected incrementally, although requiring resources to collect. For example, sales demand may be estimated from physical sales records that needs to be manually sorted and entered into a database to reduce uncertainty about true demand, or external data may require time-consuming physical measurements by observers such as traffic flow or user choices.

This work makes the following novel contributions.

- We consider a new variant of input uncertainty problems where external data can be sampled to reduce input uncertainty and it is possible to decide how much effort to put into data collection vs. simulation. This differs from previous work where new data acquisition is not an option or data comes from a streaming process and automatically arrives over time without being requested.
- We propose the **Bayesian Information Collection and Optimisation (BICO)** algorithm, which effectively balances the trade-off between simulation and real data collection when the aim is to find the best solution under the true inputs. The algorithm is an extension of the **Value of Information (VoI)** with input uncertainty idea proposed by Pearce and Branke [2017].
- We prove consistency of BICO and show that BICO automatically detects whether an external data source is relevant for optimisation.
- We demonstrate the effectiveness of BICO on several artificial and practical problems.

We start with an overview of related work in Section 2, followed by a formal definition of the problem in Section 3. Section 4 explains the statistical models, derives the suggested sampling procedures, and discusses their theoretical properties and practical computation. We perform numerical experiments in Section 5. Finally, the article concludes with a summary and some suggestions for future work in Section 6.

## 2 LITERATURE REVIEW

Running a simulation model may be computationally expensive. Response surfaces, or metamodels, can be built to predict the output of the computationally expensive simulation much more efficiently. Various metamodeling techniques have been proposed in the literature [Barton and Meckesheimer 2006]. Most relevant for this work are Gaussian processes, which belong to the category of Gaussian random fields (GRF) [Staum 2009; Vanmarcke 2010]. Gaussian processes are non-parametric regression models that provide not only a predicted value for each point in the domain but also a confidence interval.

This is exploited by BO, a global optimisation method. BO builds a Gaussian process, or Kriging, surrogate model of the simulator response surface based on a few initial samples, then uses an acquisition function, or infill criterion, to sequentially decide where to sample next to improve the model and find better solutions. For a brief introduction, refer to the work of Shahriari et al. [2016].

Several BO algorithms have been proposed in the literature. The most popular is the **Efficient Global Optimisation (EGO)** algorithm of Jones et al. [1998] that combines a Gaussian process with an expected improvement criterion for deciding where to sample next. The **Knowledge Gradient (KG)** policy for continuous parameters [Scott et al. 2011] is another myopic acquisition function that aims to maximise the new predicted optimal performance after one new sample. Different from EGO, KG accounts for covariance when judging the value of a sample and can be directly applied to noisy functions.

Conventional simulation optimisation approaches, including BO, assume that the auxiliary input parameters are known, when often this is not the case. Therefore, investigating the effect of input uncertainty has recently gained significant interest in the simulation community (for a general introduction, see, e.g., the work of Lam et al. [2016] and Corlu et al. [2020]). Currently, there are several proposed methods to assess the input uncertainty and its impact on the mean value of the simulation output. Barton and Schruben [2001] build an empirical distribution given historical data and sample from it using direct and bootstrap techniques to assess the impact of input uncertainty. Yi and Xie [2017] further improve by proposing a budget allocation approach that can efficiently employ the simulation resource. Chick [2001] uses a Bayesian posterior distribution to estimate the input distributions for the same purpose. Cheng and Holloand [1997] estimate the simulation output variability by decomposing it into random variations within the simulation model (simulation uncertainty) and input parameter uncertainty. Barton et al. [2014] replace the expensive simulation by metamodel-assisted bootstrapping using a stochastic Kriging response surface to estimate the impact of input uncertainty on the simulation output. Similarly, Yuan and Ng [2020] propose a Gaussian process based Bayesian approach to simultaneously calibrate the simulation model and select the most influential set of uncertain parameters.

The aforementioned methods assume the data to estimate input uncertainty is given. In the case when additional input data can be collected, Freimer and Schruben [2002] examine the question how much data to collect, and for what parameters. They suggest to run an initial experimental design with the endpoints of the confidence interval of the input uncertainty. Then they can use ANOVA to see whether the parameter effects are significant. If they are, then more information should be collected to reduce the uncertainty of the parameter. For a simplified setting only considering main effects, Song and Nelson [2015] propose a more efficient method that approximates the impact of input uncertainty on the overall variance in the simulation output with the help of a mean-variance metamodel depending on the means and variances of the input distributions. Similar work has also considered the trade-off between running more simulation evaluations or to collect more field data to reduce overall output performance uncertainty. Ng and Chick [2006] propose a method based on asymptotic approximations to quantify the impact of data collection. Then, a budget may be allocated in a way that minimises the overall performance uncertainty given sampling allocation constraints. Yuan and Ng [2013], similar to Yuan and Ng [2020], propose a Gaussian process based Bayesian approach to study how to allocate sampling resources by directly comparing the impact of each sampling decision using the integrated mean squared prediction error. Liu and Zhou [2020] suggest a method for quantifying the input uncertainty from streaming data that arrives sequentially over time.

When input uncertainty estimation is considered in the optimisation process, Song et al. [2015] explore the impact of model risk due to input uncertainty on indifference zone (IZ) ranking and selection. Wu and Zhou [2017] use ranking and selection in a two-stage allocation of finite budget, where the first stage consists of estimating the input parameters, followed by the budget allocation scheme to perform simulation runs in the second stage. Xiao and Gao [2018] consider taking the input uncertainty into account, but the optimisation is focused on the worst-case performance given a fixed finite number of input models. Zhou and Xie [2015] propose a formulation that allows to adapt to one's risk preference for the optimisation.

Pearce and Branke [2017] proposed extensions to EGO and KG with continuous parameters to account for input uncertainty, essentially treating optimisation with input uncertainty as optimising an integrated expensive-to-evaluate function. Similar extensions to KG have been proposed by Toscano-Palmerin and Frazier [2018], and to the Informational Approach to Global Optimization (IAGO) algorithm by Wang et al. [2018].

Only very few papers consider the case where additional information can be gathered during the optimisation process. Song and Shanbhag [2019] consider the case of optimisation under input uncertainty when additional data is received from an uncontrolled streaming data process during optimisation. They propose a stochastic approximation framework that prescribes the number of gradient descent steps to be conducted in every timestep. For the discrete ranking and selection problem, Wu and Zhou [2019] study the impact of input uncertainty assuming new data becomes available in each iteration. They propose a technique that discards the oldest simulation outputs in the estimation of the means, an elimination of designs according to its confidence bounds, and a stopping criterion that has a guaranteed probability of correct selection.

In this work, we explicitly look at the trade-off between either running more simulations or instead collecting more input parameter data, all with the aim of finding the optimal solution to a simulator with accurate input parameters. Our methodology mainly builds on preliminary work by Pearce and Branke [2017]. We consider the more general case where the algorithm is allowed to choose between external data and simulation data, and provide a theoretical analysis as well as an empirical evaluation on multiple test problems.

## 3 PROBLEM FORMULATION

For simulation data, we assume *solutions* are given by vectors in a *solution space*, $x \in X \subset \mathbb{R}^D$. The simulator may have multiple inputs for different purposes, and we refer to the concatenated vector as *parameters* in *parameter space*, $a \in A \subset \mathbb{R}^J$. The *simulator* is an arbitrary stochastic black box function we denote as

$$f : X \times A \rightarrow \mathbb{R},$$

which takes as arguments a solution and parameters and returns a noisy scalar valued *performance* $y = \theta(x, a) + \epsilon$, where $\epsilon$ is independent and identically distributed as $N(0, \sigma_\epsilon^2)$. Finally, the expectation of noisy performance is referred to as the *expected output* denoted $\theta(x, a) = \mathbb{E}[f(x, a)]$.

For parameter data collection, we let $N$ be the number of *parameter data sources* indexed by $s \in S = \{1, \ldots, N\}$. Querying a data source $s$ returns a *parameter data point* $r \sim \mathbb{P}[r|a^*, s]$, where $a^*$ is the true parameter vector. Note that $N$ may or may not equal parameter dimension $J$, as one data source may inform several input parameters (e.g., mean and variance in the newsvendor example in the following). If we denote $m$ as the number of data samples collected so far, and $r^i$ the $i$-th value observed from the parameter data source $s^i \in S$, then $\mathscr{R}^m = \{(s, r)^1, \ldots, (s, r)^m\}$ is the set of $m$ queried data pairs. Therefore, $a^*$ may be inferred using the likelihood of the data

$$\prod_{i=1}^{m} \mathbb{P}[r^i | a, s^i]. \tag{1}$$

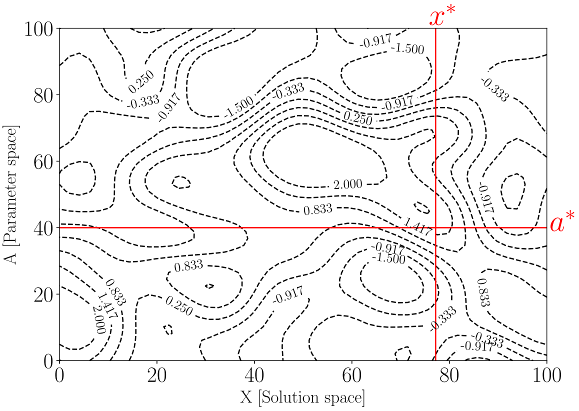


Fig. 1. The expected output $\theta(x, a)$ with true parameter $a^*$ and optimal solution $x^*$. The goal is to learn $x^* = \arg\max \theta(x, a^*)$, which requires learning both the true expected output $\theta(x, a)$ and the true parameter $a^*$ to be able to optimise the objective $\max_x \theta(x, a^*)$.

The likelihood is defined by the application at hand, and therefore we assume it is given and may be used by any algorithm. For the goal of optimisation, both simulation triplets $(x, a, y)$ and parameter data pairs $(s, r)$ must be collected to infer both $\theta(x, a)$ and $a^*$, respectively. The objective function we want to optimise is the expected output given the true input parameter, $\theta(x, a^*)$, and the aim is thus to identify

$$x^* = \arg\max_x \theta(x, a^*).$$

Figure 1 illustrates an example.

There is a budget of $B$ units that can be spent either by choosing $(x, a)$ and calling $f(x, a)$ costing $c_f$, or by choosing $s \in \{1, \dots, N\}$ and querying $\mathbb{P}[r|a^*, s]$ costing $c_s \in \{c_1, \dots, c_N\}$. After consuming the budget, a solution $x_r$ is returned to the user and its quality is determined by the difference in true performance between $x_r$ and the best solution $x^*$, or **opportunity cost (OC)**,

$$OC(x_r) = \theta(x^*, a^*) - \theta(x_r, a^*). \tag{2}$$

For example, in Section 5 we consider the newsvendor problem. A newsvendor aims to maximise profit by choosing the optimal number of newspapers to stock, $x \in \{0, 1, 2, \dots\}$ under uncertain demand $r \sim \text{Normal}(a^*)$ where the true parameter, $a^*$, is composed by the mean demand for newspapers, $\Gamma \in \mathbb{R}^+$, and demand variance, $\sigma_r^2$. Both parameters are unknown and significantly affect the optimal number of newspapers to stock. We have a stochastic simulator to evaluate any chosen stock level with any set demand, $f(x, r)$, which costs $c_f = \$1$ to run. We also have access to $N = 1$ data source, that can be queried for $c_s = \$1$ to obtain demand sales data for a past day, to infer the true parameters $a^* = [\Gamma, \sigma_r^2]$. During optimisation, we have a budget of $B = \$100$ and we can collect either more simulator data, $y = f(x, r)$, or more past sales data, $r$, to find to true optimal stock level, $x^*$, that maximises expected profit for the true demand level, $\max_x \mathbb{E}[f(x, a^*)] = \max_x \theta(x, a^*)$. Note that the costs $c_s$ and $c_f$ describe the relative cost incurred by running additional simulations or collecting additional input data, respectively, and they may correspond to time, money, computational power or other forms of resources required. In some cases, both might be monetary and measured in the same units—for example, if simulations are run on the cloud and input data is collected via a crowdsourcing platform, each would have a known monetary cost. In many other cases, the relative cost has to be set by a domain expert.

## 4 THE BICO ALGORITHM

We propose BICO, which automatically decides whether to conduct additional simulation experiments to find better solutions or to collect additional parameter data to reduce parameter uncertainty. In Sections 4.1 and 4.2, we describe the statistical models for inferring the expected output $\theta(x, a)$ and true parameters $a^*$, respectively. Section 4.3 derives the general VoI procedure, and Sections 4.5 and 4.6 apply this to value collecting simulation and collecting parameter data, respectively. At each iteration, the action is simply determined by what has the highest value. Together, the modelling and automated value based data collection form the BICO algorithm summarised as Algorithm 1 in Section 4.7. We then prove properties about BICO behaviour in Section 4.8.

### 4.1 Statistical Model for the Expected Simulator Output

Let us denote the $n$-th simulation point by $(x, a)^n$, performance by $y^n = f(x^n, a^n)$ and the set of points up to $n$ as $\mathscr{F}^n = \{(x, a, y)^1, \ldots, (x, a, y)^n\}$. For convenience, we define the concatenated simulator arguments $\tilde{X}^n = \{(x, a)^1, \ldots, (x, a)^n\}$ with $\tilde{x} = (x, a)$ and vector of outputs $Y^n = (y^1, \ldots, y^n)$. Then, we propose to use a Gaussian process to model $\theta(x, a)$.

A Gaussian process is defined by a mean function $\mu^0(\tilde{x}) : X \times A \rightarrow \mathbb{R}$ and a covariance function $k^0(\tilde{x}, \tilde{x}') : (X \times A) \times (X \times A) \rightarrow \mathbb{R}$. Given the simulator dataset $\mathscr{F}^n$, predictions of the expected output $\theta(x, a)$ at new locations $(x, a)$ are given by

$$\begin{aligned}\mathbb{E}[\theta(x, a)|\mathscr{F}^n] &= \mu^n(x, a)\\ &= \mu^0(x, a) + k^0((x, a), \tilde{X}^n)(k^0(\tilde{X}^n, \tilde{X}^n) + I\sigma_\epsilon)^{-1}(Y^n - \mu^0(\tilde{X}^n)),\end{aligned} \tag{3}$$

$$\begin{aligned}\mathrm{Cov}\,[\theta(x, a), \theta(x', a')|\mathscr{F}^n] &= k^n((x, a), (x', a'))\\ &= k^0((x, a), (x', a')) - k^0((x, a), \tilde{X}^n)(k^0(\tilde{X}^n, \tilde{X}^n) + I\sigma_\epsilon)^{-1}k^0(\tilde{X}^n, (x', a')).\end{aligned} \tag{4}$$

This model is also referred to as Kriging with a nugget effect [Yin et al. 2011] or stochastic Kriging with (constant, diagonal) intrinsic uncertainty. Although it may be used for the general heteroscedastic noise case (e.g., [Ankenman et al. 2008]), we only consider homogeneous noise in this article. The prior mean $\mu^0(x, a)$ is typically set to $\mu^0(x, a) = 0$ and the $k^0(\tilde{x}, \tilde{x}')$ allows the user to encode known properties of the expected output $\theta(x, a)$ such as smoothness and periodicity. In Section 5, we use the popular squared exponential kernel that assumes $\theta(x, a)$ is a smooth function such that nearby $(x, a)$ have similar outputs while widely separated points have unrelated outputs,

$$k^0((x, a), (x', a')) = \sigma_0^2 \exp\left(\frac{||(x, a) - (x', a')||^2}{2l_{XA}^2}\right), \tag{5}$$

where $\sigma_0 \geq 0$ and $l_{XA} > 0$ are hyper-parameters estimated from the data $\mathscr{F}^n$ by maximum marginal likelihood described in the appendix. Further details can be found in the work of Rasmussen and Williams [2006].

Note that we use a Gaussian process to model the output over the space $X \times A$. Durrande et al. [2012] show that the number of data points required by a Gaussian process increases exponentially with the dimension of the input space. Therefore, this approach may not work for a high number of uncertain parameters or dimensions of the solution space.

### 4.2 Statistical Model for the True Parameters

We use a Bayesian approach to estimate the true parameters $a^*$. The sources $s^1, \ldots, s^m \in S$ are deterministically chosen by the algorithm and the observed $r^1, \ldots, r^m$ are each independently

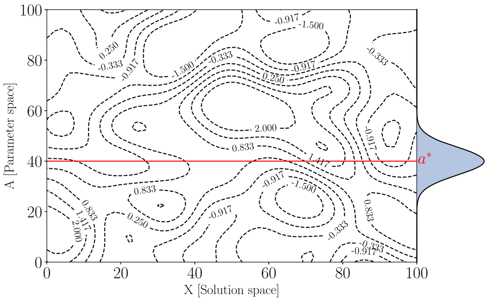


Fig. 2. Surface $\theta(x, a)$ sliced by the unknown true parameter $a^*$ (red) and uncertainty distribution $\mathbb{P}[a|\mathscr{R}^m]$ (blue) over the possible input values given.

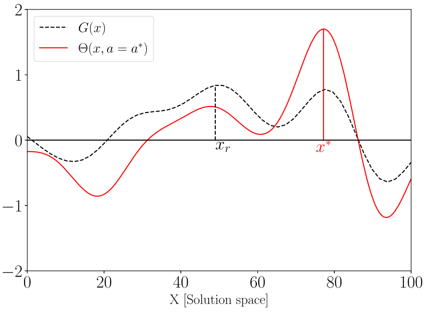


Fig. 3. True expected output defined using true parameter $a^*$ (red) and estimated performance using the parameter distribution $\mathbb{P}[a|\mathscr{R}^m]$.

generated from each corresponding source and have a likelihood given by Equation (1). To supplement data with expert knowledge, we can combine this with a prior distribution $\mathbb{P}[a]$ resulting in a posterior distribution

$$\mathbb{P}[a|\mathscr{R}^m] \propto \mathbb{P}[a] \prod_{i=1}^{m} \mathbb{P}[r^i|a, s^i].$$

By assuming a convenient and intuitive prior distribution, the posterior distribution $\mathbb{P}[a|\mathscr{R}^m]$ can be computed analytically and updated as new sources are queried. In practice, prior information may be modelled based on historical data on the distribution of parameter values, data from experiments done prior to the one being undertaken, or in case of no prior information, a non-informative prior such as Jeffrey's prior can be used. Using conjugate priors facilitates closed-form posterior distributions which increases sampling efficiency.

In this work, we assume a uniform prior $\mathbb{P}[a]$ over the box-constrained space $A$ thereby restricting $a^*$ to realistic values. In our experiments in Section 5, we work with Gaussian distributed data $\mathbb{P}[r|a^*, s]$, therefore the posterior $\mathbb{P}[a|\mathscr{R}^m]$ is a truncated Gaussian which is analytically tractable. Figure 2 shows we can evaluate the true expected output by taking a slice through the surface $\theta(x, a = a^*)$. However, we can only estimate a distribution $\mathbb{P}[a|\mathscr{R}^m]$ through collected data.

### 4.3 Action Space

At any iteration $t = m + n$, the algorithm can choose a simulation point $(x, a) \in X \times A$ and observe $y = f(x, a)$, or it may choose a parameter data source $s \in S = \{1, \dots, N\}$ and observe $r \sim \mathbb{P}[r|a^*, s]$. Therefore, the set of actions available to the algorithm is $\{X \times A, S\}$. In the following, we follow the VoI procedure to derive the expected realised benefit of performing a given action (i.e., an acquisition function over the action set). The algorithm, in each iteration, then selects the action with the largest value.

### 4.4 Predicted Performance

Firstly, we consider the output at the end of executing the algorithm. After exhausting the budget $B$, the algorithm must return a recommended solution $x_r$ to the user. The *true* value of any given solution $x$ is the expected output of the perfect simulator, the objective, $\theta(x, a^*)$. However, both $\theta(x, a)$ and $a^*$ are unknown, hence we need to make two approximations. Firstly, approximate $\theta(x, a)$ with the Gaussian process prediction $\mu^n(x, a)$. Secondly, replace the fixed point $a^*$ with the posterior $\mathbb{P}[a|\mathscr{R}^m]$. Thus, the best estimate of the true quality of solution $x$, $\theta(x, a^*)$, given the data so far $\mathscr{F}^n, \mathscr{R}^m$ is denoted as $G(x;\mathscr{R}^m,\mathscr{F}^n)$ and given by

$$G(x;\mathscr{R}^m,\mathscr{F}^n) = \mathbb{E}_a\left[\mathbb{E}[\theta(x,a)|\mathscr{F}^n]\middle|\mathscr{R}^m\right] = \int_A \mu^n(x,a)\mathbb{P}[a|\mathscr{R}^m]da. \tag{6}$$

Then, the best solution to recommend, $x_r$, is the solution that maximises the model's current prediction of true output

$$x_r(\mathscr{R}^m,\mathscr{F}^n) = \arg\max_x G(x;\mathscr{R}^m,\mathscr{F}^n). \tag{7}$$

By using the preceding $x_r$, the corresponding predicted true output is the maximum of $G(\cdot)$ which we denote as

$$G^*(\mathscr{R}^m,\mathscr{F}^n) = \max_x G(x;\mathscr{R}^m,\mathscr{F}^n). \tag{8}$$

We use $G^*(\mathscr{R}^m,\mathscr{F}^n)$ as the measure of value or quality of the data we currently have. A VoI procedure quantifies the value of an action by computing the one-step look-ahead future expectation of this value and performing the action with maximum value.

The difference between using the true parameter $a^*$ and the parameter distribution can be seen in Figure 3. The predicted solution quality $G(x)$ with the recommended solution $x_r$ and true quality $\theta(x,a^*)$ with true best solution $x^*$ may differ substantially. Simulation data helps to improve $\mu^n(x,a)$ to converge towards $\theta(x,a)$. However, even with full simulator information, $\mu^n(x,a) = \theta(x,a)$, the predicted output $G(x)$ must marginalise over $a$ by Equation (6) which is still imperfect and $x_r \neq x^*$.

We next derive the VoI of performing any action. This is computed by assuming an action is taken and considering the hypothetical predicted performance at the next timestep, either $G^*(\mathscr{R}^{m+1},\mathscr{F}^n)$ or $G^*(\mathscr{R}^m,\mathscr{F}^{n+1})$.

### 4.5 VoI for Simulation Data

If a simulation point $(x,a,y)^{n+1}$ were to be collected thereby augmenting $\mathscr{F}^{n+1} = \mathscr{F}^n \cup \{(x,a,y)^{n+1}\}$, then the updated predicted performance would be $G(\mathscr{R}^m,\mathscr{F}^{n+1})$. At time $t = m+n$, given the next simulation point $(x,a)^{n+1}$ and before collecting the new performance $y^{n+1}$, we may compute the one-step look-ahead incremental increase in predicted performance which is the VoI of taking the action $(x,a)^{n+1}$,

$$\mathrm{VoI}((x,a);\mathscr{R}^m,\mathscr{F}^n) = \mathbb{E}_{y^{n+1}}\left[\frac{G^*(\mathscr{R}^m,\mathscr{F}^{n+1}) - G^*(\mathscr{R}^m,\mathscr{F}^n)}{c_f}\middle|(x,a)^{n+1} = (x,a),\mathscr{F}^n\right], \tag{9}$$

where $c_f$ is the cost of running a simulation. Assuming the datasets are given, $\mathrm{VoI}((x,a)^{n+1};\cdot) : X \times A \to \mathbb{R}$ is a scalar valued function over the domain of the simulator. It returns the expected increase in the peak of the predicted simulator output, $G(x;\mathscr{R}^m,\mathscr{F}^n)$, per unit cost of running the simulator.

To evaluate $\mathrm{VoI}((x,a)^{n+1};)$, we next derive the predictive distribution of $G(x;\mathscr{R}^m,\mathscr{F}^{n+1})$ given data at time $t = n+m$. This requires an updating formula for the posterior mean $\mu^{n+1}(x,a)$. By setting the posterior mean and covariance after $n$ samples, $\mu^n(x,a), k^n((x,a),(x',a'))$, as the prior mean and covariance in Equation (3), we can write the formula for the mean after the $(n+1)$-th sample as

$$\mu^{n+1}(x,a) = \mu^n(x,a) + \frac{k^n((x,a),(x,a)^{n+1})}{k^n((x,a)^{n+1},(x,a)^{n+1}) + \sigma_\epsilon^2}(y^{n+1} - \mu^n(x,a)), \tag{10}$$

where $(x,a)^{n+1}$ is a given argument to $\text{VoI}(\cdot)$ and $y^{n+1}$ is unknown. The Gaussian process model provides a predictive distribution for the new function value

$$y^{n+1} \sim N(\mu^n(x,a)^{n+1}, k^n((x,a)^{n+1},(x,a)^{n+1}) + \sigma_\epsilon^2). \tag{11}$$

By writing $y^{n+1} = \mu^n(x,a) + \sqrt{k^n((x,a)^{n+1},(x,a)^{n+1}) + \sigma_\epsilon^2}Z$ with $Z \sim N(0,1)$, substituting into Equation (10) and simplifying leads to the following parametrisation of $\mu^{n+1}(x,a)$,

$$\mu^{n+1}(x,a) = \mu^n(x,a) + \tilde{\sigma}^n((x,a),(x,a)^{n+1})Z, \tag{12}$$

where $\tilde{\sigma}^n((x,a),(x,a)^{n+1})$ is a deterministic function parametrised by $(x,a)^{n+1}$ that is the additive update to the posterior mean scaled by $Z$

$$\tilde{\sigma}^n((x,a),(x,a)^{n+1}) = \frac{k^n((x,a),(x,a)^{n+1})}{\sqrt{k^n((x,a)^{n+1},(x,a)^{n+1}) + \sigma_\epsilon^2}}. \tag{13}$$

Therefore, the predictive distribution of the new posterior mean is given by

$$\mu^{n+1}(x,a) \sim N(\mu^n(x,a), \tilde{\sigma}^n((x,a),(x,a)^{n+1})^2), \tag{14}$$

and the predicted performance after a new sample $(x,a)^{n+1}$ can then be written as

$$\begin{aligned} G(x;\mathscr{R}^m,\mathscr{F}^{n+1}) &= \int_A \mu^{n+1}(x,a)\mathbb{P}[a|\mathscr{R}^m]da, && (15)\\ &= \int_A \mu^n(x,a)\mathbb{P}[a|\mathscr{R}^m]da + Z\int_A \tilde{\sigma}^n((x,a),(x,a)^{n+1})\mathbb{P}[a|\mathscr{R}^m]da, && (16)\\ &= G(x;\mathscr{R}^m,\mathscr{F}^n) + Z\tilde{\Sigma}^n(x,(x,a)^{n+1}), && (17) \end{aligned}$$

where $\tilde{\Sigma}^n(x,(x,a)^{n+1})$ is the final term in Equation (16). The predictive distribution of a new observation after evaluating $(x,a)^{n+1}$ is then given by

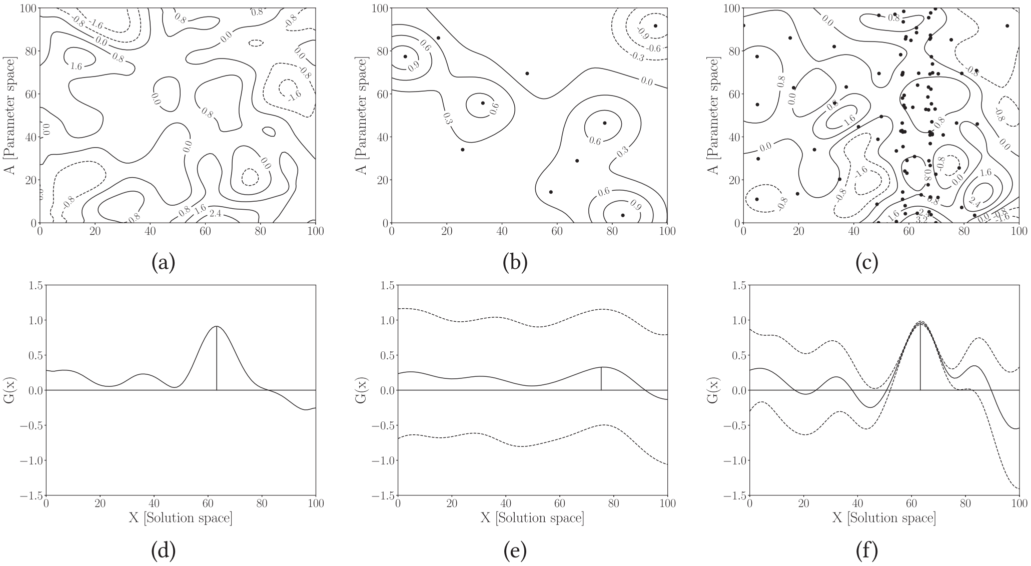
$$G(x;\mathscr{R}^m,\mathscr{F}^{n+1}) \sim N(G(x;\mathscr{R}^m,\mathscr{F}^n), \tilde{\Sigma}^n(x,(x,a)^{n+1})^2). \tag{18}$$

The new sample at $(x,a)^{n+1}$ causes the posterior mean to change at other solutions and inputs according to the additive update $Z\tilde{\Sigma}^n(x,(x,a)^{n+1})$. So, replacing the derived $G(x,\mathscr{R}^m,\mathscr{F}^{n+1})$ (Equation (18)) in the VoI of acquiring a new simulation point $(x,a)$ (Equation (9)) results in

$$\begin{aligned} \text{VoI}((x,a)^{n+1};\mathscr{R}^m,\mathscr{F}^n) &= \frac{1}{c_f}\mathbb{E}_{y^{n+1}}\left[G^*(\mathscr{R}^m,\mathscr{F}^{n+1}) - G^*(\mathscr{R}^m,\mathscr{F}^n)\middle|(x,a)^{n+1}\right], && (19)\\ &= \frac{1}{c_f}\mathbb{E}_Z\left[\max_x\left\{G(x;\mathscr{R}^m,\mathscr{F}^n) + Z\tilde{\Sigma}^n(x,(x,a)^{n+1})\right\} - G^*(\mathscr{R}^m,\mathscr{F}^n)\middle|(x,a)^{n+1}\right]. && (20) \end{aligned}$$

The final expectation is identical to KG under input uncertainty [Pearce and Branke 2017; Toscano-Palmerin and Frazier 2018]. Following these works, the expectation can be evaluated by traditional KG for continuous parameters using Gaussian processes [Frazier et al. 2009] where the maximisation over $x \in X$ embedded within the expectation and within $G^*(\cdot)$ are replaced with a maximisation over a disretised set $x \in X_D \subset X$. With this replacement, the expectation over $Z$ can be evaluated analytically. The $\text{VoI}((x,a);\mathscr{R}^m,\mathscr{F}^n)$ acquisition function may be optimised over the joint solution-input space to find the most beneficial $(x,a)^{n+1}$ and corresponding max $\text{VoI}(\cdot)$.

The preceding VoI contains an integration over $a$, $G(x,\mathscr{R}^m,\mathscr{F}^n)$, which must be approximated through Monte Carlo (special cases can be done analytically, although we do not consider them



(a) (b) (c)

(d) (e) (f)

Fig. 4. In all plots, small points represent function evaluations. (a) Expected output $\theta(x, a)$, and (d) shows $G(x)$ using the expected output $\theta(x, a)$ and uniform parameter distribution. After 10 initial samples, (b) shows the surface $\mu^{10}(x, a)$ and (e) $G(x, \mathscr{R}^0, \mathscr{F}^{10})$. After 90 samples allocated by Equation (20), (c) shows the surface given by $\mu^{100}(x, a)$, and (f) shows $G^{100}(x, \mathscr{R}^0, \mathscr{F}^{100})$.

here). We use $N_A$ sampled parameter values, $a_k \sim \mathbb{P}[a|\mathscr{R}^m]$ for $k \in 1, \ldots, N_A$, and we may write

$$G(x, \mathscr{R}^m, \mathscr{F}^n) \;\approx\; \hat{G}(x, \mathscr{R}^m, \mathscr{F}^n) \;=\; \frac{1}{N_A} \sum_{k=1}^{N_A} \mu^n(x, a_k),$$

with the same expression for $\tilde{\Sigma}^n(x, (x, a)^{n+1})$.

Figure 4 shows KG with input uncertainty [Pearce and Branke 2017]. At the start of sampling, initial samples are allocated by Latin hypercube sampling, the Gaussian process prediction of $\theta(x, a)$ and $G(x; \mathscr{R}^0, \mathscr{F}^{10})$ after the initial allocation are shown in Figure 4(b) and (e) assuming a uniform distribution for $\mathbb{P}[a]$. Then a budget of $B$ samples is allocated sequentially according to Equation (20) (Figure 4(c)). Once all samples have been allocated, based on the learned Gaussian process model, the design $x$ with the largest predicted performance, according to Equation (7), is recommended to the user (Figure 4(f)).

### 4.6 VoI of Data from External Sources

Instead of collecting simulation data, we may collect data from a parameter data source $r^{m+1} \sim \mathbb{P}[r|a^*, s^{m+1}]$ thereby augmenting the corresponding dataset $\mathscr{R}^{m+1} = \mathscr{R}^m \cup \{(s, r)^{m+1}\}$. This also generates an improvement in predicted performance that we refer to as the VoI of collecting additional external data, given by

$$\mathrm{VoI}(s; \mathscr{R}^m, \mathscr{F}^n) \quad = \quad \mathbb{E}_{r^{m+1}} \left[ \frac{G^*(\mathscr{R}^{m+1}, \mathscr{F}^n) - G^*(\mathscr{R}^m, \mathscr{F}^n)}{c_s} \Big| s^{m+1} = s, \mathscr{R}^m \right], \tag{21}$$

with $c_s$ being the cost of sampling external data source $s$. $\mathrm{VoI}(s; \mathscr{R}^m, \mathscr{F}^n)$ is computed using Monte Carlo with samples $r^{m+1}$ generated from to the predictive density $\mathbb{P}[r^{m+1}|s^{m+1}, \mathscr{R}^m]$. Each sample leads to a realisation of $G^*(\mathscr{R}^{m+1}, \mathscr{F}^n)$. The difference between the average of future

$G^*(\mathscr{R}^{m+1}, \mathscr{F}^n)$ realisations and the current $G^*(\mathscr{R}^m, \mathscr{F}^n)$ yields a non-negative quantity that can be used to assess the benefit of sampling a parameter data source $s$.

However, naively using Monte Carlo integration can lead to high variance and expensive computation. We instead recommend to use importance sampling, common random numbers and control variates as follows.

We use the same Monte Carlo set $A_{MC} = \{a_1, \ldots, a_{N_A}\}$ as used to compute the benefit of simulation data $\mathrm{VoI}((x,a);\cdot)$, thereby the two types of actions, simulator $(x,a)$ and external data $s$, are compared with common random numbers. Given a hypothetical $(s,r)^{m+1}$ (where $s^{m+1}$ is chosen and $r^{m+1}$ is a sample), the one-step look-ahead expected performance function $G(x, \mathscr{R}^{m+1}, \mathscr{F}^n)$ may be estimated using $A_{MC}$ with importance sampling

$$G(x, \mathscr{R}^{m+1}, \mathscr{F}^n) \approx \hat{G}(x, \mathscr{R}^{m+1}, \mathscr{F}^n) = \frac{1}{N_A} \sum_{k=1}^{N_A} \mu^n(x, a_k) \frac{\mathbb{P}[a_k|\mathscr{R}^{m+1}]}{\mathbb{P}[a_k|\mathscr{R}^m]},$$

and the corresponding $\hat{G}^*(\mathscr{R}^{m+1}, \mathscr{F}^n)$ is found by a Nelder-Mead optimiser over $x \in X$. Thus, to evaluate $\mathrm{VoI}(s;\cdot)$, given a source $s^{m+1} = s$, we generate $N_r$ samples $r_l^{m+1} \sim \mathbb{P}[r^{m+1}|s^{m+1}; \mathscr{R}^m]$ for $l \in \{1, \ldots N_r\}$. Each sample $r_l$ produces a realisation of a future dataset, $\mathscr{R}_l^{m+1}$, a future posterior density, $\mathbb{P}[a|\mathscr{R}_l^{m+1}]$, and importance weights, $\frac{\mathbb{P}[a_k|\mathscr{R}_l^{m+1}]}{\mathbb{P}[a_k|\mathscr{R}^m]}$. The weights define a function $\hat{G}(x, \mathscr{R}_l^{m+1}, \mathscr{F}^n)$ with peak $\hat{G}^*(\mathscr{R}_l^{m+1}, \mathscr{F}^n)$. The VoI of collecting external data from source $s$ is the average of new peaks minus the current peak

$$\mathrm{VoI}(s; \mathscr{R}^m, \mathscr{F}^n) = \frac{1}{c_s}\left(\left(\frac{1}{N_R}\sum_{l=1}^{N_R} \hat{G}^*(\mathscr{R}_l^{m+1}, \mathscr{F}^n)\right) - \hat{G}^*(\mathscr{R}^m, \mathscr{F}^n)\right). \tag{22}$$

See Figure 5 (centre left, centre right) illustrating the densities and the one-step look-ahead update.

Finally, we may further reduce variance by modifying $\hat{G}^*(\mathscr{R}^m, \mathscr{F}^n)$ to act as a control variate. This is achieved by decomposing the distribution over $a$ as follows:

$$\mathbb{P}[a|\mathscr{R}^m] = \int_{r^{m+1}} \underbrace{\mathbb{P}[a|\mathscr{R}^{m+1}]\mathbb{P}[r^{m+1}|s^{m+1}, \mathscr{R}^m]}_{\mathbb{P}[a, r^{m+1}|s^{m+1}, \mathscr{R}^m]} dr^{m+1}, \tag{23}$$

which may be substituted into the expression for $G^*(\mathscr{R}^m, \mathscr{F}^n)$ and after simplification yields

$$G^*(\mathscr{R}^m, \mathscr{F}^n) = \mathbb{E}_{r^{m+1}}\left[G\left(x_r^{m,n}, \mathscr{R}^{m+1}, \mathscr{F}^n\right)\Big|s^{m+1}, \mathscr{R}^m\right],$$

where the argument $x_r^{m,n} = x_r(\mathscr{R}^m, \mathscr{F}^n)$ is the *current* optimal solution. The final expression for the VoI of external data is therefore

$$\mathrm{VoI}(s; \mathscr{R}^m, \mathscr{F}^n) = \frac{1}{c_s}\frac{1}{N_R}\sum_{l=1}^{N_r}\left[\max_x \hat{G}(x, \mathscr{R}_l^{m+1}, \mathscr{F}^n) - \hat{G}(x_r^{m,n}, \mathscr{R}_l^{m+1}, \mathscr{F}^n)\right].$$

Note that the argument within the summation is strictly non-negative, in contrast with Equation (22) whose summation terms may be positive or negative. See Figure 5 (right) illustrating realisations of the summation argument $\hat{G}(x, \mathscr{R}_l^{m+1}, \mathscr{F}^n) - \hat{G}(x_r^{m,n}, \mathscr{R}_l^{m+1}, \mathscr{F}^n)$ whose peaks are all non-negative.

For the remainder of this work, we will use the shorthand $\mathrm{VoI}^t(\cdot) = \mathrm{VoI}(\cdot; \mathscr{R}^m, \mathscr{F}^n)$ to refer to the VoI at iteration $m+n = t$. We note that extending the method to account for multiple parameter data sources is simply a case of computing $\mathrm{VoI}^t(\cdot)$ for each individual parameter data source $s \in S$ and taking the maximum.

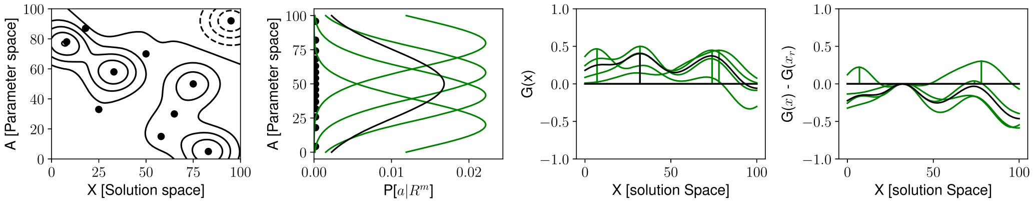


Fig. 5. Left: The simulation metamodel given 10 points. Centre left: Black shows $\mathbb{P}[a|\mathscr{R}^{10}]$, and green shows four realisations of $\mathbb{P}[a|\mathscr{R}_l^{11}]$. Centre right: Black shows $G(x, \mathscr{R}^{10}, \mathscr{F}^{10})$, and green shows the four realisations of $G(x, \mathscr{R}_l^{11}, \mathscr{F}^{10})$. Right: Using the control variate $G(x_r^{10,10}, \mathscr{R}_l^{11}, \mathscr{F}^{10})$ reduces variance and enforces non-negativity.

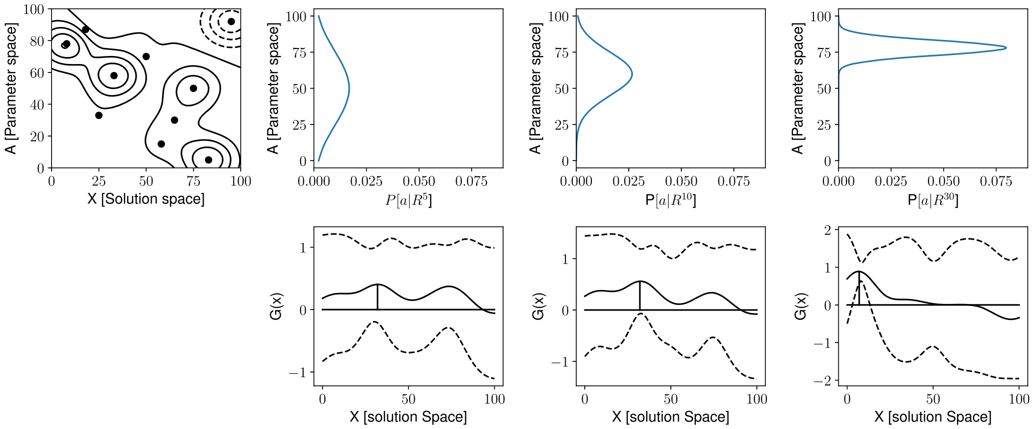
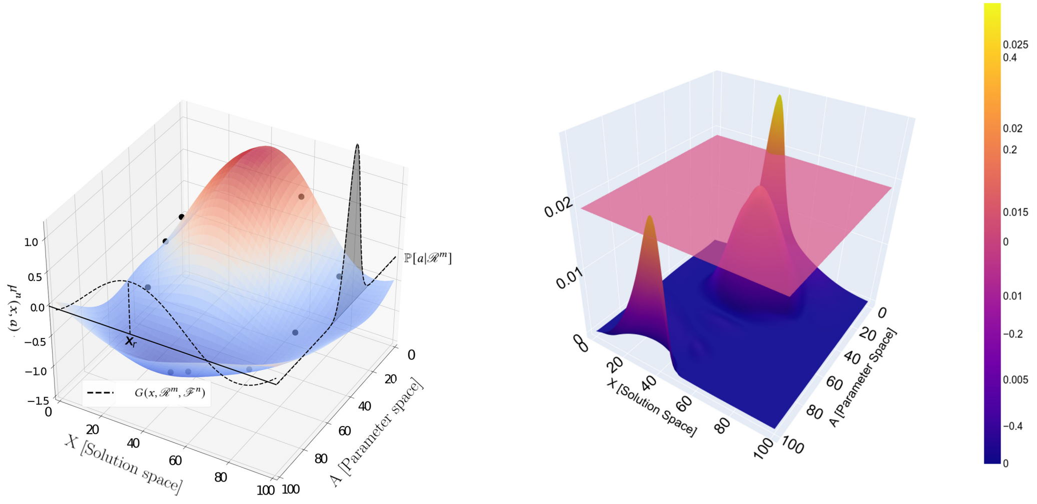


Fig. 6. Iterations of BICO only collecting external data, the simulation metamodel $\mu^n(x, a)$ (left) is unchanged and only the input parameter uncertainty, $\mathbb{P}[a|\mathscr{R}^m]$, is updated. Remaining top plots give $\mathbb{P}[a|\mathscr{R}^5]$, $\mathbb{P}[a|\mathscr{R}^{10}]$ and $\mathbb{P}[a|\mathscr{R}^{30}]$. Bottom plots show the corresponding $G(x, \mathscr{R}^5, \mathscr{F}^{10})$, $G(x, \mathscr{R}^{10}, \mathscr{F}^{10})$, $G(x, \mathscr{R}^{30}, \mathscr{F}^{10})$ which are found by integrating the simulation metamodel over the (vertical) $a \in A$ dimension with the corresponding $\mathbb{P}[a|\cdot]$. Dashed lines represent the confidence interval.

As external input parameter $(s, r)$ pairs are collected, the distribution over input parameter $a$ is updated, in turn updating $G(x, \mathscr{R}^m, \mathscr{F}^n)$. Figure 6 illustrates the model updating with increasing external data.

### 4.7 The Overall Algorithm

BICO is outlined in Algorithm 1. On line 1, the algorithm begins by fitting a Gaussian process model to a set of initial simulation points $\mathscr{F}^n$ using a Latin hypercube (LHS) 'space-filling' experimental design. In addition, we compute the posterior parameter distribution for any collected parameter data source points $\mathscr{R}^m$. After initialisation, the algorithm continues in an optimisation loop until the budget $B$ has been consumed. During each iteration, we compute the VoI of collecting a new simulation point $(x^{n+1}, a^{n+1}, y^{n+1})$ according to $\text{VoI}^t((x, a))$ (line 3) and the VoI of collecting a new sample for each one of the parameter data sources $s \in S$ $\text{VoI}^t(s)$ (line 4). Figure 7(a) shows a Gaussian process simulation metamodel with the input parameter distribution. The corresponding VoI is displayed in Figure 7(b). The multi-modal surface shows the value of collecting a simulation point at $\text{VoI}^t(x, a)$ while the value of collecting external data, $\text{VoI}^t(s)$,



(a) (b)

Fig. 7. (a) The two inferred posterior distributions within BICO. The surface shows the Gaussian process simulation metamodel, $\mu^n(x,a)$, after nine simulation points while the gray bell curve shows the estimated input parameter distribution, $\mathbb{P}[a|\mathscr{R}^m]$ after 10 samples. (b) The VoI used within BICO. The multi-modal surface shows the value of collecting a simulation point at $\text{VoI}^t(x,a)$ while the value of collecting external data, $\text{VoI}^t(s)$, is overlayed as a constant plane. The optimal action is to collect simulation data $y^{n+1} = f(30,0)$.

is overlayed as a constant plane. The action that gives greatest value determines whether we collect a sample $(x,a,y)^{n+1}$ or $r^{m+1}$. In the first case, the Gaussian process model is updated according to the new solution sample (lines 6–9), and for the second case, the posterior parameter distribution is updated according to the new parameter data source sample (lines 11–14). At the end of $B$ samples, the design $x$ with the largest predicted performance $G(x)$ is recommended to the user (line 15). The source code can be downloaded from https://github.com/JuanUngredda/Input-Uncertainty.

## 4.8 Theoretical Properties of BICO

We first outline the proof of asymptotic consistency of BICO with an infinite sampling budget and leave full details to the appendix. We specifically show that if $X$ is discrete and $A \subset \mathbb{R}^d$ is continuous, the BICO algorithm will find the true optimal solution $x^*$ as well as the true parameters $a^*$ in the limit. This builds on a previous proof by Toscano-Palmerin and Frazier [2018] that shows consistency for input uncertainty and collection of simulation points. In the proof, we assume a discrete solution space which simplifies matters significantly. However, we would like to note that the discretisation may be arbitrarily fine.

Proposition 3 shows that if a single action is performed infinitely often, then the value of performing the action vanishes. This implies the value of all actions eventually vanishes.

Proposition 3. *Let $(x',a') \in X \times A$ and suppose that $f(x',a')$ is repeatedly observed, then $\text{VoI}^t((x',a')) \rightarrow 0$ as $t \rightarrow \infty$. Let $s' \in S$ and suppose that $\mathbb{P}[r|a^*,s']$ is repeatedly observed, then $\text{VoI}^t(s') \rightarrow 0$ as $t \rightarrow \infty$.*

Furthermore, if the VoI of all actions is zero, this implies that $x^*$ is known and $\mathbb{P}[a|\mathscr{R}^m]$ is a point-mass distribution on $a^*$.

**ALGORITHM 1:** BICO algorithm. The algorithm starts with an initialization phase to collect preliminary data. Then it proceeds to a sequential phase, dynamically determining to collect simulation data to improve the metamodel or external data to reduce input parameter uncertainty.

```
Input: simulator $f : X \times A \to \mathbb{R}$ with cost $c_f$, external data sources $\mathbb{P}[r|s, a^*] : \{s_1, \dots, s_N\} \to r$ with costs
       $\{c_1, \dots, c_N\}$, external data likelihood functions $\mathbb{P}[r|a, s]$, sampling budget $B$
Collect initial simulation data, $\mathscr{F}^n$, and fit a Gaussian process, $\mu^n(x, a)$
Collect initial external data, $\mathscr{R}^m$, and compute a posterior distribution, $\mathbb{P}[a|\mathscr{R}^m]$
While $b$ < B do:
    Compute $(x, a)^{n+1} = \arg\max_{(x,a)\in X\times A} \mathrm{VoI}^t((x, a))$.
    Compute $s^{m+1} = \arg\max_{s\in S} \mathrm{VoI}^t(s)$
    If $\max_{(x,a)\in X\times A} \mathrm{VoI}^t((x, a)) > \max_{s\in S} \mathrm{VoI}^t(s)$:
        Collect from simulator, $y^{n+1} = f((x, a)^{n+1})$
        Update simulation dataset, $\mathscr{F}^{n+1} \leftarrow \mathscr{F}^n \cup \{(x, a, y)^{n+1}\}$
        Fit a Gaussian process to $\mathscr{F}^{n+1}$
        Update budget consumed, $b \leftarrow b + c_f, n \leftarrow n + 1$
    Else:
        Collect from parameter data source , $r^{m+1} \sim \mathbb{P}[r|a^*, s^{m+1}]$
        Update external dataset, $\mathscr{R}^{m+1} \leftarrow \mathscr{R}^m \cup \{(s, r)^{m+1}\}$
        Compute a posterior distribution $\mathbb{P}[a|\mathscr{R}^{m+1}]$
        Update budget consumed, $b \leftarrow b + c_{s^{m+1}}, m \leftarrow m + 1$
Return: Recommend solution, $x_r = \arg\max_x G(x; \mathscr{R}^m, \mathscr{F}^n)$
```

PROPOSITION 4. *Given a squared exponential kernel with finite length scale, $l_a < \infty$. If VoI($(x, a)$; $\mathscr{R}^m, \mathscr{F}^n$) $= 0$ and VoI($s; \mathscr{R}^m, \mathscr{F}^n$) $= 0$ for all $(x, a)$ and $s$, then* $\arg\max_x G(x; \mathscr{R}^m, \mathscr{F}^\infty) = \arg\max_x \int_A \theta(x, a)\mathbb{P}[a|\mathscr{R}^m]da$ *and* $\mathbb{P}[a|\mathscr{R}^m] = \delta_{a=a^*}$.

We next prove that BICO will not sample from irrelevant information sources. If an input parameter $a$ does not influence the simulator outcome, then an algorithm should not waste budget trying collecting external data to reduce its uncertainty. If we use a squared exponential kernel, then the length scales for each input parameter dimension $a \in A$ determine the relevance of that parameter. A long length scale suggests the model is *less sensitive* to change in such input parameter, hence learning more about such a parameter will have less impact and an algorithm should not sample such data. Remark 1 proves that BICO behaves exactly in this way and will not sample irrelevant external information sources.

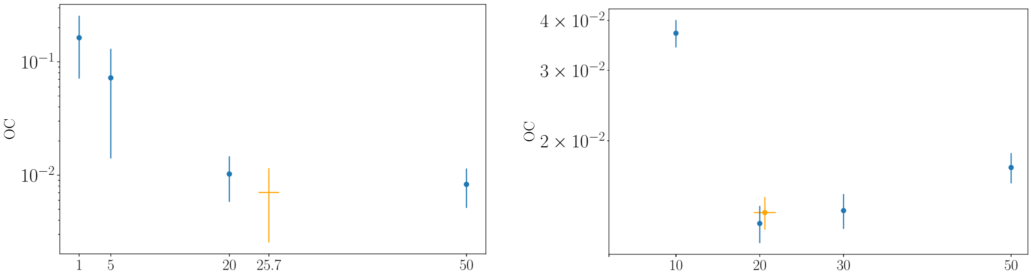
*Remark 1.* Let $(x, a) \in X \times A$ where $A \subset \mathbb{R}$ and assume a single information source $N = 1$. Let the Gaussian process model use a squared exponential kernel,

$$k^0((x, a)(x, a)') = \sigma_f^2 e^{-\frac{1}{2}\left(\frac{(x-x')^2}{l_x} + \frac{(a-a')^2}{l_a}\right)}. \tag{24}$$

Then $\mathrm{VoI}^t(s, \mathscr{R}^m, \mathscr{F}^n) \to 0$ as $l_a \to \infty$ for any $m$ and $n$.

## 5 RESULTS AND DISCUSSION

To demonstrate the performance of BICO, we compare it against a two-stage algorithm which first collects $m$ data source samples to update the input posterior distribution. Then, in stage two the remaining budget is dedicated to sequentially sample from the simulator. If there are two or more input distributions, we equally distribute the initial portion, $m$, over the different inputs. Since in



(a) (b)

Fig. 8. Results for Gaussian process generated test functions. Mean and 95% confidence interval for the OC plotted in a semi-log scale for $B = 100$. (a) With one-dimensional solution space and one parameter. (b) With one-dimensional solution space and two parameters. In each experiment, each parameter has one parameter data source. Each confidence interval is generated using 20 replications.

practice it is not possible to know in advance how many data source samples $m$ should be collected, we will show results for different settings of $m$. The best setting of $m$ can then be regarded as an ideal case which cannot be obtained for real problems.

For simplicity and without loss of generality, in all figures, we consider the same acquisition cost, $c_s = c_f = 1$. In practice, the only requirement is that $c_s$ and $c_f$ are comparable measures of 'cost' for data collection. They are application specific and often depend on the user's preferences. Results for BICO are shown in orange, with OC error bars (see Equation (2)) as vertical lines and the average number of samples and its error bar as horizontal lines.

### 5.1 Gaussian Process Generated Experiments

We consider a test function with solution space $X = [0, 100]$ and either one parameter in $A = [0, 100]$ or two parameters with $A = [0, 100]^2$ generated from a Gaussian process with a squared exponential kernel with known hyper-parameters $l_{XA} = 10$, $\sigma_0^2 = 1$ , $\sigma_\epsilon^2 = (0.1)^2$. The total budget in both cases was set to $B = 100$. To model input uncertainty, we assume a uniform prior $\mathbb{P}[a] = \frac{1}{100}$ and normally distributed data source samples for each source with unknown mean and known variance $\sigma_s^2 = 10$ for all data sources.

Results are shown in semi-log scale in Figure 8, on the left for the case of a single parameter and on the right the case of two parameters with equal variance. The horizontal axis shows the number of samples $m$ allocated to the parameter data source to update $\mathbb{P}[a|\mathscr{R}^m]$, whereas the vertical axis shows the confidence interval of the OC after the budget $B$ has been completely allocated. In both cases, BICO balances the sampling allocation effort in a sensible way, finding comparable results to taking the optimal initial number of parameter data sources samples. Somewhat surprisingly, it seems more effort should be allocated to external data collection if there is only one data source. A possible reason for this is that in case of two data sources, the space over which the simulator is defined is higher, requiring more simulation effort to build a credible Gaussian process model.

Figure 9 shows the case of two parameter data sources with different variances. More specifically, we consider two sources $S = \{1, 2\}$ with $\sigma_{s=1}^2 = 5$ and $\sigma_{s=2}^2 = 10$, where the horizontal and vertical axis show the number of samples allocated to parameter data source 1 and parameter data source 2, respectively. In both cases, for BICO (orange) and the fixed initial allocation (dotted yellow box), the best budget allocation is located around 13 samples for parameter data source 1 and 18 samples for parameter data source 2. Since all length scales are equal for this experiment, differences in the number of samples collected by BICO are mainly due to the difference in variance of the parameter

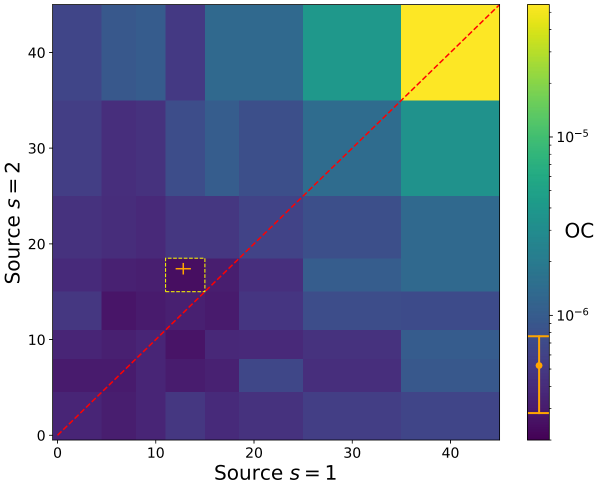


Fig. 9. Grid plot: Number of samples allocated to the parameter data source 1 (horizontal axis) and the parameter data source 2 (vertical axis), where the BICO sample allocation (orange) is represented by a confidence interval for source 1 (horizontal) and source 2 (vertical). Colours inside the grid show the mean OC for the fixed initial proportion approach according to the colour bar for each allocation combination. The best OC for the fixed initial allocation is shown in yellow (dotted line). Colour bar: OC value in a log scale where BICO (orange) is represented as a vertical confidence interval. BICO and each allocation combination are averaged over 20 replications.

data sources, and so as expected, it is best to allocate more samples to the data source with higher variance. The colour bar in Figure 9 compares the confidence interval of the OC for BICO with the benchmark.

### 5.2 Newsvendor Simulation Optimisation

Here, we consider the problem of a newsvendor, who must decide how many copies of the day's paper to stock in the face of uncertain demand and where any unsold copies will be worthless at the end of the day. The solution $x$ is the number of newspapers ordered and demand is a random variable $r \sim \text{Normal}(\Gamma, \sigma_r^2)$ with true mean $\Gamma = 40$ and $\sigma_r^2 = \sqrt{10}$. In this experiment, we assume the solution space  is continuous. Both parameters, expected demand $\Gamma$ and variance $\sigma_r^2$, are unknown in the simulation and must be estimated through real data. The profit, $f(x, r)$, can be calculated as

$$f(x, r) = p \min(x, r) - lx,$$

where $p$ is the selling price and $l$ the production/purchase cost of a newspaper, with $p > l$. For this experiment, we set $p = 5$, $l = 3$ and $x \in [0, 100]$. We use an initial allocation of 10 samples to train the Gaussian process model from an overall budget of $B = 100$. There is only one data source, from which the unknown mean and variance has to be estimated. To update input uncertainty, we assume a non-informative Gamma prior, which is a conjugate prior of the normal distribution so $\mathbb{P}[\Gamma, \sigma_r^2 | \mathscr{R}^m]$ can be sampled hierarchically using $\Gamma | \sigma_r^2, \mathscr{R}^m \sim \text{Normal}(\bar{r}, \sigma_r^2/m)$ and $1/\sigma_r^2 | \mathscr{R}^m \sim \text{Gamma}((m-1)/2, s^2(m-1)/2)$. $\mathbb{P}[r^{m+1} | \mathscr{R}^m]$ is distributed as Student $t$ distribution $t_{m-1}(\bar{r}, s^2(1 + 1/m))$, where $\bar{r}$ and $s^2$ are the sample mean and unbiased sample variance of the collected data $\mathscr{R}^m$.

Figure 10 shows that BICO (orange) again manages to allocate the budget $B$ close to an optimal fixed initial number of samples (blue).

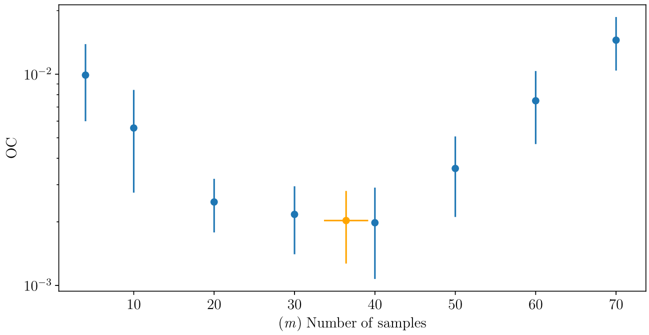


Fig. 10. Newsvendor with unknown mean and variance of demand. Mean and 95% confidence interval of OC where $B = 100$. Blue points: At the start, $m$ input parameter samples are collected to estimate $a^*$, and thereafter standard KG for (fixed) input uncertainty is applied to allocate the remaining budget to simulation points. There is no automatic trade-off between data types; $m$ must be user specified. Orange: BICO algorithm, the horizontal confidence interval showing the range of sample sizes $m$ chosen by BICO. BICO automatically collects an appropriate number of data source samples. Each confidence interval is generated using 25 replications.

### 5.3 Production Line Optimisation

We consider a production line of a manufacturing plant from the Simulation Optimisation Library [Pasupathy and Henderson 2011]. The production line presents three service queues of finite capacity in sequence. Parts leaving after service are immediately transferred to the next queue. In case the next machine is busy, and with full service queue, the previous machine must stop working until the next machine is free since there is no room in the next queue. Parts arrive to queue 1 according to a Poisson process with true rate $\lambda = 0.5$, and the service time for each machine is exponentially distributed with rate $\xi_i, i = 1, 2, 3$. The arrival rate is an unknown parameter in the simulation and must be estimated through real-data acquisition. The decision variables to be optimised are the (continuous) service times of the machines, $\vec{\xi}^T = [\xi_1, \xi_2, \xi_3] \in [0, 2]^3$.

The goal is to maximise the revenue function,

$$R(\vec{\xi}, \lambda) = \frac{10,000\rho(\vec{\xi}, \lambda)}{1 + \vec{c}^T \vec{\xi}} - 400,$$

over a time horizon of $t = 1,000$ timesteps. In the preceding revenue function, $\rho(\vec{\xi}, \lambda)$ denotes the throughput of the production line and is defined as the time-averaged number of parts leaving the last queue, which is a function of the service times and the unknown rate $\lambda$. $\vec{c}^T = [1, 5, 9]$ are cost factors related to the chosen service times of the machines.

To update input uncertainty distribution, we assume a non-informative Gamma prior, which is a conjugate prior of the Gamma distribution. So $\mathbb{P}[\lambda|\mathscr{R}^m]$ can be sampled using a Gamma distribution with shape $1/2 + m$ and rate $\bar{r}m$. $\mathbb{P}[r^{m+1}|\mathscr{R}^m]$ is distributed as a Pareto density with minimum value parameter 0, shape parameter $1/2 + m$ and scale parameter $\bar{r}m$, where $\bar{r}$ is the sample mean of the collected data $\mathscr{R}^m$. Figure 11 shows an adequate allocation for both methods that coincides with around 25% of the overall budget ($B = 100$).

## 6 CONCLUSION

In this article, we proposed a novel unified algorithm for simulation optimisation under input uncertainty. In each iteration, it automatically determines whether to perform more simulation

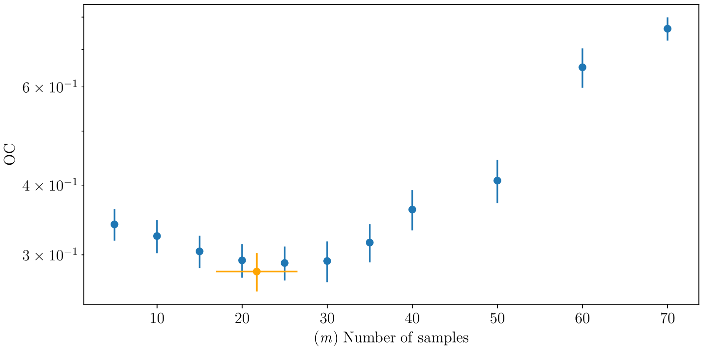


Fig. 11. Production line optimisation experiment plot with mean and 95% confidence interval for the OC plotted in a semi-log scale for $B = 100$. Each confidence interval is generated using 50 replications.

experiments or instead collect more real-world data to reduce the uncertainty about the input parameters. A comparison with an algorithm that allocates a fixed, pre-determined budget $m$ of the available budget to external data collection demonstrated that BICO's allocation mechanism is very powerful and results in a solution performance and fraction of budget allocated to external data collection similar to what can be achieved with the optimal allocation, which is not known in practice.

There are some interesting extensions of this work with concrete practical applications which are possible to pursue. One example is the extension to multi-objective optimisation, where the uncertainty about a user's preferences over objectives can be reduced by querying the user. Further computational efficiency may be attained by transfer learning to carry over information from one stage to the next and speed up the optimisation since the problems solved at every stage are very similar. Lastly, the scalability of BICO with respect to parameter and solution space dimensionality should be investigated.

## A APPENDICES

### A.1 Monte Carlo Convergence of External Data Source

In this section, we present convergence rates for the Monte Carlo approximation to estimate $\text{VoI}(s; \mathscr{R}^m, \mathscr{F}^n)$. Finding an overall error convergence rate, $\phi_{N_R,N_A}$, involves two applications of Monte Carlo integration. Firstly, the inner expectation over $a$ to approximate $G^*(\cdot) \approx \hat{G}^*(\cdot)$ and the outer expectation over $r^{m+1}$ to approximate the next timestep. We define the error as the difference between the (theoretical) truth and the Monte Carlo integrated approximation,

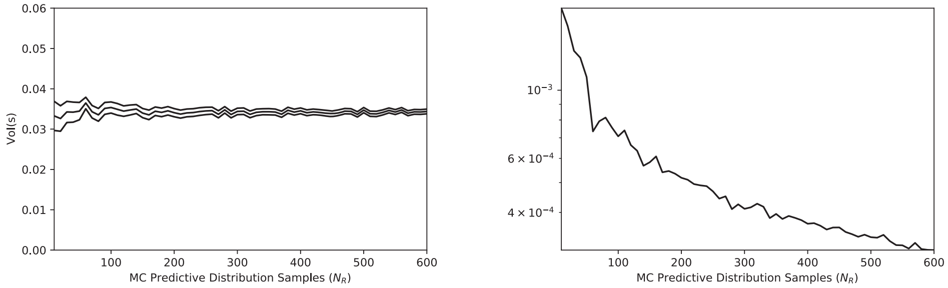
$$\phi_{N_R,N_A} = \mathbb{E}_{r^{m+1}}\left[G^*(\mathscr{R}^{m+1}, \mathscr{F}^n) - G^*(\mathscr{R}^m, \mathscr{F}^n)\Big|s, \mathscr{R}^m\right] - \frac{1}{N_R}\frac{1}{N_A}\sum_{l=1}^{N_R}\left[\sum_{k=1}^{N_A}\left[\mu^n(x_r^{m+1,n}, a_k) - \mu^n(x_r^{m,n}, a_k)\right]\right].$$

We may decompose $\phi_{N_R,N_A}$ into the approximation errors from each Monte Carlo approximation. We may consider the differences from the expression with only one Monte Carlo expression

$$\mathbb{E}_{r^{m+1}}\left[\frac{1}{N_A}\sum_{k=1}^{N_A}\left[\mu^n(x_r^{m+1,n}, a_k) - \mu^n(x_r^{m,n}, a_k)\right]\right].$$

For example, we may write $\phi_{N_A}$ that represents error dependent only on the inner integration over $a$ given by

$$\phi_{N_A} = \mathbb{E}_{r^{m+1}}\left[G^*(\mathscr{R}^{m+1}, \mathscr{F}^n) - G^*(\mathscr{R}^m, \mathscr{F}^n)\Big|s, \mathscr{R}^m\right] - \mathbb{E}_{r^{m+1}}\left[\frac{1}{N_A}\sum_{k=1}^{N_A}\left[\mu^n(x_r^{m+1,n}, a_k) - \mu^n(x_r^{m,n}, a_k)\right]\right].$$



(a) (b)

Fig. 12. At a given iteration of BICO, we fix $N_A$ samples as follows. (a) $\text{VoI}(s; \mathscr{R}^m, \mathscr{F}^n)$ mean and error bars with 95% confidence interval over number of predictive distribution samples $N_R$. (b) The Monte Carlo standard error over number of predictive distribution samples $N_R$.

We may then also write $\phi_{N_R}$ that represents error dependent only on the outer integration over $r^{m+1}$ given by

$$\phi_{N_R} = \mathbb{E}_{r^{m+1}}\left[\frac{1}{N_A}\sum_{k=1}^{N_A}\left[\mu^n(x_r^{m+1,n}, a_k) - \mu^n(x_r^{m,n}, a_k)\right]\right] - \frac{1}{N_R}\frac{1}{N_A}\sum_{l=1}^{N_R}\left[\sum_{k=1}^{N_A}\left[\mu^n(x_r^{m+1,n}, a_k) - \mu^n(x_r^{m,n}, a_k)\right]\right].$$

And finally note that we have $\phi_{N_R,N_A} = \phi_{N_R} + \phi_{N_A}$ as the intermediate term cancels out.

Each sum is composed of independent and identically distributed random samples, thus the error rates for $\phi_{N_A}$ is $O(N_A^{-1/2})$ and $\phi_{N_R}$ is $O(N_R^{-1/2})$, which constitutes an overall error rate $\phi_{N_A,N_R}$ of $O(N_A^{-1/2} + N_R^{-1/2})$. In some cases, $G^*(\cdot)$ may be derived in closed form—that is, when normally distributed $\mathbb{P}[a|\mathscr{R}^m]$ is considered [Le and Branke 2020]—thus $\phi_{N_A} = 0$.

We computed all Monte Carlo estimations with a fixed size $N_A = 150$ and varying $N_R$ on the newsvendor simulation optimisation problem (see Section 5.2). Mean and standard error are presented in Figure 12.

### A.2 Computational Complexity of BICO

We start with the complexity of Gaussian process model fitting, $\mu^n(x, s)$, $k^n((x, a), (x', a'))$. Each iteration of BICO starts by fitting the Gaussian process hyper-parameters by maximising the marginal likelihood. Each marginal likelihood evaluation requires computing the inverse of the covariance matrix $k^0(\tilde{X}^n, \tilde{X}^n) + I\sigma$ costing $O(n^3)$. By caching this inverted matrix, all following posterior mean and variance computations are reduced to $O(n)$ and $O(n^2)$, respectively.

Next we must compute the VoI for simulation data $\text{VoI}^t(x, a)$. This can be explicitly computed using a finite discretisation on the space $X \times A$. We discretise $X$ and $A$ by generating $N_X$ Latin hypercube samples, $X_{MC}$, for $X$ and $N_A$ posterior samples $A_{MC} \sim \mathbb{P}[a|\mathscr{R}^m]$ for $A$. Therefore, the size of the finite discretisation of the product space $X \times A$ is $N_A N_X$. However, we also include the candidate simulation point $(x, a)^{n+1}$ from the optimiser which enlarges the number of mean and covariance computations to $N_A(N_X + 1)$. Since $\tilde{X}_{MC} = X_{MC} \times A_{MC}$ is fixed during the optimisation runs, the posterior mean over these points, $\mu^n(\tilde{X}_{MC})$, may be computed once and stored. Posterior covariance computations may be reduced as follows:

$$k^n(\tilde{X}_{MC}; (x, a)^{n+1}) = k^0(\tilde{X}_{MC}; (x, a)^{n+1}) - k^0(\tilde{X}_{MC}; \tilde{X}^n)(k^0(\tilde{X}^n, \tilde{X}^n) + I\sigma)^{-1} k^0(\tilde{X}^n; (x, a)^{n+1}).$$

Notice that $k^0(\tilde{X}_{MC}; \tilde{X}^n)(k^0(\tilde{X}^n, \tilde{X}^n) + I\sigma)^{-1}$ does not change with the new sample $(x, a)^{n+1}$ and need only be computed once and stored. Therefore, we reduce the mean and covariance computation to $O(N_A N_X n)$ for the points $N_A N_X$. The remaining points $((x^{n+1}, A_{MC}))$ result in a posterior mean and covariance cost computation of $O(N_A n^2)$. We then apply the algorithm developed in the work of Scott et al. [2011] that takes $O((N_X + 1)log(N_X + 1))$ to compute $VoI(x, a)$. Finally, let $K$ be the number of times VoI$(x, a\cdot)$ is called during optimisation of VoI$(x, a\cdot)$. Therefore, the overall optimisation cost results in $O(K(N_A N_X n + N_A n^2 + N_X log(N_X)))$.

Finally, we must compute the VoI of data from external sources VoI$(s; \cdot)$. Let $N_R$ be the number of predictive distribution samples from $\mathbb{P}[r^{m+1} | s^{m+1}; \mathscr{R}^m]$. For each sample we perform continuous optimisation over $x \in X$, each optimisation call requires computing the posterior mean at each $(x, A_{MC})$ locations costing $O(N_A n)$. This computation is repeated for each of the $K$ optimiser iterations for each of the $N_R$ samples for each of the $N$ sources. Thus, the resulting cost is $O(KNN_A N_R n)$.

The computational time of a BICO iteration in the production line optimisation problem (see Section. 5.3) with $n = 50$, $N_A = 200$ and $N_R = 200$ is approximately 50 seconds, from which 20 seconds are dedicated to evaluate VoI(s). Time was measured on a computer with an Intel Core CPU i7-10610U 1.80-GHz processor with 16 GB of RAM and running Ubuntu 20.04.1, and using Python as the programming language.

### A.3 BICO Asymptotic Consistency

In this section, we show that if $X$ is discrete and $A \subset \mathbb{R}^d$ is continuous, then when given an infinite sampling budget $B$, or $t \to \infty$, the BICO algorithm will find the true optimal solution $x^*$ as well as the true parameters $a^*$.

The proof is composed of three parts. Firstly, Proposition 1 shows that the VoI$(\cdot; \mathscr{R}^m, \mathscr{F}^n)$ for any action is non-negative. Secondly, for any single action that is performed infinitely often, the value of performing the action again vanishes $\lim_{t\to\infty}$ VoI$(\cdot; \mathscr{R}^m, \mathscr{F}^n) \to 0$. Together, these imply that any action repeated infinitely often results in that action becoming a *minimum* of the VoI$(\cdot)$ function. As BICO performs the action that is a *maximum* of VoI$(\cdot)$, the value of all actions eventually vanishes. Thirdly, if the value of all actions is zero, this implies that $x^*$ is known.

The first proposition shows that the VoI$(\cdot; \mathscr{R}^m, \mathscr{F}^n)$ is non-negative, meaning that there is always a benefit in collecting more data. These results follow naturally from Jensen's inequality.

Proposition 1. *VoI$^t(\cdot; \mathscr{R}^m, \mathscr{F}^n) \geq 0$ for all $(x, a) \in X \times A$ and all $s \in \{1, \ldots, N\}$.*

Proof of Proposition 1. The proof for both types of action follows from the tower property and Jensen's inequality. We first prove the result for simulation data VoI$^t((x, a)) \geq 0$.

$$\begin{aligned}\mathbb{E}_{y^{n+1}}\left[\max_x G(x; \mathscr{R}^m, \mathscr{F}^{n+1}) \middle| (x, a), \mathscr{F}^n\right] &\geq \max_x \mathbb{E}_{y^{n+1}}\left[G(x; \mathscr{R}^m, \mathscr{F}^{n+1}) \middle| (x, a), \mathscr{F}^n\right] && (25)\\ &= \max_x \mathbb{E}_{y^{n+1}}[\mathbb{E}_a[\mu^{n+1}(x, a)]] && (26)\\ &= \max_x \mathbb{E}_a[\mathbb{E}_{y^{n+1}}[\mu^{n+1}(x, a)]] && (27)\\ &= \max_x \mathbb{E}_a[\mu^n(x, a)] && (28)\\ &= \max_x G(x; \mathscr{F}^n, \mathscr{R}^m) && (29)\end{aligned}$$

Therefore, VoI$^t(x, a; \cdot) = \mathbb{E}_{y^{n+1}}\left[\max_x G(x; \mathscr{R}^m, \mathscr{F}^{n+1}) \middle| (x, a), \mathscr{F}^n\right] - \max_x G(x; \mathscr{F}^n, \mathscr{R}^m) \geq 0$ and there is always some benefit in measuring simulation data. For external data, we may adopt a similar argument to show $\text{VoI}^t(s) \geq 0$.

$$\mathbb{E}_{r^{m+1}}\left[\max_x G(x;\mathscr{R}^{m+1},\mathscr{F}^n)\Big|s,\mathscr{R}^m\right] \geq \max_x \mathbb{E}_{r^{m+1}}\left[G(x;\mathscr{R}^{m+1},\mathscr{F}^n)\Big|s,\mathscr{R}^m\right] \tag{30}$$

$$= \max_x \int_{r^{m+1}} \int_A \mu^n(x,a)\mathbb{P}[a|\mathscr{R}^{m+1}]\mathbb{P}[r^{m+1}|\mathscr{R}^m]da dr^{m+1} \tag{31}$$

$$= \max_x \int_A \mu^n(x,a) \int_{r^{m+1}} \mathbb{P}[a|\mathscr{R}^{m+1}]\mathbb{P}[r^{m+1}|\mathscr{R}^m]dr^{m+1}da \tag{32}$$

$$= \max_x \int_A \mu^n(x,a)\mathbb{P}[a|\mathscr{R}^m]da \tag{33}$$

$$= \max_x G(x;\mathscr{R}^m,\mathscr{F}^n) \tag{34}$$

□

For the second part of the proof, we show that if an action is performed infinitely often, the value of performing the action tends to zero. The information gain in repeating an action decreases and eventually becomes a minimum of the $VoI^t(\cdot)$ function. Again, this is shown in for each type of action.

We first present required preliminary results regarding the limit of collecting infinite simulation data (Proposition 2) or infinite external data (Theorem 1).

Proposition 2. *Let $x, x' \in X$; $a, a' \in A$, and $n \in \mathbb{N}$. The limits of the series $(\mu^n(x,a))$ and $(V^n((x,a),(x',a')))$ (shown in the following) exist.*

$$\mu^n(x,a) = \mathbb{E}_n[f(x,a)] \tag{35}$$

$$V^n((x,a),(x',a')) = \mathbb{E}_n[f(x,a)\cdot f(x',a')] \tag{36}$$

$$= k^n((x,a),(x',a')) + \mu^n(x,a)\cdot\mu^n(x',a') \tag{37}$$

*Denote their limits by $\mu^\infty(x,a)$ and $V^\infty = ((x,a),(x',a'))$ respectively.*

$$\lim_{n\to\infty} \mu^n(x,a) = \mu^\infty(x,a) \tag{38}$$

$$\lim_{n\to\infty} V^n((x,a),(x',a')) = V^\infty((x,a),(x',a')) \tag{39}$$

*If $(x',a')$ is sampled infinitely often, then $\lim_{n\to\infty} V^n((x,a),(x',a')) = \mu^\infty(x,a)\cdot\mu^\infty(x',a')$ holds almost surely.*

Proof of Proposition 2. Cinlar [2011] states in Proposition 2.8 that any sequence of conditional expectations of an integrable random variable under an increasing convex function is a uniformly integrable martingale. Thus, both sequences converge almost surely to their respective limit. If $(x',a')$ is sampled infinitely often, then its posterior variance goes to zero, and $\mathbb{E}_n\left[f(x,a)\cdot f(x',a')\right] \to \mu^\infty(x,a)\cdot\mu^\infty(x',a')$. □

Theorem 1. *If $a$ is defined on a compact set and $C$ is a neighbourhood of $a^*$ with nonzero prior probability, then $\mathbb{P}(a \in C|\mathscr{R}^m) \to 1$ as $m \to \infty$, where $a^*$ is the value of $C$ that minimises the Kullback-Leibler divergence.*

A proof is given in Appendix B in the work of Gelman et al. [2014] and is omitted here for brevity.

Proposition 3. *Let $(x', a') \in X \times A$ and suppose that $f(x', a')$ is repeatedly observed, then $VoI^t((x', a')) \rightarrow 0$ as $t \rightarrow \infty$. Let $s \in S$ and suppose that $\mathbb{P}[r|a^*, s]$ is repeatedly observed, then $VoI^t(s) \rightarrow 0$ as $t \rightarrow \infty$.*

Proof of Proposition 3. First assume $(x', a')$ was observed infinitely often, then consider the Gaussian process one-step look-ahead update

$$\lim_{n\to\infty} \tilde{\Sigma}^n(x; (x', a')) = \lim_{n\to\infty} \int_A \tilde{\sigma}^n((x, a); (x, a)')\mathbb{P}[a|\mathscr{R}^m]da. \tag{40}$$

Since the integral is independent of $n$, we may proceed as follows:

$$\lim_{n\to\infty} \int_A \tilde{\sigma}^n((x, a); (x', a'))\mathbb{P}[a|\mathscr{R}^m]da = \int_A \lim_{n\to\infty} \tilde{\sigma}^n((x, a); (x, a)')\mathbb{P}[a|\mathscr{R}^m]da, \tag{41}$$

$$= \int_A \lim_{n\to\infty} \frac{k^n((x, a); (x, a)')}{\sqrt{k^n((x, a)'; (x, a)') + \sigma_\epsilon^2}}\mathbb{P}[a|\mathscr{R}^m]da, \tag{42}$$

$$= 0. \tag{43}$$

Where the last line is by noting the limit of the denominator $\lim_{n\to\infty} k^n((x, a), (x', a')) = 0$ (see Pearce et al. [2019], Lemma 11), and therefore $\Sigma^\infty(x; (x, a)') = 0$. Thus, we may write as follows:

$$\lim_{n\to\infty} VoI((x, a); \mathscr{F}^n, \mathscr{R}^m) = \frac{\int_{-\infty}^{\infty} \phi(Z) \max_{x''}\{G(x; \mathscr{F}^\infty \mathscr{R}^m) + \overbrace{\Sigma^\infty(x; (x', a'))}^{=0} Z\} - \max_{x''}\{G(x; \mathscr{F}^\infty \mathscr{R}^m)\}}{c_f}, \tag{44}$$

and the integral over $Z$ equates to unity and the denominator terms cancel out. Therefore,

$$\lim_{n\to\infty} VoI((x, a); \mathscr{F}^n, \mathscr{R}^m) = 0.$$

For external data, we rely on Theorem 1 that states that the parameter distribution $\mathbb{P}[a|\mathscr{R}^m]$ converges to $\delta_{a=a^*}$ as $m$ increases.

For the case when $s$ is observed infinitely often, as shown in Theorem. 1, $\mathbb{P}[a|\mathscr{R}^m] \rightarrow \delta_{a=a^*}$ as $m \rightarrow \infty$, therefore

$$G(x; \mathscr{R}^\infty, \mathscr{F}^n) = \int_A \mu^n(x, a)\delta_{a=a*}da, \tag{45}$$

$$= \mu^n(x, a^*). \tag{46}$$

Replacing $G(x; \mathscr{R}^\infty, \mathscr{F}^n)$ in $\mathrm{VoI}(s; \mathscr{R}^\infty, \mathscr{F}^n)$ results in

$$\lim_m \mathrm{VoI}(s; \mathscr{R}^m, \mathscr{F}^n) = \lim_m \mathbb{E}_{r^{m+1}} \left[\max_x G(x; \mathscr{R}^{m+1}, \mathscr{F}^n)\middle| s, \mathscr{R}^m\right] - \max_x G(x; \mathscr{R}^m, \mathscr{F}^n), \tag{47}$$

$$= \mathbb{E}_{r^\infty} \left[\max_x \mu^n(x, a^*)\middle| s, \mathscr{R}^\infty\right] - \max_x \mu^n(x, a^*), \tag{48}$$

$$= \max_x \mu^n(x, a^*) - \max_x \mu^n(x, a^*), \tag{49}$$

$$= 0. \tag{50}$$

□

Proposition 4. *Let $l_a < \infty$. If $VoI((x, a); \mathscr{R}^m, \mathscr{F}^n) = 0$ and $VoI(s; \mathscr{R}^m, \mathscr{F}^n) = 0$ for all $(x, a)$ and $s$, then $\arg\max_{x \in X} G(x; \mathscr{R}^m, \mathscr{F}^\infty) = \arg\max_{x \in X} \int_A \theta(x, a)\mathbb{P}[a|\mathscr{R}^m]da$ and $\mathbb{P}[a|\mathscr{R}^m] = \delta_{a=a^*}$.*

Proof of Proposition 4. Firstly, let us consider the case when $\mathrm{VoI}((x, a); \mathscr{R}^m, \mathscr{F}^n) = 0$ for all $(x, a)$. By Proposition 2, $\lim_{n\to\infty} \tilde{k}^n((x, a), (x, a)') = \tilde{k}^\infty((x, a), (x, a)')$ almost surely for all $x$, $x' \in X$ and $a, a' \in A$. If the posterior variance $\tilde{k}^\infty((x, a), (x, a)) = 0$ for all $(x, a) \in X \times A$, then we know the global optimiser. Now, let us define $(\hat{x},\hat{a}) \in \hat{X} = \{x, a \in X \times A | \tilde{k}^\infty((x,a),(x,a)) > 0)\}$, then

$$\tilde{\Sigma}^\infty(x;(\hat{x},\hat{a})) = \frac{\int_A k^\infty((x,a),(\hat{x},\hat{a}))\mathbb{P}[a|\mathscr{R}^m]da}{\sqrt{k^\infty((\hat{x},\hat{a}),(\hat{x},\hat{a}))+\sigma_\epsilon^2}} > 0.$$

Let us first assume $\tilde{\Sigma}^\infty(x_1;(\hat{x},\hat{a})) \neq \tilde{\Sigma}^\infty(x_2;(\hat{x},\hat{a}))$ for $x_1, x_2 \in X$. Then, $VoI((x,a);\mathscr{R}^m\mathscr{F}^\infty)$ must be strictly positive since for a value of $Z_0 \in Z$, $G(x_1;\mathscr{R}^m\mathscr{F}^\infty) + \tilde{\Sigma}^\infty(x_1;(\hat{x},\hat{a})) > G(x_2;\mathscr{R}^m\mathscr{F}^\infty) + \tilde{\Sigma}^\infty(x_2;(\hat{x},\hat{a}))$ for $Z > Z_0$ and vice versa. Therefore, $\tilde{\Sigma}^\infty(x''';(\hat{x},\hat{a})) = \tilde{\Sigma}^\infty(x'';(\hat{x},\hat{a}))$ must hold for any $x''', x'' \in X$ for $VoI((x,a)) = 0$, which results in

$$\frac{\int_A k^\infty((x''',a),(\hat{x},\hat{a}))\mathbb{P}[a|\mathscr{R}^m]da}{\sqrt{k^\infty((\hat{x},\hat{a}),(\hat{x},\hat{a}))+\sigma_\epsilon^2}} = \frac{\int_A k^\infty((x'',a),(\hat{x},\hat{a}))\mathbb{P}[a|\mathscr{R}^m]da}{\sqrt{k^\infty((\hat{x},\hat{a}),(\hat{x},\hat{a}))+\sigma_\epsilon^2}}.$$

Since $\sigma_\epsilon^2 > 0$,

$$\int_A \left[k^\infty((x''',a),(\hat{x},\hat{a})) - k^\infty((x'',a),(\hat{x},\hat{a}))\right]\mathbb{P}[a|\mathscr{R}^m]da = 0.$$

So $\tilde{\Sigma}^\infty(x;(\hat{x},\hat{a}))$ does not change for all $x \in X$. Moreover, by integrating with respect to $\hat{a}$, as $\tilde{K}(x;\hat{x}) = \int \tilde{\Sigma}^\infty(x'';(\hat{x},\hat{a}))d\hat{a}$ the resulting kernel does not vary with respect to $x$, it must be positive semidefinite, and symmetric. Therefore, by symmetry, the resulting $\tilde{K}(x;\hat{x})$ does not change with respect to $\hat{x}$ and it must follow that the covariance matrix $\tilde{K}(x;\hat{x})$ is proportional to an all-ones matrix and the optimiser is known $argmax_{x\in X}G(x) = argmax_{x\in X}\int_A \theta(x,a)\mathbb{P}[a|\mathscr{R}^m]da$ but not necessarily its true value.

Let us now consider the case when $\mathrm{VoI}(s;\mathscr{R}^m,\mathscr{F}^n) = 0$ for all $s$ as

$$0 = \mathrm{VoI}(s;\mathscr{R}^m,\mathscr{F}^n) = \mathbb{E}_{r^{m+1}}\left[\max_x \int_{a'} \mu^n(x,a')\mathbb{P}[a'|\mathscr{R}^{m+1}]da'\right] - \max_x \int_a \mu^n(x,a)\mathbb{P}[a|\mathscr{R}^m]da. \tag{51}$$

Denote the current recommended solution as $x_r^t = \arg\max_x \int_{a'} \mu^n(x,a)\mathbb{P}[a|\mathscr{R}^m]da$, and the $\mathrm{VoI}^t(s)$ can be rewritten as

$$0 = \mathbb{E}_{r^{m+1}}\left[\max_x \int_{a'} \mu^n(x,a')\mathbb{P}[a'|\mathscr{R}^{m+1}]da'\right] - \int_a \mu^n(x_r^t,a)\mathbb{P}[a|\mathscr{R}^m]da, \tag{52}$$

$$= \mathbb{E}_{r^{m+1}}\left[\max_x \int_{a'} \mu^n(x,a')\mathbb{P}[a'|\mathscr{R}^{m+1}]da' - \int_a \mu^n(x_r^t,a)\mathbb{P}[a|\mathscr{R}^{m+1}]da\right], \tag{53}$$

$$= \mathbb{E}_{r^{m+1}}\left[\max_x \int_{a'} \mu^n(x,a') - \mu^n(x_r^t,a')\mathbb{P}[a'|\mathscr{R}^{m+1}]da'\right]. \tag{54}$$

Note that the random variable within the expectation is non-negative for all $r^{m+1}$. Since the expectation of the non-negative random variable is zero, every realisation of the random variable must be zero, for all $r^{m+1}$

$$\max_x \int_{a'} \mu^n(x,a') - \mu^n(x_r^t,a)\mathbb{P}[a'|\mathscr{R}^{m+1}]da' = 0. \tag{55}$$

If we denote the maximiser (which is a function of $r^{m+1}$) as $x_r^{t+1}(r^{m+1})$, the preceding equality may be written as

$$\int_{a'} \mu^n(x_r^{t+1}(r^{m+1}),a') - \mu^n(x_r^t,a)\mathbb{P}[a'|\mathscr{R}^{m+1}]da' = 0. \tag{56}$$

This equality holds if

$$\mu^n(x_r^{t+1}(r^{m+1}),a') = \mu^n(x_r^t,a) \tag{57}$$

or equivalently $x_r^{t+1}(r^{m+1}) = x_r^t$. Thus, the new maximiser does not depend on $r^{m+1}$.

$$\max_x \int_{a'} \mu^n(x,a')\mathbb{P}[a'|\mathscr{R}^{m+1}]da' \quad = \quad \max_x \int_{a'} \mu^n(x,a')\mathbb{P}[a'|\mathscr{R}^{m}]da' \tag{58}$$

for all $r^{m+1}$ and for all $\mu^n$. The left-hand side also does not depend on $r^{m+1}$ and therefore $\mathbb{P}[a'|\mathscr{R}^{m+1}]$ does not depend on $r^{m+1}$ and we have that $\mathbb{P}[a'|\mathscr{R}^{m+1}] = \mathbb{P}[a'|\mathscr{R}^{m}]$.

Under some regularity conditions, as $m \to \infty$, the posterior distribution of $a$ approaches normality with mean $a^*$ and variance $(mJ(a^*))^{-1}$, where $a^*$ is the value that minimises the Kullback-Leibler divergence and $J$ is the Fisher information. Therefore, if the variance is reduced at a rate of $m^{-1}$, the equality for the posterior distribution at $m$ and $m+1$ is reached only when the posterior distribution is concentrated around the true parameter as $\mathbb{P}[a'|\mathscr{R}^m] = \delta_{a=a^*}$ , where $\delta_{a=\hat{a}}$ is a Dirac delta function on the condition $a = a^*$. □

Therefore, BICO converges to finding the true parameter $a^*$ and true optimal solution $x^*$ as $t$ increases. However, Section 4.3 in the work of Gelman et al. [2004] describes when the regularity conditions may not hold, including underspecified models, non-identified parameters, cases in which the number of parameters grows with the sample size, unbounded likelihood functions, improper posterior distributions, or convergence to a boundary of the parameter space.

### A.4 BICO Relevance Determination

In this section, we show that if we use a squared exponential kernel, then the hyper-parameters determine the relevance of parameter data sources. Therefore, non-relevant parameter data sources will not be sampled by BICO.

*Remark 1.* Assuming a squared exponential kernel,

$$k^0((x,a)(x,a)') = \sigma_f^2 e^{-\frac{1}{2}\left(\frac{(x-x')^2}{l_x} + \frac{(a-a')^2}{l_a}\right)}, \tag{59}$$

and without loss of generality, a parameter $a \in A$, and a solution $x \in X$. Then, $\mathrm{VoI}^t(s, \mathscr{R}^m, \mathscr{F}^n) = 0$ as $l_a \to \infty$ for any $m$ and $n$.

Proof of Remark 1. As $l_a \to \infty$ the posterior mean $\mu^n(x,a)$ only depends on the solution $x$,

$$\lim_{l_a \to \infty} k^0((x,a)(x,a)') = \sigma_f^2 e^{-\frac{1}{2}\left(\frac{(x-x')^2}{l_x}\right)} = k^0(x;x'). \tag{60}$$

Let us denote $\tilde{X}_x^n = \{x^1, \dots, x^n\}$ and assume $\mu^0(x,a) = 0$, then it follows from (60),

$$\begin{aligned}
\mu^n(x,a) &= -k^0((x,a),\tilde{X}^n)(k^0(\tilde{X}^n,\tilde{X}^n) + I\sigma)^{-1}Y^n, &(61)\\
&= -k^0(x,\tilde{X}_x^n)(k^0(\tilde{X}_x^n,\tilde{X}_x^n) + I\sigma)^{-1}Y^n, &(62)\\
&= \mu^n(x). &(63)
\end{aligned}$$

Since $\mu^n$ does not depend on $a$, $G(x;\mathscr{R}^m,\mathscr{F}^n) = \mu^n(x)$, and the $VoI^t(\cdot)$ for $s \in \{1,\dots,N\}$ is

$$\begin{aligned}
\mathrm{VoI}^t(\cdot) &= \mathrm{VoI}(s;\mathscr{R}^m,\mathscr{F}^n), &(64)\\
&= \mathbb{E}_{r^{m+1}}\left[\max_x G(x;\mathscr{R}^{m+1},\mathscr{F}^n)\Big|s,\mathscr{R}^m\right] - \max_x G(x;\mathscr{R}^m,\mathscr{F}^n), &(65)\\
&= \mathbb{E}_{r^{m+1}}\left[\max_x \mu^n(x)\Big|s,\mathscr{R}^m\right] - \max_x \mu^n(x), &(66)\\
&= \max_x \mu^n(x) - \max_x \mu^n(x), &(67)\\
&= 0. &(68)
\end{aligned}$$

□

Therefore, external data is never collected if $a$ is not 'influential' on the predicted simulation output $\mu^n(x, a)$.

## REFERENCES

B. Ankenman, B. L. Nelson, and J. Staum. 2008. Stochastic kriging for simulation metamodeling. In *Proceedings of the Winter Simulation Conference*. IEEE, Los Alamitos, CA, 362–370.

R. Barton and M. Meckesheimer. 2006. Metamodel-based simulation optimization. In *Simulation*, Shane G. Henderson and Barry L. Nelson (Eds.). Handbooks in Operations Research and Management Science, Vol. 13. Elsevier, 535–574.

R. Barton and L. Schruben. 2001. Resampling methods for input modeling. In *Proceedings of the Winter Simulation Conference*. IEEE, Los Alamitos, CA, 372–378.

R. R. Barton, B. L. Nelson, and W. Xie. 2014. Quantifying input uncertainty via simulation confidence intervals. *INFORMS Journal on Computing* 26, 1 (2014), 74–87.

R. C. H. Cheng and W. Holloand. 1997. Sensitivity of computer simulation experiments to errors in input data. *Journal of Statistical Computation and Simulation* 57, 1–4 (1997), 219–241.

S. E. Chick. 2001. Input distribution selection for simulation experiments: Accounting for input uncertainty. *Operations Research* 49, 5 (2001), 744–758.

E. Cinlar. 2011. *Probability and Stochastics*. Graduate Texts in Mathematics, Vol. 261. Springer.

C. G. Corlu, A. Akcay, and W. Xie. 2020. Stochastic simulation under input uncertainty: A review. *Operations Research Perspectives* 7 (2020), 100162.

N. Durrande, D. Ginsbourger, and O. Roustant. 2012. Additive covariance kernels for high-dimensional Gaussian process modeling. *Annales de la Faculté Des Sciences De Toulouse : Mathématiques* Ser. 6, 21, 3 (2012), 481–499.

P. Frazier, W. Powell, and S. Dayanik. 2009. The knowledge-gradient policy for correlated normal beliefs. *INFORMS Journal on Computing* 21, 4 (2009), 599–613.

M. Freimer and L. Schruben. 2002. Simulation input analysis: Collecting data and estimating parameters for input distributions. In *Proceedings of the Winter Simulation Conference*. 393–399.

A. Gelman, J. Carlin, H. Stern, D. Dunson, A. Vehtari, and D. Rubin. 2014. *Bayesian Data Analysis*. CRC Press, Boca Raton, FL.

A. Gelman, J. B. Carlin, H. S. Stern, and D. B. Rubin. 2004. *Bayesian Data Analysis* (2nd ed.). Chapman & Hall/CRC.

M. Jones, D. R. Schonlau and W. J. Welch. 1998. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization* 13, 4 (1998), 455–492.

H. Lam, T. M. K. Roeder, P. I. Frazier, R. Szechtman, E. Zhou, T. Huschka, and S. E. Chick. 2016. Advanced tutorial: Input uncertainty and robust analysis in stochastic simulation. In *Proceedings of the Winter Simulation Conference*. 178–192.

H. Le and J. Branke. 2020. Bayesian optimization searching for robust solutions. In *Proceedings of the Winter Simulation Conference*. IEEE, Los Alamitos, CA, 362–370.

T. Liu and E. Zhou. 2020. Online quantification of input model uncertainty by two-layer importance sampling. arXiv:1912.11172 *(2020)*.

Szu Hui Ng and Stephen E. Chick. 2006. Reducing parameter uncertainty for stochastic systems. *ACM Transactions on Modeling and Computer Simulation* 16, 1 (Jan. 2006), 26–51.

R. Pasupathy and S. G. Henderson. 2011. Simulation Optimization Library. Retrieved February 22, 2022 from http://www.simopt.org.

M. Pearce and J. Branke. 2017. Bayesian simulation optimization with input uncertainty. In *Proceedings of the Winter Simulation Conference*. IEEE, Los Alamitos, CA, 2268–2278.

M. Pearce, M. Poloczek, and Juergen Branke. 2019. Bayesian optimization allowing for common random numbers. *arXiv preprint arXiv:1910.09259* (2019).

C. E. Rasmussen and C. K. I. Williams. 2006. *Gaussian Processes for Machine Learning*. MIT Press, Cambridge, MA.

W. Scott, P. Frazier, and W. Powell. 2011. The correlated knowledge gradient for simulation optimization of continuous parameters using Gaussian process regression. *SIAM Journal on Optimization* 21, 3 (2011), 996–1026.

B. Shahriari, 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, 1 (2016), 148–175.

E. Song and B. L. Nelson. 2015. Quickly assessing contributions to input uncertainty. *IIE Transactions* 47 (2015), 893–909.

E. Song, B. L. Nelson, and L. J. Hong. 2015. Input uncertainty and indifference-zone ranking amp; selection. In *Proceedings of the Winter Simulation Conference*. IEEE, Los Alamitos, CA, 414–424.

E. Song and U. V. Shanbhag. 2019. Stochastic approximation for simulation optimization under input uncertainty with streaming data. In *Proceedings of the Winter Simulation Conference*. IEEE, Los Alamitos, CA, 3597–3608.

J. Staum. 2009. Better simulation metamodeling: The why, what, and how of stochastic kriging. In *Proceedings of the Winter Simulation Conference*. IEEE, Los Alamitos, CA, 119–133.

S. Toscano-Palmerin and P. Frazier. 2018. Bayesian optimization with expensive integrands. *arXiv:1803.08661 [cs.LG]* (2018).

E. Vanmarcke. 2010. *Random Fields.* World Scientific.

H. Wang, J. Yuan, and S. H. Ng. 2018. Informational approach to global optimization with input uncertainty for homoscedastic stochastic simulation. In *Proceedings of the International Conference on Industrial Engineering and Engineering Management.* IEEE, Los Alamitos, CA, 1396–1400.

D. Wu and E. Zhou. 2017. Ranking and selection under input uncertainty: A budget allocation formulation. In *Proceedings of the Winter Simulation Conference.* Article 179, 12 pages.

D. Wu and E. Zhou. 2019. Fixed confidence ranking and selection under input uncertainty. In *Proceedings of the Winter Simulation Conference.* IEEE, Los Alamitos, CA, 3717–3727.

H. Xiao and S. Gao. 2018. Simulation budget allocation for selecting the top-m designs with input uncertainty. *IEEE Transactions on Automatic Control* 63, 9 (2018), 3127–3134.

Y. Yi and W. Xie. 2017. An efficient budget allocation approach for quantifying the impact of input uncertainty in stochastic simulation. *ACM Transactions on Modeling and Computer Simulation* 27, 4 (2017), Article 25, 23 pages.

J. Yin, S. H. Ng, and K. M. Ng. 2011. Kriging metamodel with modified nugget-effect: The heteroscedastic variance case. *Computers and Industrial Engineering* 61, 3 (2011), 760–777.

J. Yuan and S. H. Ng. 2013. A sequential approach for stochastic computer model calibration and prediction. *Reliability Engineering and System Safety* 111 (2013), 273–286.

J. Yuan and S. H. Ng. 2020. An integrated method for simultaneous calibration and parameter selection in computer models. *ACM Transactions on Modeling and Computer Simulation* 30, 1 (2020), Article 7, 23 pages.

E. Zhou and W. Xie. 2015. Simulation optimization when facing input uncertainty. In *Proceedings of the Winter Simulation Conference.* IEEE, Los Alamitos, CA, 3714–3724.

Received May 2020; revised December 2021; accepted January 2022