*Proceedings of the 2025 Winter Simulation Conference*
*E. Azar, A. Djanatliev, A. Harper, C. Kogler, V. Ramamohan, A. Anagnostou, and S. J. E. Taylor, eds.*

# A SIMULATION OPTIMIZATION APPROACH TO OPTIMAL EXPERIMENTAL DESIGN FOR SYMBOLIC DISCOVERY

Kenneth L. Clarkson[1], Soumyadip Ghosh[2], Joao P. Goncalves[2], Rik Sengupta[3],
Mark S. Squillante[2], and Dmitry Zubarev[1]

[1] Mathematics of Computation, IBM Research, Almaden, USA
[2] Mathematics of Computation, IBM Research, Yorktown Heights, USA
[3] Mathematics of Computation, IBM Research, Cambridge, USA

## ABSTRACT

Symbolic discovery aims to discover functional relationships in scientific data gathered from a black-box oracle. In general, the mapping between oracle inputs and its response is constructed by hierarchically composing simple functions. In this study, we restrict ourselves to the case of selecting the most representative model from among a predefined finite set of model classes. The user is given the ability to sequentially generate data by picking inputs to query the oracle. The oracle call expends significant effort (e.g., computationally intensive simulation models), and so each input needs to be carefully chosen to maximize the information in the response. We propose an optimal experimental design formulation to sequentially identify the oracle query inputs and propose a simulation optimization algorithm to solve this problem. We present preliminary results from numerical experiments for a specific symbolic discovery problem in order to illustrate the working of the proposed algorithm.

## 1 INTRODUCTION

The field of symbolic discovery is concerned with finding functional relationships in scientific data that can be expressed by composing together simple functions. The process of selecting a composite of such functions is guided by how well it explains an available collection of data in the form of independent variables and the observed function value. Cornelio et al. (2023) provide a technique to choose among the models by applying background knowledge, to determine if any model can be proven to follow from known principles, or at least be consistent with them. Motivated by that approach, a subset of the authors studied in Clarkson et al. (2022) the model-selection problem from a Bayesian perspective in the setting of *optimal experimental design* (OED).

OED is a well-established discipline that sits at the interface of statistics, simulation and optimization. The goal is to optimize a data acquisition system so that the informational value of the response revealed by an expensive experimental oracle to each query is maximized. OED has seen wide implementation in diverse fields such as neuroscience (Shababo et al. 2013), psychology (Watson 2017), statistical learning (Gal et al. 2017), physics (Melendez et al. 2020), and clinical trials (Cheng and Shen 2005).

OED primarily designs experiments to identify best-fit parameter values for predetermined functional forms. Another variant has the goal of selecting a model out of a fixed set of parameter-free candidates. The general goal of OED for symbolic discovery combines the optimization of data acquisition with identifying the best fit composition of simple functions, each of which has parameters whose values have to be determined.Here we address the simpler yet analogous OED goal of optimizing data acquisition that helps pick the best model form (along with its ideal parameters) from a finite collection of parametric model forms that might represent the oracle. The example studied in Section 4 illustrates how such a finite collection may be obtained from compositions of simple functional forms.

979-8-3315-8726-0/25/$31.00 ©2025 IEEE

### 1.1 Model Discovery

Our general objective is to find the best explanatory representation for a noisy oracle from a finite collection of model classes, where the oracle could be a physical process or a (possibly expensive-to-run) simulation model. Following conventions from the OED literature, we shall refer to the oracle inputs $\mathbf{x} \in \mathscr{X} \subseteq \mathbb{R}^p$ as *design points*, and the output of the oracle $y \in \mathscr{Y} \subseteq \mathbb{R}$ as its *response*. Each of the model classes, which we index using $i \in \mathscr{I}$, allows for an additive noise in capturing the response:

$$Y_i(\mathbf{x}) = m_i(\mathbf{x}, \boldsymbol{\theta}_i) + \sigma_i\, \varepsilon\,, \quad \forall\, i \in \mathscr{I}, \tag{1}$$

where $\varepsilon$ represents a centered unit-variance random variable (r.v.). The parameter fitting problem for a single model of this form has been classically studied in a nonlinear least-squares formulation dating back to Levenberg (1944).

The form $m_i(\mathbf{x}, \boldsymbol{\theta}_i)$ of the mean (or deterministic) response distinguishes the various model classes in $\mathscr{I}$. The parameters $(\boldsymbol{\theta}_i, \sigma_i) \in \mathbb{R}^{d_i} \times \mathbb{R}_+$ and the dimension $d_i$ can differ over the model classes in $\mathscr{I}$. Additional constraints, such as integrality, nonnegativity, and so on, may also be placed on the components of $\boldsymbol{\theta}_i$. As a simple illustration, consider the following two-model example with $\mathbf{x} \in \mathbb{R}^4$:

$$m_1(\mathbf{x}, \boldsymbol{\theta}_1) = \theta_{1,1}\, x_1^{\theta_{1,2}}\, (x_2^{\theta_{1,3}} + \theta_{1,4}\, x_3^{\theta_{1,5}})\ \sin(\theta_{1,6}\, x_4) \qquad \text{and} \qquad m_2(\mathbf{x}, \boldsymbol{\theta}_2) = x_1 x_2^{\theta_{2,1}} x_3^{\theta_{2,2}} x_3^{\theta_{2,3}} x_4^{\theta_{2,4}}.$$

Here, the models are composed from simple functional forms such as exponents (e.g., $x_2^{\theta_{1,3}}$, $x_3^{\theta_{1,5}}$) and trigonometics (e.g., $\sin(\theta_{1,6} x_4)$) composed together hierarchically using bivariate operators such as summation and multiplication.

We consider throughout the *M-closed* setting (Bernardo and Smith 2009), in which the response of the oracle is matched correctly by (an unknown) $i^* \in \mathscr{I}$ along with the true parameters $(\boldsymbol{\theta}^*, \sigma^*)$. We allow for the $\sigma^*$ to be positive; in other words, the oracle can provide noisy responses. The main goal is to select a sequence of design points $\mathbf{x}$ that efficiently identifies the correct model $i^*$.

### 1.2 Experimental Design

A typical OED set-up consists of the user adaptively choosing the design points $\mathbf{x}$ at which the oracle is queried, thereby obtaining the dataset of design-response pairs $(\mathbf{x}, y)$. The dataset lets the user learn the best-fit parameters $(\boldsymbol{\theta}_i, \sigma_i)$ for each model class, and more importantly pick the model class that best reflects the oracle, denoted as the $i^*$-th model. Each query is expensive to evaluate, and therefore the design-point decisions should help arrive at the correct model quickly by maximizing the information gained in identifying $i^*$ (along with its parameters $(\boldsymbol{\theta}^*, \sigma^*)$) from the responses to the queries. The two outlined goals encapsulate an exploration-exploitation style of tradeoff wherein the chosen design-response pairs should adequately explore the space to fit each model class adequately while promoting the (main) goal of quickly distinguishing the correct model from the others. Our approach to the design selection problem solves a one-step (greedy) objective of maximizing the expected information gained from the oracle query. In particular, we maximize an objective (5) below that we show (via (4) below) is similar to the mutual information gain objective widely utilized in Bayesian optimization (Shahriari et al. 2016; Frazier 2018).

### 1.3 Our Contribution

We present in Algorithm 1 below our approach to efficiently pick the correct model form along with its best-fit parameters. The algorithm maintains a belief distribution on the correctness of each model as the true representation of the oracle, along with a belief on appropriate parameter values for each model.

Each iteration consists of two key decision steps. The first determines the next design point to query the oracle, which is obtained by solving for the design point $\mathbf{x}$ where the entropy of the belief distribution of the oracle's response is maximized. Section 3.1 provides a simulation optimization algorithm to find good candidate solutions to this formulation. The second step updates the belief distributions based on the response revealed by the oracle at the selected design query. We follow a Bayesian posterior update procedure in incorporating the new information. Section 3.2 details how the belief on the appropriate parametrization of each model class is updated. These posterior distributions can exhibit features such as multiple modes that preclude closed-form descriptions in most settings of interest, and thus we follow a Markov Chain Monte Carlo (MCMC) approach to draw approximately close samples from the updated distribution. In particular, we describe a procedure based on umbrella sampling (Matthews et al. 2018) that orchestrates multiple MCMC processes to draw samples that adequately cover the modes of the posterior distribution. Finally, the belief on each model's correctness is updated using a multiplicative form.

Proposition 1 in Section 3.1 provides an unbiased estimator of the gradient of the entropy objective (5) for the stochastic approximation (SA) steps (7) that enjoys canonical rates of convergence with respect to (w.r.t.) the sampling effort when the density is available in closed form. However, the posterior beliefs can only be accessed via approximate sampling via MCMC, which makes such unbiased estimation seem out of reach for our method. Thus our Algorithm 1 utilizes a simultaneous perturbation-based finite difference construction in Algorithm 2 to estimate the gradient, where common random numbers is utilized to improve the variance properties of the estimator.

Section 4 presents preliminary results on an experimental set-up from our original motivation of symbolic discovery. We compare three different models that are composed of the same building blocks of simple functions. Algorithm 1 is run to identify the correct form,and our preliminary results show how quickly the probability of correct selection grows with the information revealed at each optimized design query.

### 1.4 Relationship to Literature

The topic of experimental design, particularly in a Bayesian framework, is rich and well-studied; an overview is provided by Ryan et al. (2016). The primary thrust of OED is in conducting experiments that maximize the information gained in each iteration. Information-theoretic metrics that are used as objectives to measure the information gained include mutual information (Drovandi et al. 2014), Jensen-Shannon divergence (Vanlier et al. 2014), and response entropy going back to Borth (1975). The specific form that the information-theoretic metric takes depends on the primary objective of the exercise. Simulation meta-modeling techniques, such as the response surface methodology (Angün et al. 2002) and stochastic kriging (Ankenman et al. 2010; Staum 2009), may aim to obtain a good fit everywhere in a compact subset of the design space, or may desire the minima (or maxima) of the functional relationship that is being modeled. A specific instance of this arises in Bayesian optimization formulations (Frazier 2018), which propose to fit a Gaussian field process to represent a nonlinear objective in order to find its minima. The mutual information maximization goal leads to an elegant design selection policy that selects design points where the estimated confidence bounds on the objective function value is the lowest.

Our problem seeks good-fit values of model parameters from a given collection of models in order to be able to better differentiate between them and identify the correct model among them. In (4) below, we present a Jensen-Shannon divergence maximization objective. We establish in (5) that this is equivalent to finding the design point $\mathbf{x}$ where the current estimated response $Y(\mathbf{x})$ variable enjoys the maximum entropy. Thus, our (greedy) policy seeks to experiment where we are the *least certain* about the oracle's response.

Our motivation of selecting the best among a finite collection of alternatives is similar to the classical simulation formulation of ranking and selection (R&S) (Hong et al. 2021). R&S's goal of minimizing the probability of incorrect selection of the best alternative bears a strong resemblance to our objective. While R&S primarily assumes static expected reward, in our set-up the performance gap between the true model and the alternatives depend on the design point $\mathbf{x}$. The contextual bandits formulation (Bietti et al. 2021) generalizes multi-armed bandits to allow the reward distribution to depend on a context. However, their motivation is different in that the best alternative may itself be context dependent and the optimal policy aims to quickly identify this relationship. In contrast, the true model in our setting is a superior fit to the oracle at every "context" $\mathbf{x}$.

Finally, the one-step greedy design-point decision problem may be extended to a sequential decision process with a finite or infinite horizon of queries. An optimal decision policy for this problem can be obtained by, for instance, deducing the corresponding value function of a Markov Decision Process formulation. Indeed, various contextual bandit formulations (Bietti et al. 2021) and Bayesian OED formulations such as A/B testing (Rainforth et al. 2024) employ reinforcement learning to estimate such value functions. The sample complexity to learn such an approximation is in general higher and may be prohibitively expensive in our context of an experiment oracle, but nevertheless this is a potential area for further exploration.

### 1.5 Notation and Definitions

We use the boldfaced notation $\mathbf{x}$ for multidimensional vectors with components $x_j$, for $j = 1,\dots,p$. The notation $\mathbf{x}^{(k)}$ is used to denote sequences of vectors, such as the value of the vector $\mathbf{x}$ in the $k$-th iteration.

The entropy of a density $f$ is defined as $H(f) \triangleq \mathsf{E}[-\log f] = -\int \log(f(u))\, f(u)du$. The Kullback-Leibler (KL) divergence between two measures with densities $f$ and $q$ is defined as

$$D_{KL}(f,q) \triangleq \mathsf{E}_f\left[\log\frac{f}{q}\right] = \int \log\frac{f(u)}{q(u)}\, f(u)\, du \;=\; -\int \log\frac{q(u)}{f(u)}\, f(u)\, du.$$

When the context is clear, we will refer to the entropy of a measure or an r.v. when the term is implicitly applied to the corresponding density, and similarly for divergence between measures and r.v.s. KL divergence is nonnegative and zero only if $f = q$ almost everywhere. But, since it not symmetric and does not obey the triangle inequality, it is not a metric. The Jensen-Shannon (JS) divergence is a generalization of $D_{KL}$ that addresses the symmetry issue as

$$D_{JS}(f,q) \;\triangleq\; \frac{1}{2}D_{KL}\left(f,\frac{f+q}{2}\right) + \frac{1}{2}D_{KL}\left(q,\frac{f+q}{2}\right) = H\left(\frac{f+q}{2}\right) - \frac{1}{2}[H(f)+H(q)].$$

More generally, it can be used to measure the divergence among any finite collection $\{f_1,\dots,f_v\}$ of densities. Define an r.v. $V$ that takes values in $\{1,\dots,v\}$ with probability $\mathsf{P}(V=j) = \pi_j$ and collect these in the probability mass function (p.m.f.) $\boldsymbol{\pi}$. Let $f_V$ represent the random mix of the $f_j$ with probability $\boldsymbol{\pi}$. The average density is $\mathsf{E}_V f_V = \sum_j \pi_j f_j(\cdot)$. Then, the JS divergence of the collection (w.r.t. to $\boldsymbol{\pi}$, also known as the *mutual information* between $V$ and $f_V$) is defined as

$$D_{JS}(\{f_j,\ j=1,\dots,v\}) \;\triangleq\; D_{JS}(V,f_V) \;=\; \sum_{j=1}^{v}\pi_j D_{KL}\left(f_j,\sum_{j=1}^{v}\pi_j f_j\right) \;=\; H(\mathsf{E}_V f_V) - \sum_{j=1}^{v}\pi_j H(f_j).$$

## 2 A MODEL FOR EXPERIMENTAL DESIGN

We now describe the primitives that are used to construct an algorithm to solve the sequential decision-making problem of identifying the best-fit model to the oracle. For each model, the algorithm maintains a joint product-form belief density $f_{\boldsymbol{\theta}_i}(\cdot)f_{\sigma_i}(\cdot)$ on the parameters $(\boldsymbol{\theta}_i,\sigma_i)$ of model $i \in \mathscr{I}$ that fit the data generated so far. Let $f_\varepsilon$ represent the density of the independent noise $\varepsilon$ in the form (1). In addition, let the p.m.f. $\boldsymbol{\mu} = (\mu_i, i \in \mathscr{I}) \in \{\boldsymbol{\mu} \geq 0 \,|\, \sum_i \mu_i = 1\}$ represent our belief in the correctness of the models as representative of the oracle.

The OED procedure iteratively updates the belief distributions to identify the correct model $i^* \in \mathscr{I}$. In iteration $k$, the algorithm chooses the best design point $\mathbf{x}^{(k)}$ to query the oracle based on the beliefs $\boldsymbol{\mu}^{(k)}$ and $f^{(k)}_{\boldsymbol{\theta}_i}$ and $f^{(k)}_{\sigma_i}$. This generates a response $y^{(k)}$ from the oracle, which in turn allows the algorithm to update the beliefs to $\boldsymbol{\mu}^{(k+1)}$ and $f^{(k+1)}_{\boldsymbol{\theta}_i}$ and $f^{(k+1)}_{\sigma_i}$. Let $\boldsymbol{\mu}^* = (0,\dots,1,\dots,0)$ be the p.m.f. with the 1 at the index $i^*$. We therefore seek to ensure that the convergence $\boldsymbol{\mu}^{(k)} \to \boldsymbol{\mu}^*$ happens quickly as $k \nearrow$.

For any $\mathbf{x} \in \mathscr{X}$, the density of the response $Y_i(\mathbf{x})$ from the $i$-th model is believed to be given by:

$$\begin{aligned} f_{Y_i(\mathbf{x})}(y) \;=\; f_i(y) \;&= \int_{\mathbb{R}^{d_i}} \int_{\mathbb{R}_+} \mathsf{P}(Y_i(\mathbf{x}) = y | \boldsymbol{\theta}_i = \boldsymbol{\theta}, \sigma_i = \sigma)\, f_{\boldsymbol{\theta}_i}(\boldsymbol{\theta})\, f_{\sigma_i}(\sigma)\, d\boldsymbol{\theta}\, d\sigma, \quad \forall i \in \mathscr{I}, \\ &= \int_{\mathbb{R}^{d_i}} \int_{\mathbb{R}_+} \frac{1}{\sigma} f_\varepsilon\left(\frac{y - m_i(\mathbf{x}, \boldsymbol{\theta})}{\sigma}\right) f_{\boldsymbol{\theta}_i}(\boldsymbol{\theta})\, f_{\sigma_i}(\sigma)\, d\boldsymbol{\theta}\, d\sigma, \quad \forall i \in \mathscr{I}, \end{aligned} \tag{2}$$

where we use the notation $f_i$ to denote $f_{Y_i}$, and the last expression uses a change of variable $y/\sigma = \varepsilon$. Our overall belief is that the response $Y(\mathbf{x})$ satisfies:

$$f_{Y(\mathbf{x})}(y) \;=\; \sum_{i \in \mathscr{I}} f_i(y)\, \mathsf{P}(i = i^*) \;\;=\; \sum_{i \in \mathscr{I}} \mu_i f_i(y). \tag{3}$$

This combines our belief $f_i$ on the individual densities with the belief $\boldsymbol{\mu}$ on the correctness of the models.

Define an integer r.v. $I$ that takes values in $\mathscr{I}$ following the belief $\boldsymbol{\mu}$. The overall response $Y(\mathbf{x})$ is thus also the average of the individual model responses $Y_i(\mathbf{x}), i \in \mathscr{I}$, w.r.t. $I$, i.e., $Y(\mathbf{x}) = \mathsf{E}_I Y_I(\mathbf{x})$, of the probabilistic mixture $Y_I(\mathbf{x})$ with mixture probabilities $\boldsymbol{\mu}$. Recall that, by assumption, the correct response is $Y_{i^*}(\mathbf{x})$. We measure the gap in performance of the belief $\boldsymbol{\mu}$ at design point $\mathbf{x}$ via the KL divergence $D_{KL}(Y_{i^*}(\mathbf{x}) \,\|\, \mathsf{E}_I Y_I(\mathbf{x}))$. This is exactly zero *everywhere in* $\mathbf{x} \in \mathscr{X}$ only when the belief $\boldsymbol{\mu}$ matches $\boldsymbol{\mu}^*$.

Our best guess for $\boldsymbol{\mu}^*$ is $\boldsymbol{\mu}$, and thus the empirical expected performance gap at the design point $\mathbf{x}$ is:

$$\sum_{i \in \mathscr{I}} \mu_i\, D_{KL}(Y_i(\mathbf{x}) \,\|\, \mathsf{E}_I Y_I(\mathbf{x})) \;=\; \mathsf{E}_I D_{KL}(Y_I(\mathbf{x}) \,\|\, \mathsf{E}_I Y(I(\mathbf{x}))) \;=\; D_{JS}(I, Y_I(\mathbf{x})),$$

where the last equality recognizes that this is the same as the JS divergence of the mixture $Y_I(\mathbf{x})$. This motivates the choice of the next design point $\mathbf{x}^{(k+1)}$ to be the argument of the optimization problem:

$$\mathbf{x}^{(k+1)} \;=\; \arg\max_{\mathbf{x}} D_{JS}(I^{(k)}, Y_{I^{(k)}}(\mathbf{x}))\,, \tag{4}$$

where $I^{(k)}$ is the value of the r.v. $I$ in the $k$-th iteration. This chosen design point $\mathbf{x}^{(k+1)}$ highlights the divergence of $\boldsymbol{\mu}^{(k)}$ from $\boldsymbol{\mu}^*$.

The objective (4) can be simplified by applying the fact that the overall belief in the response $Y(\mathbf{x})$ can be written as the expectation $\mathsf{E}_I Y_I(\mathbf{x})$. We note from the definition of the JS divergence that:

$$\begin{aligned} D_{JS}(I, Y) = H(\mathsf{E}_I Y_I) - \mathsf{E}_I H(Y_I) \;&=\; H(Y) - \sum_i \mu_i H(Y_i) \\ = H(Y) - \sum_i \mu_i \left(\mathsf{E}_{\boldsymbol{\theta}_i, \sigma_i}[H(\varepsilon) + \log(\sigma_i)]\right) \;&=\; H(Y) - H(\varepsilon) - \sum_i \mu_i \mathsf{E}_{\sigma_i}[\log(\sigma_i)]. \end{aligned}$$

The third line follows from the second since the entropy of the conditional $Y_i | \boldsymbol{\theta}_i, \sigma_i$ depends only on $\sigma_i \varepsilon$.

From the viewpoint of optimization w.r.t. the design point $\mathbf{x}$, the left hand is equivalent to the first term on the right. Thus, we can equivalently obtain the next query design by solving the formulation:

$$\mathbf{x}^{(k+1)} \;=\; \arg\max_{\mathbf{x}} \left\{ z(\mathbf{x}) \;\triangleq\; H(Y^{(k)}(\mathbf{x})) \;\; = -\mathsf{E}_{Y^{(k)}(\mathbf{x})}[\log f_{Y^{(k)}(\mathbf{x})}(\mathbf{x})] \right\}. \tag{5}$$

Intuitively, we want the design point $\mathbf{x}$ where the response $Y(\mathbf{x})$ shows the highest *entropy* (i.e., the least certainty), given our current inference from the information available to us.

This objective also bears a superficial resemblance to a standard log likelihood maximization problem encountered in fitting distributions. However, note that the expectation taken in (5) is w.r.t. the density $f_{Y(\mathbf{x})}(\mathbf{x})$ itself while it is the underlying distribution of the observable data in the standard formulation.

## 3 OPTIMIZATION ALGORITHM

The correct-model identification algorithm consists of three main steps within each iteration. First, the user decides on the optimal design point $\mathbf{x}^{(k)}$ by finding a good candidate solution to the formulation (5). Next, the oracle is queried on this design point. Finally, the resulting oracle response $y^{(k)}$ is used as additional information in updating the inference on the parameters of each model, as well as our belief on which model is correct. Refer to Algorithm 1 below, each step of which is detailed in the subsequent subsections.

We note at the outset that the third step performs an update to the belief distributions on the parameters of each candidate model. We follow a Bayesian posterior approach, and the updated posterior does not necessarily have a closed-form expression. This is especially true for the parameters of the incorrect models. We use an MCMC procedure to generate samples from the posterior distribution by constructing an MCMC process whose stationary distribution is the desired posterior. It is in general hard to obtain unbiased steady-state samples via simulations of MCMC processes, so we will maintain an approximation of the posterior-inferred distribution for the parameters $(\boldsymbol{\theta}_i, \sigma_i)$ of each model that suffers from the bias induced by the transience of MCMC sample paths. In particular, this takes the form of an empirical distribution over a finite set of samples obtained by applying a deterministic stopping rule to the MCMC sample paths. This is akin to a particle simulation approach to Bayesian inference (Drovandi et al. 2014).

Denote by $\hat{\boldsymbol{\Theta}}_i^{(k)} = \{(\boldsymbol{\theta}_{i,n}^{(k)}, \sigma_{i,n}^{(k)}),\ n = 1, \ldots, N\}$ a sample set of size $N$ obtained from the MCMC-based Bayesian posterior sampling at iteration $k$. (While our approach can accommodate differing sample set sizes $N_i$, $i \in \mathscr{I}$, we focus on the case of the same sample size $N$ for all $i \in \mathscr{I}$, to elucidate the exposition.) Then, the joint density $f_{\boldsymbol{\theta}_i}(\cdot)\, f_{\sigma_i}(\cdot)$ is approximated by $(1/N)\,\sum_n \mathbb{I}(\boldsymbol{\theta}_i = \boldsymbol{\theta}_{i,n}^{(k)},\ \sigma_i = \sigma_{i,n}^{(k)})$, where $\mathbb{I}$ is the indicator function. Consequently, we obtain an approximation for the response from the $i$-th model as

$$\hat{f}_{Y_i(\mathbf{x})}^{(k)}(y) \;=\; \hat{f}_i^{(k)}(y) \;=\; \frac{1}{N}\sum_{n=1}^{N} \frac{1}{\sigma_{i,n}^{(k)}} f_\varepsilon\left(\frac{y - m_i(\mathbf{x}, \boldsymbol{\theta}_{i,n}^{(k)})}{\sigma_{i,n}^{(k)}}\right), \tag{6}$$

with change of variable $\varepsilon = y/\sigma$. The approximation to the density of the overall response variable $Y$ is a mixture of the individual densities with our inferred probabilities $\hat{\boldsymbol{\mu}}^{(k)}$, given by $\hat{f}_Y^{(k)}(y) \;=\; \sum_i \hat{\mu}_i \hat{f}_i^{(k)}(y)$. We therefore have a form for $\hat{f}_Y$ which is similar to a kernelized approximation, with $f_\varepsilon$ as the kernel.

### 3.1 Design Point Selection

We obtain a good candidate solution to (5) using the SA iterations given in (7), where $G(\cdot)$ assembles an SA of the gradient of the entropy term $H(Y(\mathbf{x}))$, and $\gamma_t$ is the standard gain, step-length, or learning rate sequence. The following result establishes an unbiased estimator for the gradient of the entropy of an r.v.

**Proposition 1** Let $S \triangleq S(\mathbf{x}) \in \mathscr{Y} \subseteq \mathbb{R}$ be a one-dimensional r.v. with density $p_{\mathbf{x}}(y)$ that is parametrized by $\mathbf{x} \in \mathscr{X} \subseteq \mathbb{R}^d$. Define $L_{\mathbf{x}}(y) \triangleq -\log p_{\mathbf{x}}(y)$ as the negative log density of $S(\mathbf{x})$. Assume that the density satisfies two conditions almost everywhere in $x \in \mathscr{X}$:

A. the gradient $\nabla_{\mathbf{x}} L_{\mathbf{x}}(y)$ exists and is well defined; and
B. the product $p_{\mathbf{x}}(y)\, L_{\mathbf{x}}(y)$ is Lipschitz continuous, with a finite mean (stochastic) Lipschitz bound.

Let $z(\mathbf{x}) \;=\; \mathsf{E}_{S(\mathbf{x})}[H(S(\mathbf{x}))]$ be the entropy of $S(\mathbf{x})$. Then, the gradient of $z(\mathbf{x})$ at any $\mathbf{x}_0 \in \mathscr{X}$ is given by

$$\nabla_{\mathbf{x}} z(\mathbf{x}_0) \;=\; \mathsf{E}_{S(\mathbf{x}_0)}\left[(1 - L_{\mathbf{x}_0}(y))\, \nabla_{\mathbf{x}} L_{\mathbf{x}_0}(y)\right]. \tag{8}$$

**Proof of Proposition 1** We employ the likelihood method (Section VII.3 in Asmussen and Glynn 2007) to compute the gradient of $z(\mathbf{x})$ at $\mathbf{x}_0$. We start by expressing the objective function as

$$z(\mathbf{x}) \;=\; \mathsf{E}_{S(\mathbf{x})}[H(S(\mathbf{x}))] \;=\; -\int_y \log p_{\mathbf{x}}(y)\;\; p_{\mathbf{x}}(y)dy \;=\; \mathsf{E}_{S(\mathbf{x}_0)}[L_{\mathbf{x}}(y)\;\; R_{\mathbf{x},\mathbf{x}_0}(y)],$$

```
Algorithm 1: OPTIMAL-EXPERIMENTAL-DESIGN($K$)
  For iterations  $k = 1, 2, \ldots, K$  do

    1. Design Point $\mathbf{x}^{(k)}$ Selection: Find an approximate maximizer to the objective $H(Y(\mathbf{x}))$ in (5).
       Start with initial iterate $\mathbf{x}^{(0)}$, gain sequence $\{\gamma_t\}$, and finite-difference sequence $\{h_t\}$.
       For  $t = 1, 2, \ldots$ do
         (a) Obtain gradient estimate $G(\mathbf{x}^{(t+1)}) \leftarrow$ ENTROPYGRADIENT$\left(\mathbf{x}^{(t+1)}, \{\hat{\mathbf{\Theta}}_i^{(k)}, i \in \mathscr{I}\}, \boldsymbol{\mu}^{(k)}, h_t, L\right)$.
         (b) If stopping criterion satisfied, then Set $T \leftarrow t$ and break.
         (c) Set
                 $\mathbf{x}^{(t+1)} = \mathbf{x}^{(t)} + \gamma_t G(\mathbf{x}^{(t)}).$        (7)
       end for
       Set next query design as $\mathbf{x}^{(k)} \leftarrow \mathbf{x}^{(T)}$.
    2. Oracle Query: Obtain the (noisy) true response $y^{(k)}$ from the oracle at design point $\mathbf{x}^{(k)}$.
    3. Posterior Updates:
         (a) Update empirical approximations $\hat{f}_{\boldsymbol{\theta}_i}^{(k+1)}$ and $\hat{f}_{\sigma_i}^{(k+1)}$ of belief distributions $f_{\boldsymbol{\theta}_i}^{(k+1)}$ and $f_{\sigma_i}^{(k+1)}$ of
             parameters via particle sets $\{\hat{\mathbf{\Theta}}_i^{(k+1)}, i \in \mathscr{I}\}$ for each model $i \in \mathscr{I}$ using the MCMC procedure
             outlined in Section 3.2 to implement approximate Bayesian inference.
         (b) Update belief $\boldsymbol{\mu}^{(k+1)}$ of $i \in \mathscr{I}$ as the (unknown) true model $i^*$ via multiplicative update (10).

  end for
  return $i^* \leftarrow \arg\max_i \mu_i^{(K)}$.
end algorithm
```

**Algorithm 1:** Sketch of the OED iterative procedure with input $K$ specifying the total budget of queries.

where $R_{\mathbf{x},\mathbf{x}_0} \triangleq p_{\mathbf{x}}/p_{\mathbf{x}_0}$ is called the likelihood ratio of $p_{\mathbf{x}}$ w.r.t. $p_{\mathbf{x}_0}$, and the expectation in the last term is also w.r.t. $p_{\mathbf{x}_0}$. The gradient of the objective $z(\mathbf{x})$ at $\mathbf{x} = \mathbf{x}_0$ can be obtained via interchange of the integral from the expectation with the differential operator, which is allowed under Assumption B. Thus, the $u$-th component of the gradient of the entropy $z(\mathbf{x})$ of $S(\mathbf{x})$ is given by

$$\begin{aligned}
\left[\nabla_{\mathbf{x}} z(\mathbf{x})|_{\mathbf{x}=\mathbf{x}_0}\right]_u &= \frac{d}{dx_u} \mathsf{E}_{S(\mathbf{x}_0)}[L_{\mathbf{x}}(y)\, R_{\mathbf{x},\mathbf{x}_0}(y)]\Big|_{\mathbf{x}=\mathbf{x}_0} = -\int_{\mathscr{Y}} \frac{d}{dx_u}\left(\log p_{\mathbf{x}}(y)\, \frac{p_{\mathbf{x}}(y)}{p_{\mathbf{x}_0}(y)}\right)\Big|_{\mathbf{x}=\mathbf{x}_0} p_{\mathbf{x}_0}(y)\, dy \\
&= -\int_{\mathscr{Y}} \left(\frac{1}{p_{\mathbf{x}}(y)}\, \frac{dp_{\mathbf{x}}(y)}{dx_u}\, \frac{p_{\mathbf{x}}(y)}{p_{\mathbf{x}_0}(y)} + \log p_{\mathbf{x}}(y) \frac{1}{p_{\mathbf{x}_0}(y)}\, \frac{dp_{\mathbf{x}}(y)}{dx_u}\right)\Big|_{\mathbf{x}=\mathbf{x}_0} p_{\mathbf{x}_0}(y)\, dy \\
&= -\int_{\mathscr{Y}} (1 + \log p_{\mathbf{x}_0}(y))\, \frac{dp_{\mathbf{x}_0}(y)}{dx_u}\, \frac{1}{p_{\mathbf{x}_0}(y)}\, p_{\mathbf{x}_0}(y) dy = -\mathsf{E}_{S(\mathbf{x}_0)}\left[(1 + \log p_{\mathbf{x}_0}(y))\, \frac{d\,(\log p_{\mathbf{x}_0}(y))}{dx_u}\right].
\end{aligned}$$

Here, the interchange in the first line is justified by the Lipshitz continuity of $p_{\mathbf{x}}\, L_{\mathbf{x}}$, and the last expression follows from applying the differential to $L_{\mathbf{x}_0}(y)$. This leads to the expression in (8). □

Proposition 1 estimates the required gradient of the objective in (5) as an expectation of a function of the pdf $p_{\mathbf{x}_0}$ w.r.t. to the same density. This also implies that the estimator will enjoy the canonical fastest (square-root) rate of convergence w.r.t. the sample size when the density $p_{\mathbf{x}_0}$ is available in closed form.

In our setting, we do not have such an explicit form available for the density $f_{Y(\mathbf{x})}$ of $Y(\mathbf{x})$. Indeed, it is defined only implicitly as the weighted sum of the marginals emerging from (2) (via (3)) by integrating out the belief distributions of the parameters $\boldsymbol{\theta}_i$ and $\sigma_i$. The kernel density like estimator $\hat{f}_{Y(\mathbf{x})}$ is constructed from samples of $\boldsymbol{\theta}_i$ and $\sigma_i$ with $f_\varepsilon$ serving as the kernel function. Nonparametric estimation of functionals of densities has been of deep interest to the simulation and statistics community; see Beirlant et al. (1997).

The main nonparametric estimation methods use a similar kernel density estimation along with some form of resubstitution estimation (Laurent and Massart 2000; Jones 1992). Their best rate of convergence are strictly slower than the canonical rate. Additionally, we can only sample from an approximately close distribution to the Bayesian posterior by using MCMC (see Section 3.2), potentially degrading the convergence rate. Teasing out this interplay between MCMC convergence and kernel density estimation convergence remains of high interest.

For the purpose of this article, we sidestep this issue for the moment by using a simultaneous perturbation finite difference (Spall 2001) gradient estimator. Algorithm 2 outlines our estimator. Step 1 samples a unit vector uniformly from the surface of a unit ball in $\mathbb{R}^p$, and Step 2 defines the design points where the entropy objective $z(\cdot)$ is estimated in Step 4, leading to the simultaneous perturbation stochastic approximation (SPSA) gradient approximation in Step 5. In Step 3(d), we exploit the form (1) to construct the estimator using common random numbers to reduce its variance. The sampling effort in constructing $G(\mathbf{x})$ is thus the same as it would be if we were able to directly implement (8) using a closed-form expression for $f_Y(\mathbf{x})$.

**Algorithm 2:** EntropyGradient($\mathbf{x}$, $\{\hat{\boldsymbol{\Theta}}_i, i \in \mathscr{I}\}$, $\boldsymbol{\mu}$, $h$, $L$)

```
  1. Sample a standard Gaussian $\Delta \sim \mathscr{N}(0,I)$ in $\mathbb{R}^p$ and normalize to unit direction $\Delta \leftarrow \frac{\Delta}{\|\Delta\|_2}$.
  2. Set $\underline{\mathbf{x}} \leftarrow \mathbf{x} - h\Delta$ and $\bar{\mathbf{x}} \leftarrow \mathbf{x} + h\Delta$.
  3. **Generate** a set of sample pairs $\{(\underline{y}_\ell, \bar{y}_\ell), \ell = 1, \ldots, L\}$ from $Y \sim \hat{f}_Y$.
     **For** $\ell = 1, \ldots, L$:
       (a) Sample a model index $i$ from the discrete $I \sim \boldsymbol{\mu}$.
       (b) Sample a parameter sample index $n \sim \mathscr{U}\{1, \ldots, N\}$.
       (c) Sample $\varepsilon \sim f_\varepsilon$.
       (d) Set $\underline{y}_\ell \leftarrow m_i(\underline{\mathbf{x}}, \theta_{i,n}) + \sigma_{i,n}\varepsilon$ and $\bar{y}_\ell \leftarrow m_i(\bar{\mathbf{x}}, \theta_{i,n}) + \sigma_{i,n}\varepsilon$ .
     **end for**
  4. **Estimate** $z(\underline{\mathbf{x}}) = -\frac{1}{L}\sum_{\ell=1}^{L} \log \hat{f}_Y(\underline{y}_\ell)$ and $z(\bar{\mathbf{x}}) = -\frac{1}{L}\sum_{\ell=1}^{L} \log \hat{f}_Y(\bar{y}_\ell)$.
  5. **Estimate Gradient** $G(\mathbf{x})$ with $u$-th component $G_u(\mathbf{x}) \leftarrow \frac{z(\bar{\mathbf{x}})-z(\underline{\mathbf{x}})}{2h\Delta_u}$ for $u = 1, \ldots, d$.

  **return** $G(\mathbf{x})$
**end algorithm**
```

**Algorithm 2:** Procedure to estimate gradient of entropy $z(\mathbf{x})$ objective in (5), with input $\mathbf{x}$ specifying the design point where the gradient is desired, particle sets $\{\hat{\boldsymbol{\Theta}}_i, i \in \mathscr{I}\}$ representing an approximate sample from Bayesian posteriors of parameters, p.m.f. $\boldsymbol{\mu}$ is the current belief in the correctness of each model $i$ as $i^*$, the $h$ as the finite difference width, and $L$ defining the number of samples to use in the SPSA estimation.

We expect that the convergence of the SA iterations (7) to a first-order optimal solution should be near the best rate of $t^{-1/3}$ achieved by centered-difference Keifer Wolfowitz algorithms (Asmussen and Glynn 2007, Ch VIII.3). We provide a brief informal sketch of the arguments. This rate result needs the entropy estimators in Algorithm 2 to converge at the canonical rate $L^{-1/2}$, which can be achieved for the resubstitution-based entropy estimator that uses kernel density estimators if the density $f_i^{(k)}$ satisfies the Holder's inequality with a sufficiently strong coefficient (Hall and Morton 1993). The convergence of the distribution of the MCMC iterates to the parameter posterior too can be exponentially fast (in a total variation sense) with sufficient continuity conditions (such as the Holder's inequality) on the likelihood probability model $f_\varepsilon$. Thus, estimation of entropy can attain near canonical rates, with a small bias from the finite-iterations termination of the MCMC posterior sampler.

### 3.2 Post-Query Belief Updates

We follow a Bayesian inference viewpoint in updating our belief distributions on model parameters $\boldsymbol{\theta}_i$ and $\sigma_i$ for each model $i$, as well as the belief $\boldsymbol{\mu}$ on the correctness of the models. We maintain a product-form joint density $f_{\boldsymbol{\theta}_i,\sigma_i}(\cdot,\cdot) = f_{\boldsymbol{\theta}_i}(\cdot)\, f_{\sigma_i}(\cdot)$ on $\mathbb{R}^{d_i} \times \mathbb{R}_+$ of the likely values of $\boldsymbol{\theta}_i$ and $\sigma_i$ given the observational data. The Bayes' update rule for the belief on latent variables such as $\boldsymbol{\theta}_i$ and $\sigma_i$ posits, upon observing the $k$-th data pair $(\mathbf{x}^{(k)}, y^{(k)})$, that:

$$f^{(k)}_{\boldsymbol{\theta}_i}(\boldsymbol{\theta})\, f^{(k)}_{\sigma_i}(\sigma) \quad \propto \quad \mathsf{P}(Y_i(\mathbf{x}^{(k)}) = y^{(k)} \,|\, \boldsymbol{\theta}, \sigma)\;\; f^{(k-1)}_{\boldsymbol{\theta}_i}(\boldsymbol{\theta})\, f^{(k-1)}_{\sigma_i}(\sigma), \tag{9}$$

where the first term represents the likelihood of $y^{(k)}$ under the $i$-th model given parameters $\boldsymbol{\theta}$ and $\sigma$, and it is derived via (1) as $f_\varepsilon((y^{(k)} - m_i(\mathbf{x}^{(k)}, \boldsymbol{\theta}))/\sigma)$. As is usual for Bayesian update rules, the normalizing constant for the posterior is hard to compute in closed form in general, and so generating an i.i.d. sample set $\boldsymbol{\Theta}_i$ is not straightforward. We follow the MCMC literature in sampling the posterior.

A key difficulty in using MCMC procedures lies with the pathologies of the posterior distribution, for example a multi-modal form where the MCMC struggles to jump between modes. Shalizi (2009) study the properties of the Bayesian update rule as the number of samples grow. They show that the posterior distribution concentrates around the solution to the maximum (negative) log likelihood estimator (MLLE):

$$\max_{\boldsymbol{\theta},\sigma} \frac{1}{K} \sum_{k=1}^{K} \left[ -\log \mathsf{P}(Y_i(\mathbf{x}^{(k)}) = y^{(k)} \,|\, \boldsymbol{\theta}, \sigma) \right] \quad \xrightarrow[K\to\infty]{a.s.} \quad \max_{\boldsymbol{\theta},\sigma} \mathsf{E}_{Y_{i^*}} \left[ -\log \mathsf{P}(Y_i(\mathbf{x}) = y \,|\, \boldsymbol{\theta}, \sigma) \right].$$

When the model is correctly specified and that the optimal parameters $\boldsymbol{\theta}^*$ and $\boldsymbol{\sigma}^*$ are unique, they further show that this convergence holds almost surely (a.s.) to atoms centered on these values. When the model class is misspecified, the posterior converges to measures that live on the local optima of the MLLE problem.

We recognize the possibility of scenarios with multi-modal posterior distributions for $\boldsymbol{\theta}_i$ and $\sigma_i$ where regular sampling approaches might encounter difficulties. In such cases, we propose to adapt umbrella sampling (Matthews et al. 2018). For clarity, we define $\bar{\boldsymbol{\theta}} \triangleq (\boldsymbol{\theta}, \sigma)$ and drop the superscripts $(k)$. The $\bar{\boldsymbol{\theta}}$ is to be drawn from a multi-modal posterior $f(\bar{\boldsymbol{\theta}}) = \mathsf{P}(\bar{\boldsymbol{\theta}} \mid Y) \propto \mathsf{P}(Y(\mathbf{x}) = y \mid \bar{\boldsymbol{\theta}})\mathsf{P}(\bar{\boldsymbol{\theta}})$ defined via Bayes rule. Suppose $\{\Xi_o\}_{o=1}^{M}$ is a partition of the support set of $\bar{\boldsymbol{\theta}}$ into (possibly overlapping) sub-regions such that $\bigcup_o \Xi_o$ spans the support space. We can then define umbrella functions $\{\psi_o(\bar{\boldsymbol{\theta}})\}_{o=1}^{M}$, which are nonnegative biasing functions that are usually normalized or have known integrals, such that $\psi_o(\bar{\boldsymbol{\theta}}) > 0$ when $\bar{\boldsymbol{\theta}} \in \Xi_o$. If modes of the posterior are clearly identifiable from exploratory sampling runs, e.g., by solving the sample version of the MLLE formulations given above, then umbrella functions can be placed at or near the modes.

Each subregion $\Xi_o$ defines a biased distribution $f_o(\bar{\boldsymbol{\theta}}) \propto f(\bar{\boldsymbol{\theta}})\psi_o(\bar{\boldsymbol{\theta}})$. The umbrella functions $\psi_o$ flatten the posterior within their respective regions and allow sampling from underrepresented areas, like less probable modes. For each umbrella function $\psi_o$, $o \in \{1, \ldots, M\}$, we sample $\bar{\boldsymbol{\theta}}_{o,q_1}, \ldots, \bar{\boldsymbol{\theta}}_{o,q_m} \sim f_o(\bar{\boldsymbol{\theta}})$. Note:

$$f(\bar{\boldsymbol{\theta}}) \propto \frac{f_o(\bar{\boldsymbol{\theta}})}{\psi_o(\bar{\boldsymbol{\theta}})} = \frac{\mathsf{P}(\bar{\boldsymbol{\theta}} \mid Y)\psi_o(\bar{\boldsymbol{\theta}})}{\psi_o(\bar{\boldsymbol{\theta}})} = \mathsf{P}(\bar{\boldsymbol{\theta}} \mid Y).$$

Thus, the samples from the individual umbrella densities $f_o$ can be combined together to reconstruct samples from the original posterior by reweighting the samples appropriately. We next introduce variables $g_o$, $o \in \{1, \ldots, M\}$, as an offset (normalization constant in the log domain) for each umbrella function $\psi_o$ and define the weight of each sample by $w_{o,q} = \left[\sum_{r=1}^{M} n_r \exp(g_r)\psi_r(\bar{\boldsymbol{\theta}}_{o,q})\right]^{-1}$.

These weights can be used in approximating quantities that are functions of the posterior $f$. For instance, we approximate the expectation of a function $v(\bar{\theta})$ as $\mathsf{E}_f[v(\bar{\boldsymbol{\theta}})] \approx \sum_{o=1}^{M} \sum_{q=1}^{q_o} w_{o,q} v(\bar{\boldsymbol{\theta}}_{o,q})$. The values of $g_r$ are determined according to the weighted histogram analysis method (WHAM) (Kumar et al. 1992) or the multistate Bennett acceptance ratio estimator (MBAR) (Shirts and Chodera 2008).

The Bayesian update formula also provides us with a method to update our belief in the likelihood that model $i$ is the correct model for the oracle after observing $y^{(k)}$ at $\mathbf{x}^{(k)}$:

$$\mu_i^{(k)} \;\propto\; \mathsf{P}(Y_i(\mathbf{x}^{(k)}) = y^{(k)})\; \mu_i^{(k-1)}. \tag{10}$$

Here, the model-based likelihood $\mathsf{P}(Y_i(\mathbf{x}^{(k)}) = y^{(k)})$ is given by $f_i$ from (2). In practice, we substitute in $\hat{f}_i^{(k)}$ from (6) once the new sample set $\boldsymbol{\Theta}_i^{(k)}$ has been generated. Note that (10) takes the form of a multiplicative update of $\mu^{(k)}$. We remark here that multiplicative update algorithms are a mature field of research that includes methods such as (Ada-)Boosting weak learners, and so on.

## 4 NUMERICAL EXPERIMENTS

This section presents preliminary results on our implementation of Algorithm 1 on an example with $\mathbf{x} \in \mathbb{R}^4$. We pick an equation from Feynman's lecture notes on damped oscillations of the form $E = cm^{e_1}(\omega^{e_2} + \omega_0^{e_3})z^{e_4}$. This model has four inputs $x \triangleq (m, \omega, \omega_0, z)$ and five parameters $\boldsymbol{\theta} \triangleq (c, e_1, e_2, e_3, e_4)$, and the true model $i^*$ has parameters $\boldsymbol{\theta}^* = (c = 1/4,\, e_1 = 1,\, e_2 = 2,\, e_3 = 2,\, e_4 = 2)$. The noise parameter is set to $\sigma^* = 7.5$.

The above equation can be composed from four simple functions, namely $m^{e_1}$, $\omega^{e_2}$, $\omega_0^{e_3}$, and $z^{e_4}$ along with the multiplicative parameter $c$ in front. Since each of the four models are exponential functions, we impose nonnegativity constraints on both $\mathbf{x}$ and $\boldsymbol{\theta}$ to avoid numerical instability. The compositional symbolic discovery algorithm described in (Cornelio et al. 2023) hierarchically constructs and evaluates expressions generated by applying pairwise operators to this set of four simple functions. We pick two: $E = cm^{e_1}\omega^{e_2}\omega_0^{e_3}z^{e_4}$, and $E = cm^{e_1}(\omega^{e_2} + z^{e_4})\omega_0^{e_3}$. We use these three as the set of model alternatives, with $i^* = 1$.

**Algorithm Settings.** A particle count of $N = 3000$ is chosen to represent the empirical approximation of the posteriors $\hat{f}_{\boldsymbol{\theta}_i}(\cdot)\hat{f}_{\sigma_i}(\cdot)$ at every iteration. The initial prior is sampled uniformly from the hypercube $[0, 5.0]^5$ and a Hamiltonian Monte Carlo (Ghosh et al. 2023) implementation is used to sample from the updated posteriors of the parameters. The SA gain sequence $\gamma_t$ and the SPSA finite-difference variable $h_t$ are set using typical forms: $\gamma_t = 0.5/\lceil \frac{t}{5} \rceil$ and $h_t = 0.05/\lceil \frac{t}{5} \rceil^{(1/3)}$. We take $L = 1000$ samples in constructing the gradient estimates for the response entropy in the SPSA algorithm. SA is limited to $T = 20$ iterations.

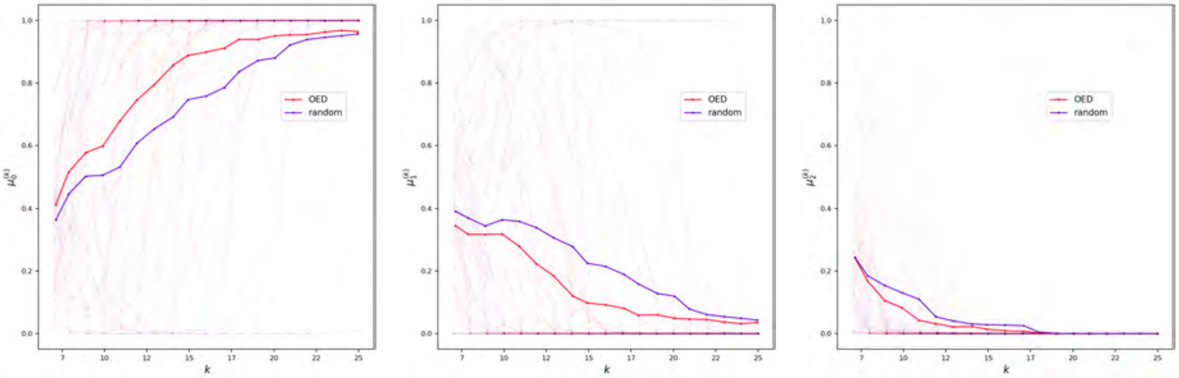

**Figure 1:** Comparison of Algorithm 1 implementing our OED approach with an approach that generates design points uniformly at random. The three figures respectively plot the belief $\mu_i^{(k)}$, $i = 1, 2, 3$ that model $i$ is the true model as $k$ grows. The lines in bold plot the average belief over 30 replications.

Figure 1 presents the results obtained from Algorithm 1 for the problem setting outlined above. Plots of the evolving belief that each of the three models represent the oracle are respectively presented in the three panels. The lines in bold plot the average belief over 30 replications. The OED algorithm is compared to a scheme RANDOM, which replaces Step 1 of Algorithm 1 with sampling to pick the design queries, generating each design point uniformly from the hypercube $\mathbf{x}^{(k)} \sim \mathscr{U}[0, 6.75]^4$. For a fair comparison, the SA algorithms are also restricted to the same hypercube. The probability of correct selection grows for both approaches, and the optimization approach reaches a higher value distinctly and quickly.

## 5 CONCLUSIONS AND FUTURE WORK

In this work, we studied an interesting OED formulation that arises from an exercise in symbolic discovery of functional relationships in scientific data. The algorithm presented here combined an SA algorithm with an SPSA-based gradient derived from steady-state samples of an MCMC procedure. Many avenues of fruitful pursuit emerge for future work. The combination of MCMC sampling of a steady-state distribution and estimation of the gradient of a function defined over them poses a challenging convergence analysis question. Another promising direction is in studying the convergence of the correct model identification problem formally using a metric such as minimizing the probability of incorrect model selection like in the ranking and selection literature (Hong et al. 2021). We have limited our MCMC implementation to plain Hamiltonian Monte Carlo in the experiments presented in Section 4 as an expedient. In follow-up work, we will implement the full umbrella sampling technique outlined in Section 3.2 to further improver convergence in the numerical results. Finally, the interaction of this OED algorithm with the programmatic symbolic discovery problem (e.g., Cornelio et al. 2023) remains high in our list of future prerogatives.

## REFERENCES

Angün, E., J. P. C. Kleijnen, D. den Hertog, and G. Gürkan. 2002. "Response Surface Methodology Revisited". In *2002 Winter Simulation Conference (WSC)* 1:377–383, https://doi.org/10.1109/WSC.2002.1172907.

Ankenman, B., B. L. Nelson, and J. Staum. 2010. "Stochastic Kriging for Simulation Metamodeling". *Operations Research* 58(2):371–382.

Asmussen, S., and P. Glynn. 2007. *Stochastic Simulation: Algorithms and Analysis*. Stochastic Modelling and Applied Probability. Springer New York.

Beirlant, J., E. Dudewicz, L. Gyor, and E. Meulen. 1997. "Nonparametric Entropy Estimation: An Overview". *International Journal of Mathematical and Statistical Sciences* 6(1):17-39.

Bernardo, J. M., and A. F. Smith. 2009. *Bayesian Theory*, Volume 405. John Wiley & Sons.

Bietti, A., A. Agarwal, and J. Langford. 2021. "A Contextual Bandit Bake-Off". *Journal of Machine Learning Research* 22(1):1-49.

Borth, D. M. 1975. "A Total Entropy Criterion for the Dual Problem of Model Discrimination and Parameter Estimation". *Journal of the Royal Statistical Society: Series B (Methodological)* 37(1):77–87.

Cheng, Y., and Y. Shen. 2005. "Bayesian Adaptive Designs for Clinical Trials". *Biometrika* 92(3):633–646.

Clarkson, K. L., C. Cornelio, S. Dash, J. Goncalves, L. Horesh, and N. Megiddo. 2022. "Bayesian Experimental Design for Symbolic Discovery". *Arxiv:2211:15860*.

Cornelio, C., S. Dash, V. Austel, T. Josephson, J. Goncalves, K. Clarkson, *et al*. 2023. "Combining Data and Theory for Derivable Scientific Discovery with AI-Descartes". *Nature Communications* 14:1777(1-10).

Drovandi, C. C., J. M. McGree, and A. N. Pettitt. 2014. "A Sequential Monte Carlo Algorithm to Incorporate Model Uncertainty in Bayesian Sequential Design". *Journal of Computational and Graphical Statistics* 23(1):3–24.

Frazier, P. I. 2018. *Bayesian Optimization*, Chapter 11, 255–278. INFORMS TutORials in Operations Research.

Gal, Y., R. Islam, and Z. Ghahramani. 2017. "Deep Bayesian Active Learning with Image Data". In *Proceedings of the 34th International Conference on Machine Learning – Volume 70*, August 7–9, Sydney, Australia, 1183–1192.

Ghosh, S., Y. Lu, and T. Nowicki. 2023. "On Convergence of Hamiltonian Monte Carlo with Asymmetrical Momentum Distributions". *Arxiv:2110:12907*.

Hall, P., and S. C. Morton. 1993. "On the Estimation of Entropy". *Annals of the Institute of Statistical Mathematics* 45:69–88.

Hong, L. J., W. Fan, and J. Luo. 2021. "Review on Ranking and Selection: A New Perspective". *Frontiers of Engineering Management* 8:321–343.

Jones, M. C. 1992. "Estimating Densities, Quantiles, Quantile Densities and Density Quantiles". *Annals of the Institute of Statistical Mathematics* 44:721–727.

Kumar, S., J. M. Rosenberg, D. Bouzida, R. H. Swendsen, and P. A. Kollman. 1992. "The Weighted Histogram Analysis Method for Free-Energy Calculations on Biomolecules. I. The Method". *Journal of Computational Chemistry* 13(8):1011–1021.

Laurent, B., and P. Massart. 2000. "Adaptive Estimation of a Quadratic Functional by Model Selection". *The Annals of Statistics* 28(5):1302–1338.

Levenberg, K. 1944. "A Method for the Solution of Certain Non-Linear Problems in Least Squares". *Quart. Appl. Math.* 2:!64–168.

Matthews, C., J. Weare, A. Kravtsov, and E. Jennings. 2018. “Umbrella Sampling: A Powerful Method to Sample Tails of Distributions”. *Monthly Notices of the Royal Astronomical Society* 480(3):4069–4079.

Melendez, J. A., R. J. Furnstahl, H. W. Grießhammer, J. A. McGovern, D. R. Phillips, and M. T. Pratola. 2020. “Designing Optimal Experiments: An Application to Proton Compton Scattering”. *The European Physical Journal A* 57(3).

Rainforth, T., A. Foster, D. R. Ivanova, and F. Bickford Smith. 2024. “Modern Bayesian Experimental Design”. *Statistical Science* 39(1):100–114.

Ryan, E. G., C. C. Drovandi, J. M. McGree, and A. N. Pettitt. 2016. “A Review of Modern Computational Algorithms for Bayesian Optimal Design”. *International Statistical Review* 84(1):128–154.

Shababo, B., B. Paige, A. Pakman, and L. Paninski. 2013, January. “Bayesian Inference and Online Experimental Design for Mapping Neural Microcircuits”. In *Advances in Neural Information Processing Systems*, December 5–10, Lake Tahoe, NV, USA, 26:1304-1312.

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(1):148–175.

Shalizi, C. R. 2009. “Dynamics of Bayesian Updating with Dependent Data and Misspecified Models”. *Electronic Journal of Statistics* 3:1039–1074.

Shirts, M. R., and J. D. Chodera. 2008. “Statistically Optimal Analysis of Samples from Multiple Equilibrium States”. *The Journal of Chemical Physics* 129(12):124105.

Spall, J. 1998. “An Overview of the Simultaneous Perturbation Method for Efficient Optimization”. *Johns Hopkins APL Technical Digest* 19(4):482–492.

Staum, J. 2009. “Better Simulation Metamodeling: The Why, What, and How of Stochastic Kriging”. In *Proceedings of the 2009 Winter Simulation Conference (WSC)*, 119–133 https://doi.org/10.1109/WSC.2009.5429320.

Vanlier, J., C. A. Tiemann, P. A. Hilbers, and N. A. van Riel. 2014. “Optimal Experiment Design for Model Selection in Biochemical Networks”. *BMC Systems Biology* 8(1):20(1-15).

Watson, A. B. 2017. “QUEST+: A General Multidimensional Bayesian Adaptive Psychometric Method”. *Journal of Vision* 17(3):10, 1–27.

## AUTHOR BIOGRAPHIES

**KENNETH L. CLARKSON** (email, web) is a Distingished Research Scientist, and manager of a theoretical computer science group at IBM Research; his research focuses on algorithms, with prior work in a geometric setting, and more recently on matrix computations, in both cases often using randomness as a key resource.

**SOUMYADIP GHOSH** is a member of IBM Research in the math. of computation department at Yorktown Heights. His research focuses on stochastic optimization and decision making under uncertainty, with applications in training large statistical learning models. His email address is ghoshs@us.ibm.com.

**JOAO P. GONCALVES** is a member of IBM Research in the mathematics of computation department at Yorktown Heights. He’s interested in computational optimization, especially linear programming and mixed-integer linear programming. He works on the implementation of optimization solution methods in a variety of application areas. His email address is jpgoncal@us.ibm.com.

**RIK SENGUPTA** is a member of IBM Research in the Mathematics of Computation department at Cambridge, MA. He is interested in complexity theory, learning algorithms, and computational approaches to algorithmic fairness. His email address is rik@ibm.com.

**MARK S. SQUILLANTE** is a Distinguished Research Scientist in Mathematics of Computation at IBM Research. His research interests broadly concern mathematical foundations of the analysis, modeling and optimization of the design, control and performance of stochastic systems. His email address is mss@us.ibm.com and his website is https://research.ibm.com/people/mark-squillante.

**DMITRIY ZUBAREV** is a member of IBM Research in the mathematics of computation department at Almaden, CA. His background is computational and theoretical chemistry, with contributions spanning stochastic projections methods for fermionic systems, formal methods in chemical evolution, and generative computation frameworks in quantum chemistry. His current research focuses on equilibria in multi-agent games and their connection to development of communication and reasoning protocols. His email address is dmitry.zubarev@ibm.com.