A Scalable Gradient-Free Method for Bayesian Experimental Design with Implicit Models

Jiaxin Zhang
Computer Science
and Mathematics Division
Oak Ridge National Laboratory
zhangj@ornl.gov

Sirui Bi
Computational Sciences
and Engineering Division
Oak Ridge National Laboratory
bis1@ornl.gov

Guannan Zhang
Computer Science
and Mathematics Division
Oak Ridge National Laboratory
zhangg@ornl.gov

Abstract

Bayesian experimental design (BED) is to answer the question that how to choose designs that maximize the information gathering. For implicit models, where the likelihood is intractable but sampling is possible, conventional BED methods have difficulties in efficiently estimating the posterior distribution and maximizing the mutual information (MI) between data and parameters. Recent work proposed the use of gradient ascent to maximize a lower bound on MI to deal with these issues. However, the approach requires a sampling path to compute the pathwise gradient of the MI lower bound with respect to the design variables, and such a pathwise gradient is usually inaccessible for implicit models. In this paper, we propose a novel approach that leverages recent advances in stochastic approximate gradient ascent incorporated with a smoothed variational MI estimator for efficient and robust BED. Without the necessity of pathwise gradients, our approach allows the design process to be achieved through a unified procedure with an approximate gradient for implicit models. Several experiments show that our approach outperforms baseline methods, and significantly improves the scalability of BED in high-dimensional problems.

Proceedings of the 24[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130.

1 Introduction

Experimental design plays an essential role in all scientific disciplines. Our ultimate goal is to determine designs that maximize the information gathered through the experiments so that improve our understanding on model comparison or parameter estimations. A broadly used approach is Bayesian experimental design (BED) (Chaloner and Verdinelli, 1995) that aims to find optimal design $\boldsymbol{\xi}^*$ to maximize a utility function $I(\boldsymbol{\xi})$, which is typically defined by the mutual information (MI) between data and model parameters

$$I(\boldsymbol{\xi}) = \mathbb{E}_{p(\boldsymbol{\theta},\boldsymbol{y}|\boldsymbol{\xi})}\left[\log p(\boldsymbol{\theta}|\boldsymbol{y},\boldsymbol{\xi}) - \log p(\boldsymbol{\theta})\right] \qquad (1)$$

where $\boldsymbol{\xi}$ represent the experimental design, $p(\boldsymbol{\theta})$ is the prior distribution of model parameters and $p(\boldsymbol{\theta}|\boldsymbol{y},\boldsymbol{\xi})$ is the posterior distribution of $\boldsymbol{\theta}$ given data $\boldsymbol{y}$ with design $\boldsymbol{\xi}$. However, finding $\boldsymbol{\xi}^*$ by maximizing $I(\boldsymbol{\xi})$ is a challenging task in practice because evaluating $I(\boldsymbol{\xi})$ needs a nested Monte Carlo simulator, which is computationally intensive, particularly in high-dimensional settings (Drovandi et al., 2018).

Most of the existing BED studies focus on the *explicit models* (Chaloner and Verdinelli, 1995; Sebastiani and Wynn, 2000; Foster et al., 2019, 2020) where the likelihood is analytically known but in nature and science, the more common scenario is the *implicit models* (Kleinegesse and Gutmann, 2019; Overstall et al., 2018), where the likelihood is intractable and not evaluated directly but sampling is possible. In other words, the model is specified based on a stochastic data generating simulator and typically has no access to the analytical form and the gradients of the joint density $p(\boldsymbol{\theta},\boldsymbol{y}|\boldsymbol{\xi})$ and marginal density $p(\boldsymbol{y}|\boldsymbol{\xi})$. The resulting BED scheme typically shares a two-stage feature: build a pointwise estimator of $I(\boldsymbol{\xi})$ and then feed this "black-box" estimator to a separate outer-level optimizer to find the optimal design $\boldsymbol{\xi}^*$. This framework substantially increases the overall computational cost and is difficult to scale the BED process to a high dimensional design

space. Recent studies (Kleinegesse and Gutmann, 2020; Foster et al., 2020; Harbisher et al., 2019) alleviate the challenges by introducing stochastic gradient-based approaches but they rely on the models with tractable likelihood functions or assume the gradient can be reasonably approximated by pathwise gradient estimators with sampling path (Kleinegesse and Gutmann, 2020), which requires that we can sample from the data distribution $p(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{\xi})$ by sampling from a base distribution $p(\boldsymbol{\epsilon})$ and then transforming the samples through a specific function $g(\boldsymbol{\epsilon}; \boldsymbol{\theta}, \boldsymbol{\xi})$, which is called the sampling path, i.e., $\boldsymbol{y} \sim p(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{\xi}) \iff \boldsymbol{y} = g(\boldsymbol{\epsilon}; \boldsymbol{\theta}, \boldsymbol{\xi}), \boldsymbol{\epsilon} \sim p(\boldsymbol{\epsilon})$. This is unlike the scope of our paper that focuses on the BED for implicit models without gradients.

1.1 Related Work

Foster et al. (2019) recently proposed to use a lower bound of MI for BED. This study relies on variational approximations to the likelihood and posterior but it is a two-stage approach where the optimal designs were determined by a separate Bayesian optimization (BO). Unfortunately, this approach has a limitation in scaling to high-dimensional design problems. A follow-up study developed by Foster et al. (2020) aims to address the scalability issue by introducing a unified stochastic gradient-based approach. However, they assumed the models with the tractable explicit likelihood or gradient approximations are available. In the scope of BED for implicit models, Ao and Li (2020) proposed an approximate KLD based BED method for models with intractable likelihoods; Kleinegesse and Gutmann (2019); Kleinegesse et al. (2020) have recently considered the use of MI combined with likelihood-free inference by ratio estimation to approximate the posterior distribution but this method is often computationally intensive. The authors rectify this in a follow-up study (Kleinegesse and Gutmann, 2020) that leverages mutual information neural estimation (MINE) (Belghazi et al., 2018) to jointly determine the optimal design and the posterior distribution. However, for the experimental design without gradients, this method falls back to the gradient-free methods, i.e., BO at the expense of reduced scalability, to solve the optimization problems within the two-stage framework. Moreover, the variational BED methods (Kleinegesse and Gutmann, 2020; Foster et al., 2019, 2020) may yield an unstable design with a high variance on the posterior distribution due to the use of variational MI estimators that exhibit a high variance argued by Song and Ermon (2019).

Contributions In this paper, we propose a novel Bayesian experimental design approach for implicit models without available gradients. This scalable method can address the aforementioned technical challenges, particularly in high dimensional problems.

- We propose a general unified framework that leverages stochastic approximate gradient without the requirement or assumption of pathwise gradients and sampling paths for implicit models.
- We propose to use a smoothed MI lower bound to conduct robust MI estimation and optimization, which allows the variance of the design and posterior distribution to be much lower than existing approaches.
- We show the superior performance of the proposed approach through several experiments and demonstrate that the approach enables the optimization to be performed by a stochastic gradient ascent algorithm and thus well scaled to considerable high dimensional design problems.

2 Bayesian Experimental Design

The Bayesian experimental design (BED) framework aims at choosing an experimental design $\boldsymbol{\xi}$ to maximize the information gained about some parameters of interest $\boldsymbol{\theta}$ from the outcome $\boldsymbol{y}$ of the experiment. Typically, the BED framework begins with a Bayesian model of the experimental process, including a prior distribution $p(\boldsymbol{\theta})$ and a likelihood $p(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{\xi})$. The information gained about $\boldsymbol{\theta}$ from running the experiment with design $\boldsymbol{\xi}$ and observed outcome $\boldsymbol{y}$ can be interpreted by the reduction in entropy from the prior to posterior

$$\mathrm{IG}(\boldsymbol{y}, \boldsymbol{\xi}) = \mathcal{Q}[p(\boldsymbol{\theta})] - \mathcal{Q}[p(\boldsymbol{\theta}|\boldsymbol{y}, \boldsymbol{\xi})]. \tag{2}$$

To define a metric to quantify the utility of the design $\boldsymbol{\xi}$ before running experiments, an expected information gain (EIG), $I(\boldsymbol{\xi})$ is often used by

$$I(\boldsymbol{\xi}) = \mathbb{E}_{p(\boldsymbol{y}|\boldsymbol{\xi})}[\mathcal{Q}[p(\boldsymbol{\theta})] - \mathcal{Q}[p(\boldsymbol{\theta}|\boldsymbol{y}, \boldsymbol{\xi})]]. \tag{3}$$

Eq. (3) can also be interpreted as a mutual information (MI) between $\boldsymbol{\theta}$ and $\boldsymbol{y}$ with a specified $\boldsymbol{\xi}$,

$$I_{\mathrm{MI}}(\boldsymbol{\xi}) = \mathbb{E}_{p(\boldsymbol{\theta})p(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{\xi})}\left[\log \frac{p(\boldsymbol{y}|\boldsymbol{\theta}, \boldsymbol{\xi})}{p(\boldsymbol{y}|\boldsymbol{\xi})}\right]. \tag{4}$$

The BED problem is therefore defined as

$$\boldsymbol{\xi}^* = \arg\max_{\boldsymbol{\xi} \in \boldsymbol{\Xi}} I_{\mathrm{MI}}(\boldsymbol{\xi}), \tag{5}$$

where $\boldsymbol{\Xi}$ is the feasible design domain. The most challenging task in the BED framework is how to efficiently and accurately estimate $I_{\mathrm{MI}}(\boldsymbol{\xi})$ in Eq. (4) and optimize $I_{\mathrm{MI}}(\boldsymbol{\xi})$ via Eq. (5) to obtain the optimal design $\boldsymbol{\xi}^*$. The following section will discuss the two core tasks in the BED framework, that is MI estimation and optimization.

2.1 Mutual Information Estimation

Mutual information (MI) estimation plays a critical role in many important problems, not only the BED framework but also other machine learning tasks such as reinforcement learning (Pathak et al., 2017) and representation learning (Chen et al., 2016; Oord et al., 2018). However, estimating mutual information from samples is always challenging (McAllester and Stratos, 2020). This is because that, in general, neither $p(\boldsymbol{\theta}|\boldsymbol{y},\boldsymbol{\xi})$ and $p(\boldsymbol{y}|\boldsymbol{\xi})$ have analytical closed-form so that classical Monte Carlo methods are intractable to compute the integral, specifically in high dimensions. One potential approach is to use a nested Monte Carlo (NMC) estimator (Myung et al., 2013; Rainforth et al., 2018), which is given by

$$I_{\mathrm{NMC}}(\boldsymbol{\xi}) = \frac{1}{N}\sum_{i=1}^{N}\log\left[\frac{p(\boldsymbol{y}_i|\boldsymbol{\theta}_{i,0},\boldsymbol{\xi})}{\frac{1}{m}\sum_{j=1}^{M}p(\boldsymbol{y}_i|\boldsymbol{\theta}_{i,j},\boldsymbol{\xi})}\right], \quad (6)$$

where $\boldsymbol{\theta}_{i,j} \sim p(\boldsymbol{\theta})$ are independent and identically distributed (i.i.d) samples, and $\boldsymbol{y}_i \sim p(\boldsymbol{y}|\boldsymbol{\theta}=\boldsymbol{\theta}_{i,0},\boldsymbol{\xi})$. However, NMC estimator requires both the inner and outer integrals and therefore converges slowly at an overall rate of $\mathcal{O}(T^{-1/3})$ (Rainforth et al., 2018) in the total computational cost $T=\mathcal{O}(NM)$.

Recently, there has been an increasingly interest in MI estimation with variational methods (Barber and Agakov, 2003; Nguyen et al., 2010; Foster et al., 2019), which can benefit from a natural incorporation with deep learning algorithms (Alemi et al., 2016; Oord et al., 2018; Belghazi et al., 2018; Poole et al., 2019).

Although the variational approaches to MI estimation have been widely used, there are still several limitations pointed out by Song and Ermon (2019), for example, a high variance of the estimators in MINE (Belghazi et al., 2018). This will bring additional challenges for the subsequent MI optimization. To mitigate the issues, Song and Ermon (2019) proposed a unified framework over variational estimators that treat variational MI estimation as an optimization problem over density ratios. This is achieved by utilizing the role of partition function estimation with variance reduction techniques. This improved MI estimation, named by a *smoothed MI lower bound estimator*, I_{SMILE}, has been demonstrated by compared to existing variational methods in Barber and Agakov (2003); Poole et al. (2019).

2.2 Mutual Information Optimization

The BED problem is to find the optimal design that maximizes the MI estimation. Given a point-by-point base MI estimator, a variety of different approaches can be used for the subsequent optimization over designs (Amzal et al., 2006; Müller, 2005; Rainforth, 2017). At a high-level, most existing methods belong to a two-stage procedure where an MI estimator of $I_{\mathrm{MI}}(\boldsymbol{\xi})$ is first made and then followed by a separate optimization algorithm which is used to select the candidate design $\boldsymbol{\xi}$ to evaluate next. However, this two-stage framework can be computationally intensive because it needs an extra level of nesting to the optimization process in Eq. (5); in other words, $I_{\mathrm{MI}}(\boldsymbol{\xi})$ must be separately estimated for each $\boldsymbol{\xi}$, which significantly increase the overall computational cost.

Recent studies (Foster et al., 2019; Kleinegesse and Gutmann, 2019) tend to use Bayesian optimization (BO) (Snoek et al., 2012) for this purpose due to its sample efficiency, robustness and capability to naturally deal with noisy MI estimator (Foster et al., 2019). However, it is difficult for BO to scale the overall BED process to high-dimensional design settings (Foster et al., 2020; Kleinegesse and Gutmann, 2020). Broadly speaking, BO is prohibitively slow and approaches the performance of "random search" if the dimensionality of design space is Ω>100 (Snoek et al., 2015). The scalability of BO is still the major challenge, even though some recent studies (Wang et al., 2017; Li et al., 2018; Mutny and Krause, 2018; Rana et al., 2017; Li et al., 2016; Eriksson et al., 2019) are involved to make improvements.

To open the door of high-dimensional BED, Foster et al. (2019) proposed a unified gradient-based BED method but they focus on models with tractable likelihood functions or assume that the gradients are available. Kleinegesse and Gutmann (2020) proposed a MINEBED method that leverages the neural MI estimation to jointly determine the optimal design and the posterior using a gradient-based optimization algorithm. However, the MINEBED only works for the experimental design with available pathwise gradient estimators. As for the implicit models without gradient, Kleinegesse and Gutmann (2020) still consider a two-stage framework that uses BO as the outer-level optimizer at the expense of reduced scalability.

3 The SAGABED Approach

We here show how to perform a stochastic approximate gradient ascent (SAGA) method to design optimal experiments for implicit models without gradients and address the computational challenges in scaling to high-dimensional problems.

3.1 Smoothed MI Estimator

Belghazi et al. (2018) have recently proposed to estimate MI by gradient ascent over neural networks and argued that the lower bound can be tightened by

optimizing the neural network parameters. This MI estimator is typically named by MINE-f or f-GAN KL Nowozin et al. (2016)

$$I_{\text{MINE}}(\boldsymbol{\xi}, \boldsymbol{\psi}) = \mathbb{E}_{p(\boldsymbol{\theta},\boldsymbol{y}|\boldsymbol{\xi})}[\mathcal{T}_{\boldsymbol{\psi}}(\boldsymbol{\theta}, \boldsymbol{y})] - \log\left(\mathbb{E}_{p(\boldsymbol{\theta})p(\boldsymbol{y}|\boldsymbol{\xi})}\left[e^{\mathcal{T}_{\boldsymbol{\psi}}(\boldsymbol{\theta},\boldsymbol{y})}\right]\right), \tag{7}$$

where $\mathcal{T}_{\boldsymbol{\psi}}(\boldsymbol{\theta}, \boldsymbol{y})$ is a neural network that is parametrized by $\boldsymbol{\psi}$ with model parameter $\boldsymbol{\theta}$ and data $\boldsymbol{y}$ as input. Incorporating neural network parameters $\boldsymbol{\psi}$ with design parameters $\boldsymbol{\xi}$, the BED problem can be formulated by maximizing the overall objective

$$\boldsymbol{\xi}^* = \arg\max_{\boldsymbol{\xi}} \max_{\boldsymbol{\psi}} \{I_{\text{MINE}}(\boldsymbol{\xi}, \boldsymbol{\psi})\}. \tag{8}$$

The optimal design $\boldsymbol{\xi}^*$ can be obtained by maximizing the MI estimator in Eq. (7) through a joint gradient-based algorithm (Kleinegesse and Gutmann, 2020; Foster et al., 2020) or a separate gradient-free updating scheme (Foster et al., 2019; Kleinegesse and Gutmann, 2020) of $\boldsymbol{\xi}$ and $\boldsymbol{\psi}$. The accuracy, efficiency and robustness for estimating and optimizing MI therefore become very important to the BED tasks. Unfortunately, I_{MINE} exhibits variance that could grow exponentially with the ground truth MI and leads to poor bias-variance trade-offs in practice (Song and Ermon, 2019). The high variance of I_{MINE} may weaken the robustness the final optimal design $\boldsymbol{\xi}^*$ and thus cause a higher variance of the posterior distributions. We propose to use a smoothed mutual information lower-bound estimator I_{SMILE} with hyperparameter τ (Song and Ermon, 2019)

$$\begin{aligned} &I_{\text{SMILE}}(\boldsymbol{\xi}, \boldsymbol{\psi}) \\ &= \mathbb{E}_{p(\boldsymbol{\theta},\boldsymbol{y}|\boldsymbol{\xi})}[\mathcal{T}_{\boldsymbol{\psi}}(\boldsymbol{\theta}, \boldsymbol{y})] \\ &\quad - \log\left(\mathbb{E}_{p(\boldsymbol{\theta})p(\boldsymbol{y}|\boldsymbol{\xi})}[\text{clip}(e^{\mathcal{T}_{\boldsymbol{\psi}}(\boldsymbol{\theta},\boldsymbol{y})}, e^{-\tau}, e^{\tau})]\right), \end{aligned} \tag{9}$$

where the clip function is defined as

$$\text{clip}(u, v, w) = \max(\min(u, w), v). \tag{10}$$

The choice of τ affects the bias-variance trade-off: when $\tau \to \infty$, I_{SMILE} converges to I_{MINE}; with a smaller τ, the variance is reduced at the cost of increasing bias (Song and Ermon, 2019). The improved MI estimation via variance reduction techniques is benefit to the optimizing process in Eq. (8) and thus yields a robust optimal design $\boldsymbol{\xi}^*$.

3.2 Stochastic Gradient Approximate

In the context of BED for implicit models, the pathwise gradients of the MI lower bound are unavailable. A potential choice for approximating the gradient is to use Gaussian Smoothing (GS) (Nesterov and Spokoiny, 2017) method to make the function smooth. The smoothed loss is defined by

$$f_\sigma(\boldsymbol{\xi}) = \mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,\mathbf{I}_d)}\left[f(\boldsymbol{\xi} + \sigma\boldsymbol{\epsilon})\right], \tag{11}$$

where $\mathcal{N}(0, \mathbf{I}_d)$ is the d-dimensional standard Gaussian distribution, and $\sigma > 0$ is the smoothing radius. The standard GS represents the $\nabla f_\sigma(\boldsymbol{\xi})$ as an d-dimensional integral and estimate it by drawing M random samples $\{\boldsymbol{\epsilon}_i\}_{i=1}^M$ from $\mathcal{N}(0, \mathbf{I}_d)$, i.e.,

$$\begin{aligned} \nabla f_\sigma(\boldsymbol{\xi}) &= \frac{1}{\sigma}\mathbb{E}_{\boldsymbol{\epsilon}\sim\mathcal{N}(0,\mathbf{I}_d)}\left[f(\boldsymbol{\xi} + \sigma\boldsymbol{\epsilon})\,\boldsymbol{\epsilon}\right] \\ &\approx \frac{1}{M\sigma}\sum_{i=1}^{M} f(\boldsymbol{\xi} + \sigma\boldsymbol{\epsilon}_i)\boldsymbol{\epsilon}_i. \end{aligned} \tag{12}$$

This method using random MC sampling is also called Evolution Strategies (ES) which have been successfully applied in a variety of high-dimensional optimization problems, including RL tasks (Salimans et al., 2017; Choromanski et al., 2018) due to the advantages in scalability and parallelization capability. However, gradient estimator of ES tends to have a higher variance for high-dimensional space such that it requires a large number of samples to be robust (Nesterov and Spokoiny, 2017). To reduce the variance of the gradient estimator and incorporate with the smoothed MI estimator I_{SMILE}, we propose to use the Guided ES (GES) algorithm that leverages the recent advances in variance-reduced ES (Maheswaranathan et al., 2018). Specifically, GES generates a subspace by keeping track of the previous k surrogate gradients during optimization, and leverage this prior information by changing the distribution of $\boldsymbol{\epsilon}_i$ in Eq. (12) to $\mathcal{N}(0, \boldsymbol{\Sigma})$ with

$$\boldsymbol{\Sigma} = (\alpha/n) \cdot \mathbf{I}_n + (1 - \alpha)/k \cdot UU^T, \tag{13}$$

where k and n are the subspace and parameter dimensions respectively, U denotes an $n \times k$ orthonormal basis for the subspace, and α is a hyperparameter that trades off variance between the subspace and full parameter space. This improved search distribution allows a low-variance estimate of the descent direction $\nabla f_\sigma^G(\boldsymbol{\xi})$, which can then be passed to a stochastic gradient ascent (SGA) optimizer. The GES algorithm is further improved by utilizing historical estimated gradients to build a low-dimensional subspace for sampling search directions and adaptively update the importance of this subspace through a exploitation and exploration trade-off (Liu et al., 2020). To this end, the variance of the GES gradient estimator $\hat{\nabla} f_\sigma^G(\boldsymbol{\xi})$ is low and it can be naturally incorporated with I_{SMILE} to find a robust optimal design $\boldsymbol{\xi}^*$.

3.3 SAGABED Algorithm for Implicit Models

To address the grand challenges in high-dimensional BED for implicit models without gradients, we propose a novel and scalable approach, named by `SAGABED`, which integrates a smoothed MI lower bound and an approximate gradient estimator. The details of `SAGABED` algorithm is provided in Algorithm 1.

Algorithm 1: The `SAGABED` algorithm

Require: neural network architectures, learning rates ℓ_ψ and ℓ_ξ, τ in I_{SMILE}, total prior samples n, total iterations T, implicit model $\mathcal{M}$
Process:
Initialize a design $\boldsymbol{\xi}_0$ by random sampling
Initialize neural network parameter $\boldsymbol{\psi}_0$
for $t = 0 : T-1$ **do**
  Draw n samples from the prior distribution of the model parameters $\boldsymbol{\theta}$: $\boldsymbol{\theta}^{(1)}, ..., \boldsymbol{\theta}^{(n)} \sim p(\boldsymbol{\theta})$
  Compute the corresponding data samples $\boldsymbol{y}^{(i)}$, $i = 1, ..., n$ using the current design $\boldsymbol{\xi}_t$ and an implicit model $\mathcal{M}$
  Evaluate the smoothed MI lower bound I_{SMILE} by Eq. (9) at the current design $\boldsymbol{\xi}_t$ and network parameters $\boldsymbol{\psi}_t$
  Compute the approximate gradient estimator $\nabla^*_{\boldsymbol{\xi}} I_{\text{SMILE}}(\boldsymbol{\xi}_t, \boldsymbol{\psi}_t)$ using the GES algorithm
  Evaluate the gradient of the I_{SMILE} with respect to the network parameters $\nabla_{\boldsymbol{\psi}} I_{\text{SMILE}}(\boldsymbol{\xi}, \boldsymbol{\psi})$
  Update design $\boldsymbol{\xi}_t$ via gradient ascent: $\boldsymbol{\xi}_{t+1} = \boldsymbol{\xi}_t + \ell_\xi \nabla^*_{\boldsymbol{\xi}} I_{\text{SMILE}}(\boldsymbol{\xi}_t, \boldsymbol{\psi}_t)$
  Update neural network parameters $\boldsymbol{\psi}_t$ via gradient ascent: $\boldsymbol{\psi}_{t+1} = \boldsymbol{\psi}_t + \ell_\psi \nabla_{\boldsymbol{\psi}} I_{\text{SMILE}}(\boldsymbol{\xi}_t, \boldsymbol{\psi}_t)$
end for

In the following, we discuss some important features of the `SAGABED` approach, specifically for high-dimensional design problems.

- *Unified framework vs two-stage framework*
  Without the requirement of pathwise gradients for implicit models, we utilize the stochastic approximate gradients and construct a unified framework that allows the design process to be performed by a simultaneous optimization with respect to both the variational and design parameters. The existing two-stage framework that builds a pointwise MI estimator before feeding this estimator to an outer-level optimizer is often computationally intensive.

- *Scalability, portability, and parallelization*
  we proposes the stochastic approximate gradient ascent procedure that naturally avoids to the scalability issue in gradient-free methods including BO. The proposed framework can be easily incorporated with other MI estimators and implicit models because we only need the forward simulation value to approximate the gradient using guided ES algorithm. Compared to gradient-based methods, our computational cost is slightly higher but we benefit from the parallelization capability of Guided ES methods and thus improve the computational efficiency.

- *Robust estimation with a lower variance*
  The smoothed MI lower bound used in this method allows us to perform a robust MI estimation and optimization for Bayesian experimental design. The resulting low variance of the optimal design and posterior samples enable a more preciously estimate of the model parameters.

After determining the optimal design $\boldsymbol{\xi}^*$ by maximizing the MI lower bound, we can obtain an estimate of the posterior $p(\boldsymbol{\theta}|\boldsymbol{y}, \boldsymbol{\xi}^*)$ given the learned neural network $\mathcal{T}_{\boldsymbol{\psi}^*}(\boldsymbol{\theta}, \boldsymbol{y})$ and prior distribution

$$p(\boldsymbol{\theta}|\boldsymbol{y}, \boldsymbol{\xi}) = \text{clip}(e^{\mathcal{T}_{\boldsymbol{\psi}}(\boldsymbol{\theta}, \boldsymbol{y})-1}, e^{-\tau}, e^{\tau}) p(\boldsymbol{\theta}). \tag{14}$$

The relationship in Eq. (14) allows to easily generate posterior samples $\boldsymbol{\theta}_i \sim p(\boldsymbol{\theta}|\boldsymbol{y}, \boldsymbol{\xi}^*)$ using Markov chain Monte Carlo (MCMC) sampling or categorical sampling (Kleinegesse and Gutmann, 2020) since the posterior density can be quickly evaluated via Eq. (14).

4 Experiments

We here demonstrate our method in three examples: a linear model with several noises, a real-world pharmacokinetic model (Ryan et al., 2014), and a scientific quantum control problem (McMichael, 2020). We compare `SAGABED` against two baselines: the two-stage framework using BO and the gradient-based framework (if gradient is available). The first two have sampling path gradients, allowing us to compare with gradient-based methods, while the third one does not.

4.1 Noisy Linear Regression

We first show our approach through a classical linear model with noisy sources that has been used by Kleinegesse and Gutmann (2020) and Foster et al. (2020). We assume the model to be

$$\boldsymbol{y} = \theta_1 \mathbf{1} + \theta_2 \boldsymbol{\xi} + \boldsymbol{\epsilon} + \boldsymbol{\nu}, \tag{15}$$

where $\boldsymbol{y}$ is response variables, $\boldsymbol{\theta} = [\theta_1, \theta_2]^T$ are model parameters, $\boldsymbol{\epsilon} \sim \mathcal{N}(0, 1)$ and $\boldsymbol{\nu} \sim \Gamma(2, 2)$ are noise terms. The design problem is to make D measurements to better estimate the model parameters $\boldsymbol{\theta}$ by

constructing a design vector $\boldsymbol{\xi} = [\xi_1, ..., \xi_D]^T$ which consists of individual experimental design. Using the linear model in Eq (15), we can obtain the corresponding independent measurement $\boldsymbol{y}_i$, which gives a data vector $\boldsymbol{y} = [y_1, ..., y_D]^T$.

This toy example enables us to use numerical integration to approximate the posterior and MI but here we assume it is an implicit model for testing our method. Since the sampling path is given by Eq (15), we can easily compute the pathwise gradients with respect to the designs $\nabla_{\boldsymbol{\xi}} I(\boldsymbol{\xi}, \boldsymbol{\psi})$ in Kleinegesse and Gutmann (2020) and thus find the optimal designs $\boldsymbol{\xi}^* = [\xi_1^*, ..., \xi_D^*]$ using gradient-based approach, which is chosen as a baseline for comparison.

We start from a simple case with only one measurement, i.e., $D = 1$. The initial design is randomly drawn from the design domain $\boldsymbol{\xi} \in [-10, 10]$ and 10,000 samples of model parameters $\boldsymbol{\theta}$ are generated from the the prior distribution $p(\boldsymbol{\theta}) = \mathcal{N}(0, 3^2)$. To estimate the MI lower bound, we use a neural network $\mathcal{T}_{\boldsymbol{\psi}}(\boldsymbol{\theta}, \boldsymbol{y})$ with one layer of 100 neurons, with a `ReLU` activation function, as well as `Adam` optimizer with learning rates $\ell_\psi = 10^{-4}$ and $\ell_\xi = 10^{-2}$. The choice of τ in I_{SMILE} affects the bias-variance trade-off, so we compared $\tau = 1, 5, 10$ and selected the $\tau = 5$ with a smaller variance but an acceptable bias for all examples.

Figure 1 shows the MI lower bound as a function of neural network training epochs. Except for the proposed `SAGABED` method that integrates I_{SMILE} and Guided ES algorithm, there are two baselines for comparison: I_{MINE} with SGD (blue curve) and I_{MINE} with BO (green curve). When $D = 1$, three methods show a fast convergence of the MI lower bound that is around 2.6, close to a reference MI value computed by the nested MC method and the computing details can be found in the supplement material. The final optimal design found by three methods are all at the boundary, i.e., $\xi^* = 10$ or -10, which is intuitive due to a larger signal-to-noise ratio at designs for the linear model.

For more complex cases, e.g., high-dimensional design problems, we adjust the neural network architectures by hyperparameter optimization and determine to use one layer with 150 neurons for $D = 10$ and five layers with 50 neurons in each layer for D =50 and 100. We use the same activate function, learning rate, and optimizer as before. As shown in Figure 1, the final MI lower bound estimated by `SAGABED` has a bias to the reference MI values but negligibly small compared to the baselines. The gradient-based method shows a comparative performance as `SAGABED` but it is not stable and tends to collapse during training, e.g., $D =$ 100 in Figure 1, in the meantime, it might be difficult to escape from a local minimum, which may yield suboptimal results, particularly in high-dimensional design space. The method using BO is limited by its computational and scalable issues, and shows a large bias compared to the reference MI value at $D = 50$ and 100. We also demonstrate superior performance of the `SAGABED` on estimation variance that is much smaller than the baselines across all tasks shown in Figure 1.

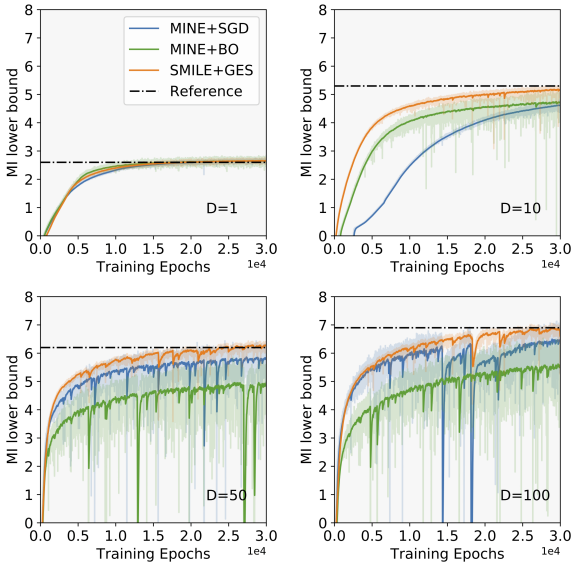

Figure 1: The MI lower bound as a function of neural network training epochs for D=1, 10, 50 and 100 measurements in noisy linear model. The dotted lines are reference MI values at optimized design t^* computed by the nested MC approach.

After finding the optimal design $\boldsymbol{\xi}^*$, we can compute the corresponding data $\boldsymbol{y}^*$ by performing a real-world experiment; then we can estimate the model parameters $\boldsymbol{\theta}$. Here we assume to have a true model parameter $\boldsymbol{\theta}_{\text{true}} = [1, 4]$ and then use it to generate $\boldsymbol{y}^*$. We can compute the posterior density using the trained neural network $\mathcal{T}_{\boldsymbol{\psi}^*}(\boldsymbol{\theta}, \boldsymbol{y})$ and Eq. (14). We therefore obtain the posterior samples from MCMC sampling and use them to estimate the model parameters. Table 1 shows the estimating mean and standard deviation of the posterior samples of the model parameters $\boldsymbol{\theta} = (\hat{\theta}_1, \hat{\theta}_2)$. The estimating error in $D = 1$ is very large because only one measurement is naturally infeasible for accurate estimation. As more measurements are taken, the posterior distribution is narrower and more accurate for $D = 50$ than for $D = 10$. However, I_{MINE} with BO due to the scalability issue and computational challenge still shows a relatively high variance at $D = 100$, while `SAGABED` and gradient-based method estimate the parameters more precisely and accurately. The resulting difference in parameter estimation exactly maps the difference in the MI lower bound estimator.

Table 1: Estimating mean and standard deviation of the posterior samples of the model parameters $\boldsymbol{\theta}$ using optimal designs $\boldsymbol{d}^*$ and real data observation $\boldsymbol{y}^*$ (use $\boldsymbol{\theta}_{\text{true}} = [1, 4]$ to generate $\boldsymbol{y}^*$)

Method	D=1		D=10		D=50		D=100	
	$\hat{\theta}_1$	$\hat{\theta}_2$	$\hat{\theta}_1$	$\hat{\theta}_2$	$\hat{\theta}_1$	$\hat{\theta}_2$	$\hat{\theta}_1$	$\hat{\theta}_2$
MINE + SGD	-1.39±2.54	6.03±**0.93**	0.51±**0.44**	2.99±0.67	1.20±0.18	3.79±0.23	0.97±0.05	4.04±0.04
MINE + BO	-1.42±**0.81**	2.98±1.19	1.22±0.58	4.93±0.91	0.71±0.25	3.66±0.40	1.35±0.21	4.79±0.26
SMILE + GES	2.76±1.36	5.74±3.08	0.83±0.56	4.69±**0.58**	1.11±**0.13**	4.25±**0.19**	1.02±**0.04**	3.98±**0.03**

4.2 Pharmacokinetic Model

Next we illustrate our approach for the design problem of determining the optimal measurements of blood sampling times for a pharmacokinetic (PK) study, which basically involves the administration of a drug to an individual and then investigate the drug distribution, absorption and elimination at certain times to analyze the underlying kinetics. The computational model introduced by Ryan et al. (2014) is used to simulate the drug concentration at a specific time. There are three governed model parameters: the volume of distribution V, the absorption rate k_a and the elimination rate k_e. The PK model is defined by a latent variable for the drug concentration at design time t,

$$z(t) = \frac{D_v}{V}\frac{k_a}{k_a - k_e}\left(e^{-k_e t} - e^{-k_a t}\right)(1+\epsilon_{1t})+\epsilon_{2t}, \quad (16)$$

where $k_a > k_e$ is a constraint, $D_v = 400$ is a single fixed dose at the beginning of the experiment, $t \in [0, 24]$ is the design domain, $\epsilon_{1t} \sim \mathcal{N}(0, 0.01)$ and $\epsilon_{2t} \sim \mathcal{N}(0, 0.1)$ are multiplicative noisy sources that are often observed in the drug data. Note that Eq. (16) is a simplified model and we assumes that the design parameters $\boldsymbol{\theta} = (V, k_a, k_e)$ are the same for the group of patients. As suggested by Ryan et al. (2014), the prior distribution for $\boldsymbol{\theta}$ is given by

$$\log(\boldsymbol{\theta}) \sim \mathcal{N}\left(\begin{bmatrix}\log 20\\ \log 1\\ \log 0.1\end{bmatrix}, \begin{bmatrix}0.05 & 0 & 0\\ 0 & 0.05 & 0\\ 0 & 0 & 0.05\end{bmatrix}\right) \quad (17)$$

In this example, we assume to take D measurements that mean we have a group of D patients but take only one blood sample from each patient at time t. Thus the design parameters are $\boldsymbol{\xi} = [t_1, ..., t_D]^T$ and the corresponding observations are $\boldsymbol{y} = [y_1, ..., y_D]^T$. It is noted that the PK model in Eq. (16) allows us to analytically derive the pathwise gradients in terms of the sampling path (Kleinegesse and Gutmann, 2020)

$$\frac{\partial z}{\partial t} = \frac{D}{V}\frac{k_a}{k_a - k_e}\left(k_a e^{-k_a t} - k_e e^{-k_e t}\right)(1 + \epsilon_{1t}), \quad (18)$$

The available gradients in Eq (18) enable to find the optimal design of blood sampling times using gradient-based methods, which can be considered as a baseline method.

Instead of the simpler case by setting $D = 1$ in the linear example, we initially set $D = 10$ here and randomly generate 10 design time in $t \in [0, 24]$ hours as the initial state. Then 10,000 model parameters samples $\boldsymbol{\theta}^{(i)}$ are drawn from the prior distribution in Eq. (17). Using these samples, we can collect the corresponding data samples $\boldsymbol{y}^{(i)}$ using the current design t. For neural network model $\mathcal{T}_{\boldsymbol{\psi}}(\boldsymbol{\theta}, \boldsymbol{y})$, we use one hidden layer of 100 neurons with `ReLU` activation function as well as `Adam` optimizer with learning rates $\ell_\psi = 10^{-4}$ and $\ell_\xi = 10^{-2}$. Hyperparameters tuning details can be found in the supplementary material.

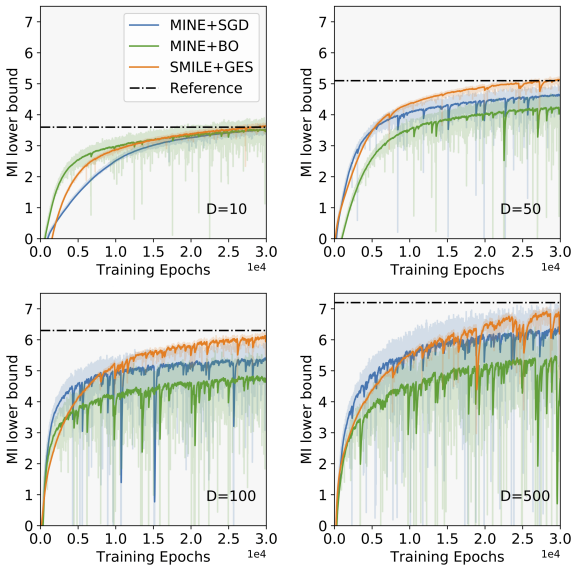

Figure 2: MI lower bound as a function of neural network training epochs for different number of measurements: D=10, 50, 100 and 500 in the study of PK model.

Figure 2 shows the MI lower bound as a function of the neural network training epochs in different sampling times $D = 10, 50, 100$ and 500 for the PK study. We compare the `SAGABED` (refers SMILE+GES in Figure 2) with the baselines that use MINE with SGD and BO optimizer respectively. When $D = 10$, the MI lower bound using `SAGABED` converges to around 3.6 that is close to a reference MI value at the optimal design t^*. The baselines show a comparative performance compared to `SAGABED`. As more measurements $D = 50$ are performed, all three methods converge to a higher MI lower bound; this is as expected since more

information are obtained from more data of model parameters. SAGABED shows a negligibly small bias compared to the reference MI value and has a low variance, which outperforms the other two baselines.

We now focus on the high dimensional cases, e.g., $D = 100$ and 500, that are more challenging settings. Here we use a 5 layered neural network with 50 neurons for each layer instead of the neural network architectures in the low dimensional cases. While SAGABED shows a larger bias of the final MI lower bound compared to the reference value due to increasing dimensions, it is still reasonably acceptable and higher than the baselines. In the high dimensional setting, the gradient-based method increases fast initially but it is easy to be trapped into local minima; the BO method has no this issue but does not scale to such high dimensionality. Also, the MI lower bound estimated by I_{MINE} shows a high variance whereas I_{SMILE} has a lower variance across all different tasks. That is an important feature that may potentially reduce the variance of final optimized design and also avoid additional efforts on optimizing hyperparameters, e.g., learning rates.

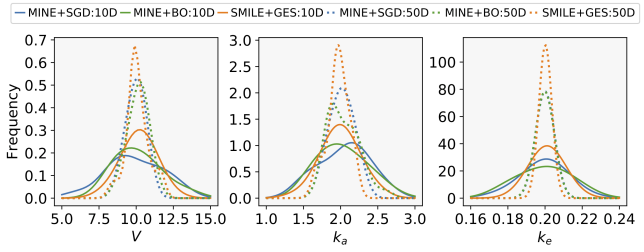

Figure 3: Marginal posterior distributions of the model parameters for $D = 10$ (solid curves) and $D = 50$ (dotted curves). Kernel density estimate is used to approximate the posterior samples. SAGABED shows a much narrower posterior distribution of the model parameters.

Once the optimal design t^* is obtained, we can conduct the experiment to generate the real-world data $\boldsymbol{y}^*$ by assuming a true model parameter $\theta_{\text{true}} = [20, 2.0, 0.2]$. Then we compute the posterior density $p(\boldsymbol{\theta}|t^*, y^*)$ using the learned neural network $\mathcal{T}_{\boldsymbol{\psi}^*}(\boldsymbol{\theta}, \boldsymbol{y})$, and obtain the posterior samples using MCMC sampling. Figure 3 shows the marginal posterior distribution of each model parameters. Using I_{SMILE}, we achieve a more accurate and much narrower estimate of all the model parameters with a lower variance than the baseline methods.

4.3 Tuning for Quantum Control

Typically, the reliable capability to manipulate qbit states is critical to quantum technology. For instance, radio-frequency pulses can be used to change of state of spin-up and spin-down, and patients benefit from MRI scans that use these approaches. In this example, we aim to conduct SAGABED to simulate a tuning process so that we can actively control the desired duration and frequency of pulses to flip electron spins in a reliable scheme. Specifically, the model parameters $\boldsymbol{\theta} = (\theta_1, \theta_2)$ are Rabi frequency and true center of resonance relative to the reference frequency, and the design parameters $\boldsymbol{\xi} = (t, \Delta f)$ are the duration time of microwave pulse and the detuning relative to a reference frequency. We here use the physical simulator developed by McMichael (2020) to simulate the tuning process for quantum control. The physical simulator used here can be interpreted as an implicit model where the gradient can not be computed exactly. We assume that our measurements are subject to Gaussian noise, $\mathcal{N}(0, 1)$. The simulation domain is discretized by 101×101 grids and a finer grid would lead to a more accurate prediction but at a higher computational cost. The design domain is set to $t \in [0, 1]$ and $\Delta f \in [-10, 10]$ and we therefore assume uniform prior distributions $p(t) = U(0, 1)$ and $p(\Delta_f) = U(-10, 10)$.

We use a one-layer neural network with 200 neurons and a ReLU activate function as well as Adam optimizer with learning rates $\ell_\psi = 10^{-3}$ and $\ell_\xi = 10^{-2}$. We still start from a simple case, i.e., only one measurement $N = 1$, and then gradually increase the number of measurements to $N = 5, 10, 50, 100$, and 500 finally. We randomly draw 10,000 uniform prior samples $\boldsymbol{\theta}^{(i)}$ and then simulate 10,000 corresponding samples of the outcome $\boldsymbol{y}$ that is the Rabi counts. The optimal design results are shown in the left column in Figure 4. Once we have done the neural network training, we assume the true model parameters $\boldsymbol{\theta}_{\text{true}} = [3.85, 1.67]$ as the true Rabi and detuning frequency. As similar to the previous examples, we can conduct a real-world measurement of the tuning control process given the optimal design $\boldsymbol{\xi}^*$ and learned neural network $\mathcal{T}_{\boldsymbol{\psi}^*}(\boldsymbol{\theta}, \boldsymbol{y})$. As a result, the posterior samples are obtained and illustrated by the right column in Figure 4.

Our main purpose in this example is to demonstrate the superior performance of SAGABED on the robustness of the optimal design and their effect on the variance of the posterior samples. As a comparison, BED with the BO method shows a similar performance on the low-dimensional cases but a high variance when the dimensionality increases to 100. Even more design are collected, the posterior distribution in the BO method is difficult to be narrowed, as shown in Figure 4. On the contrary, SAGABED demonstrates a faster narrowing rate to the true model parameters and it displays a much lower variance of the posterior samples in the high-dimensional design problems. This also illustrates the significant advantages of SAGABED in addressing the scalability challenges compared to the BO method.

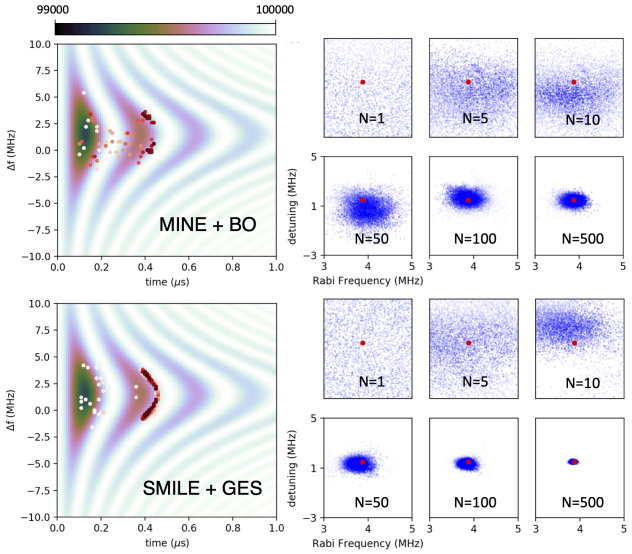

Figure 4: Performance comparison between `SAGABED` and BO method for tuning quantum pulse. The contour image (left column) shows the model photon counts for optically detected spin manipulation for pulse duration and amounts of detuning from the spin's natural resonance frequency. The right column displays the evolution of the posterior distribution with the number of designed measurements. The red points are the true model parameters.

5 Conclusion

In this paper, we develop a general unified framework that utilizes the stochastic approximate gradient for BED with implicit models. Without the requirement or assumption of pathwise gradients, our approach allows the optimization to be carried out by stochastic gradient ascent algorithms and therefore scaled to substantial high dimensional design problems. Several experiments demonstrate that our approach outperforms the baseline methods, and significantly improves the scalability of BED in high dimensional settings.

The future work will focus on the extension of our proposed framework to sequential Bayesian experimental design (SBED) that involves an iterative update of the prior and posterior distribution. We plan to improve the computational efficiency of the `SAGABED`, which may play a more critical role in the SBED scenario. We are also interested in improving the stochastic approximate gradient methods and enhancing the global exploration capability to escape the sub-optimal design caused by the issue of local minima in current SGA algorithms, e.g., standard and guided ES (Zhang et al., 2020, 2021).

Acknowledgements

This work was supported by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research (ASCR), Applied Mathematics program; and by the Artificial Intelligence Initiative at the Oak Ridge National Laboratory (ORNL). This work used resources of the Oak Ridge Leadership Computing Facility, which is supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC05-00OR22725.

References

Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. Deep variational information bottleneck. *arXiv preprint arXiv:1612.00410*, 2016.

Billy Amzal, Frédéric Y Bois, Eric Parent, and Christian P Robert. Bayesian-optimal design via interacting particle systems. *Journal of the American Statistical association*, 101(474):773–785, 2006.

Ziqiao Ao and Jinglai Li. An approximate {KLD} based experimental design for models with intractable likelihoods. *arXiv preprint arXiv:2004.00715*, 2020.

David Barber and Felix V Agakov. The im algorithm: a variational approach to information maximization. In *Advances in neural information processing systems*, page None, 2003.

Mohamed Ishmael Belghazi, Aristide Baratin, Sai Rajeswar, Sherjil Ozair, Yoshua Bengio, Aaron Courville, and R Devon Hjelm. Mine: mutual information neural estimation. *arXiv preprint arXiv:1801.04062*, 2018.

Kathryn Chaloner and Isabella Verdinelli. Bayesian experimental design: A review. *Statistical Science*, pages 273–304, 1995.

Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Advances in neural information processing systems*, pages 2172–2180, 2016.

Krzysztof Choromanski, Mark Rowland, Vikas Sindhwani, Richard E Turner, and Adrian Weller. Structured evolution with compact architectures for scalable policy optimization. *arXiv preprint arXiv:1804.02395*, 2018.

Christopher C Drovandi, Minh-Ngoc Tran, et al. Improving the efficiency of fully Bayesian optimal design of experiments using randomised quasi-Monte Carlo. *Bayesian Analysis*, 13(1):139–162, 2018.

David Eriksson, Michael Pearce, Jacob Gardner, Ryan D Turner, and Matthias Poloczek. Scalable global optimization via local Bayesian optimization. In *Advances in Neural Information Processing Systems*, pages 5496–5507, 2019.

Adam Foster, Martin Jankowiak, Elias Bingham, Paul Horsfall, Yee Whye Teh, Thomas Rainforth, and

Noah Goodman. Variational Bayesian optimal experimental design. In *Advances in Neural Information Processing Systems*, pages 14036–14047, 2019.

Adam Foster, Martin Jankowiak, Matthew O'Meara, Yee Whye Teh, and Tom Rainforth. A unified stochastic gradient approach to designing Bayesian-optimal experiments. In *International Conference on Artificial Intelligence and Statistics*, pages 2959–2969. PMLR, 2020.

Sophie Harbisher, Colin S Gillespie, and Dennis Prangle. Bayesian optimal design using stochastic gradient optimisation and Fisher information gain. *arXiv preprint arXiv:1904.05703*, 2019.

Steven Kleinegesse and Michael U Gutmann. Efficient Bayesian experimental design for implicit models. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 476–485, 2019.

Steven Kleinegesse and Michael U Gutmann. Bayesian experimental design for implicit models by mutual information neural estimation. *arXiv preprint arXiv:2002.08129*, 2020.

Steven Kleinegesse, Christopher Drovandi, and Michael U Gutmann. Sequential Bayesian experimental design for implicit models via mutual information. *arXiv preprint arXiv:2003.09379*, 2020.

Cheng Li, Sunil Gupta, Santu Rana, Vu Nguyen, Svetha Venkatesh, and Alistair Shilton. High dimensional Bayesian optimization using dropout. *arXiv preprint arXiv:1802.05400*, 2018.

Chun-Liang Li, Kirthevasan Kandasamy, Barnabás Póczos, and Jeff Schneider. High dimensional Bayesian optimization via restricted projection pursuit models. In *Artificial Intelligence and Statistics*, pages 884–892, 2016.

Fei-Yu Liu, Zi-Niu Li, and Chao Qian. Self-guided evolution strategies with historical estimated gradients. *IJCAI*, 2020.

Niru Maheswaranathan, Luke Metz, George Tucker, Dami Choi, and Jascha Sohl-Dickstein. Guided evolutionary strategies: escaping the curse of dimensionality in random search. 2018.

David McAllester and Karl Stratos. Formal limitations on the measurement of mutual information. In *International Conference on Artificial Intelligence and Statistics*, pages 875–884, 2020.

R. D. McMichael. Optimal Bayesian experimental design, 2020. URL https://github.com/usnistgov/optbayesexpt.

Peter Müller. Simulation based optimal design. *Handbook of Statistics*, 25:509–518, 2005.

Mojmir Mutny and Andreas Krause. Efficient high dimensional Bayesian optimization with additivity and quadrature fourier features. In *Advances in Neural Information Processing Systems*, pages 9005–9016, 2018.

Jay I Myung, Daniel R Cavagnaro, and Mark A Pitt. A tutorial on adaptive design optimization. *Journal of mathematical psychology*, 57(3-4):53–67, 2013.

Yurii Nesterov and Vladimir Spokoiny. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17(2):527–566, 2017.

XuanLong Nguyen, Martin J Wainwright, and Michael I Jordan. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 56(11): 5847–5861, 2010.

Sebastian Nowozin, Botond Cseke, and Ryota Tomioka. f-gan: Training generative neural samplers using variational divergence minimization. In *Advances in neural information processing systems*, pages 271–279, 2016.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Antony Overstall, James McGree, et al. Bayesian design of experiments for intractable likelihood models using coupled auxiliary models and multivariate emulation. *Bayesian Analysis*, 2018.

Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 16–17, 2017.

Ben Poole, Sherjil Ozair, Aaron van den Oord, Alexander A Alemi, and George Tucker. On variational bounds of mutual information. *arXiv preprint arXiv:1905.06922*, 2019.

Thomas William Gamlen Rainforth. *Automating inference, learning, and design using probabilistic programming*. PhD thesis, University of Oxford, 2017.

Tom Rainforth, Rob Cornish, Hongseok Yang, Andrew Warrington, and Frank Wood. On nesting Monte Carlo estimators. In *International Conference on Machine Learning*, pages 4267–4276. PMLR, 2018.

Santu Rana, Cheng Li, Sunil Gupta, Vu Nguyen, and Svetha Venkatesh. High dimensional Bayesian optimization with elastic Gaussian process. In *International Conference on Machine Learning*, pages 2883–2891, 2017.

Elizabeth G Ryan, Christopher C Drovandi, M Helen Thompson, and Anthony N Pettitt. Towards Bayesian experimental design for nonlinear models that require a large number of sampling times.

Computational Statistics & Data Analysis, 70:45–60, 2014.

Tim Salimans, Jonathan Ho, Xi Chen, Szymon Sidor, and Ilya Sutskever. Evolution strategies as a scalable alternative to reinforcement learning. *arXiv preprint arXiv:1703.03864*, 2017.

Paola Sebastiani and Henry P Wynn. Maximum entropy sampling and optimal Bayesian experimental design. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 62(1):145–157, 2000.

Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical Bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, pages 2951–2959, 2012.

Jasper Snoek, Oren Rippel, Kevin Swersky, Ryan Kiros, Nadathur Satish, Narayanan Sundaram, Mostofa Patwary, Mr Prabhat, and Ryan Adams. Scalable Bayesian optimization using deep neural networks. In *International conference on machine learning*, pages 2171–2180, 2015.

Jiaming Song and Stefano Ermon. Understanding the limitations of variational mutual information estimators. *arXiv preprint arXiv:1910.06222*, 2019.

Zi Wang, Chengtao Li, Stefanie Jegelka, and Pushmeet Kohli. Batched high-dimensional Bayesian optimization via structural kernel learning. *arXiv preprint arXiv:1703.01973*, 2017.

Jiaxin Zhang, Hoang Tran, Dan Lu, and Guannan Zhang. A novel evolution strategy with directional Gaussian smoothing for blackbox optimization. *arXiv preprint arXiv:2002.03001*, 2020.

Jiaxin Zhang, Sirui Bi, and Guannan Zhang. A directional Gaussian smoothing optimization method for computational inverse design in nanophotonics. *Materials & Design*, 197:109213, 2021.