# Variational Bayesian Optimal Experimental Design with Normalizing Flows

Jiayuan Dong[a,*], Christian Jacobsen[b], Mehdi Khalloufi[a,d], Maryam Akram[c], Wanjiao Liu[c], Karthik Duraisamy[b], Xun Huan[a]

[a]*Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI, 48109, USA*
[b]*Department of Aerospace Engineering, University of Michigan, Ann Arbor, MI, 48109, USA*
[c]*Ford Research & Innovation Center, Dearborn, MI, 48121, USA*
[d]*The Dow Chemical Company, Core R&D, Engineering and Process Science, Lake Jackson, TX 77566, USA*



**Abstract**

Bayesian optimal experimental design (OED) seeks experiments that maximize the expected information gain (EIG) in model parameters. Directly estimating the EIG using nested Monte Carlo is computationally expensive and requires an explicit likelihood. Variational OED (vOED), in contrast, estimates a lower bound of the EIG without likelihood evaluations by approximating the posterior distributions with variational forms, and then tightens the bound by optimizing its variational parameters. We introduce the use of normalizing flows (NFs) for representing variational distributions in vOED; we call this approach vOED-NFs. Specifically, we adopt NFs with a conditional invertible neural network architecture built from compositions of coupling layers, and enhanced with a summary network for data dimension reduction. We present Monte Carlo estimators to the lower bound along with gradient expressions to enable a gradient-based simultaneous optimization of the variational parameters and the design variables. The vOED-NFs algorithm is then validated in two benchmark problems, and demonstrated on a partial differential equation-governed application of cathodic electrophoretic deposition and an implicit likelihood case with stochastic modeling of aphid population. The findings suggest that a composition of 4–5 coupling layers is able to achieve lower EIG estimation bias, under a fixed budget of forward model runs, compared to previous approaches. The resulting NFs produce approximate posteriors that agree well with the true posteriors, able to capture non-Gaussian and multi-modal features effectively.

*Keywords:* uncertainty quantification, expected information gain, information lower bound, variational inference, normalizing flows, conditional invertible neural networks, coupling layers, implicit likelihood

## 1. Introduction

Mathematical models are central for understanding and making predictions in physical systems. Parameters in models are often uncertain, but their uncertainty can be reduced when information is gained from experimental observations. Since experiments are costly in many applications, optimal experimental design (OED) (see, e.g., [1, 2, 3, 4, 5]) can bring substantial resource savings by identifying the experiments that produce the most valuable data.

Bayesian OED further incorporates the prior and posterior uncertainty in its objective function. The seminal paper by Lindley [6] proposed to use mutual information (MI) between model parameters and experimental data as the design criterion, which is equivalent to the expected information gain (EIG) in the model parameters. The EIG, however, is generally intractable to compute and has to be approximated numerically. One strategy entails directly approximating the EIG. For example, Ryan [7] introduced a

* Corresponding author
*Email addresses:* jiayuand@umich.edu (Jiayuan Dong), csjacobs@umich.edu (Christian Jacobsen), khalloufi.mehdi@gmail.com (Mehdi Khalloufi), makram13@ford.com (Maryam Akram), lwanjiao@ford.com (Wanjiao Liu), kdur@umich.edu (Karthik Duraisamy), xhuan@umich.edu (Xun Huan)

nested Monte Carlo (NMC) estimator for the EIG, but it is computationally expensive with the number of likelihood evaluations scaling as the product of sample sizes of the nested Monte Carlo loops. Advanced numerics have been developed to accelerate NMC computations through surrogate modeling [8, 9], reusing samples across nested loops [8], and 'Gaussianizing' the posterior via Laplace approximations [10, 11, 12, 13]. Another estimator similar to the NMC is the prior contrastive estimator (PCE) [14]. A key difference is that PCE reuses each outer loop sample once in the inner loop, resulting the estimator mean to be negatively biased (i.e., a lower bound of the EIG). The above techniques, however, all require the ability to evaluate the likelihood function, and cannot accommodate implicit or intractable likelihood settings that often arise from stochastic models. An EIG lower bound that is able to handle intractable likelihood is the LB-KLD [15], which can be derived from Shannon's entropy power inequality.

Another strategy is to adopt a variational bound for the EIG [16], and tighten the bound by optimizing its variational parameters. (Note that PCE and LBKLD do not have variational parameters for tightening the bounds, and therefore are not considered to be variational bounds.) We call these variational OED (vOED) approaches. For example, the tractable unnormalized Barber–Agakov (TUBA) lower bound [16] incorporates the tuning of a 'critic' function, and the Nguyen–Wainwright–Jordan (NWJ) bound [17], also known as the mutual information neural estimation f-divergence (MINE-f) bound [18], can be shown to be a special case of TUBA and has been used by Kleinegesse and Gutmann [19] in the OED context. Another lower bound, the information noise-contrastive estimation (InfoNCE) bound [20], takes a similar form as PCE, but replaces the likelihood with an exponentiated critic function. More recently, Ivanova *et al.* [21] extended the NWJ and InfoNCE bounds to the sequential OED setting. All of these bounds can accommodate implicit likelihood.

A particular type of variational bounds, known as the Barber–Agakov (BA) lower bound [22], emerges from approximating the posterior density function directly. Foster *et al.* [23] were the first to use the BA bound in the OED setting, but remained with relatively simple variational distributions such as Gaussian, Bernoulli, and Gamma. The approximations thus can deteriorate when the true posterior distributions depart from these forms. Improving upon their work, Foster *et al.* [14] then introduced the adaptive contrastive estimation (ACE) lower bound based on the PCE but used a variational biasing distribution for the inner loop sampling. However, ACE required explicit likelihood.

One of the key contributions of this paper is therefore to reduce the BA bound bias in estimating the EIG by improving the accuracy of posterior approximation through enriching the space of variational distributions using normalizing flows (NFs) [24, 25, 26, 27]. We achieve this while maintaining the ability to handle implicit likelihood.

NFs are rooted in measure transport theory [28, 29, 30] which seek an invertible mapping between a target distribution (e.g., the posterior) and a reference distribution (e.g., a standard normal). Once such a transport map is established, the density function of the target distribution can be obtained from the density of the reference, and vice versa, through the standard change-of-variable formula. Hence, an approximate posterior can be represented by a parameterized transport map, and the problem becomes finding the best transport map from its parametric class. In vOED, 'best' is meant by maximizing the lower bound; in the related field of expectation propagation [31] (a relative of variational inference but uses the 'reverse' KL divergence), 'best' is meant by minimizing the Kullback–Lebiler (KL) divergence from the approximate posterior to the true posterior—we will show that these two notions are equivalent in expectation.

Different structures and parameterizations of transport maps have been proposed. Triangular and block-triangular forms [32], often parameterized using multivariate polynomials [33, 29], radial basis functions [29], tensor decomposition [34], and partially convex potential maps and conditional optimal transport flows [35]. Invertibility is then enforced through local monotonicity constraints, or setting parameterizations in ways to guarantee monotonicity [29, 35]. For instance, Baptista *et al.* [36] proposed a rectification operator that can always transform sufficiently smooth non-monotone functions into monotone component functions of a triangular map, and also demonstrated the elimination of spurious local minima that often arises in the training of neural network-based maps. Furthermore, the triangular structures provide a convenient means to extract conditional densities (e.g., the posterior or likelihood) and to compute the Jacobian of transformation.

Elsewhere, efforts in the machine learning community in creating practical transport maps focused on

composing together invertible, often neural network-based, functions, known as normalizing flows. For recent reviews of NFs, see [27, 26, 26]. While first represented as a simple 3-trainable-parameter function [37] and then as a composition of localized radial expansion to achieve greater expressiveness [38], the architectural form of NFs has evolved rapidly. Examples include planar and radial flows [25, 39, 40, 41], coupling layers and autoregressive flows [42, 24, 43, 44, 45], splines [46], residual flows [47] and neural ordinary differential equations [48]. While NFs have been used for approximating the posterior in a variational inference context ([25] and many subsequent works), they have not yet been explored in OED.

In this paper, we propose and investigate the use of NFs for approximating posterior distributions *en route* to optimizing the BA bound estimates of the EIG for vOED; we call this approach **vOED-NFs**. The key novelty and contributions of this paper are as follows.

- We present the use of NFs for estimating the EIG lower bound in vOED, and illustrate its sampling efficiency (lower bias) compared to previous approaches.
- We show the ability of NFs trained for vOED to achieve good posterior approximations, including non-Gaussian and multi-modal distributions.
- We validate vOED-NFs against established reference methods, and demonstrate vOED-NFs on a partial differential equation (PDE)-governed application of cathodic electrophoretic deposition.
- We illustrate vOED-NFs in handling an implicit likelihood case with stochastic modeling of aphid population.

This paper is structured as follows. Section 2 reviews the OED and vOED problem formulations. Section 3 illustrates the computational methods for solving the vOED problem. Section 4 then presents a number of numerical experiments, starting with simple benchmarks to validate vOED-NFs against existing established approaches, and then demonstrating it on a PDE-governed design problem for cathodic electrophoretic deposition (e-coating) studied in the automotive industry and an implicit case for aphid population modeling. The paper ends with conclusions and future work in Sec. 5.

## 2. Problem Formulation

### *2.1. Bayesian optimal experimental design*

We adopt the following notation: upper case for random variable, lower case for realization, bold for vector or matrix, and subscript in a probability density function (PDF) is generally omitted but retained in some cases for clarification or emphasis; for example, $p_{\mathbf{X}}(\mathbf{X}=\mathbf{x}) = p(\mathbf{x})$ denotes the PDF of random vector $\mathbf{X}$ evaluated at value $\mathbf{X}=\mathbf{x}$. When an experiment is conducted under design $\mathbf{d} \in \mathcal{D} \subseteq \mathbb{R}^{n_d}$ and yields an observation $\mathbf{Y}=\mathbf{y} \in \mathbb{R}^{n_y}$, the PDF for the model parameters $\boldsymbol{\Theta} \in \mathbb{R}^{n_\theta}$ is updated via Bayes' rule by

$$p(\boldsymbol{\theta}|\mathbf{y},\mathbf{d}) = \frac{p(\boldsymbol{\theta})\, p(\mathbf{y}|\boldsymbol{\theta},\mathbf{d})}{p(\mathbf{y}|\mathbf{d})}, \tag{1}$$

where $p(\boldsymbol{\theta})$ is the prior PDF (we assume the prior does not depend on $\mathbf{d}$), $p(\mathbf{y}|\boldsymbol{\theta},\mathbf{d})$ is the data likelihood, $p(\mathbf{y}|\mathbf{d})$ is the marginal likelihood (model evidence), and $p(\boldsymbol{\theta}|\mathbf{y},\mathbf{d})$ is the posterior PDF. The likelihood can be computed based on an underlying observation model, for example,

$$\mathbf{Y} = \mathbf{G}(\boldsymbol{\Theta},\mathbf{d}) + \boldsymbol{\mathcal{E}}, \tag{2}$$

where $\mathbf{G}: \mathbb{R}^{n_\theta} \times \mathbb{R}^{n_d} \to \mathbb{R}^{n_y}$ is the forward model (e.g., from solving a governing system of PDEs) and $\boldsymbol{\mathcal{E}}$ is the random noise associated with the observation model. Each likelihood evaluation thus entails computing $p(\mathbf{y}|\boldsymbol{\theta},\mathbf{d}) = p_{\boldsymbol{\mathcal{E}}}(\mathbf{y} - \mathbf{G}(\boldsymbol{\theta},\mathbf{d}))$, which requires a forward model solve.

We adopt the EIG on $\boldsymbol{\Theta}$ to reflect the expected utility of performing an experiment. Mathematically, the EIG is the expected KL divergence from the parameter prior to the posterior, and can be interpreted as the expected parameter uncertainty reduction due to the experiment:

$$U(\mathbf{d}) = \mathbb{E}_{\mathbf{Y}|\mathbf{d}}\left[D_{\mathrm{KL}}(\, p_{\boldsymbol{\Theta}|\mathbf{Y},\mathbf{d}} \,||\, p_{\boldsymbol{\Theta}}\,)\right] = \iint p(\boldsymbol{\theta},\mathbf{y}|\mathbf{d}) \ln\left[\frac{p(\boldsymbol{\theta}|\mathbf{y},\mathbf{d})}{p(\boldsymbol{\theta})}\right] d\boldsymbol{\theta}\, d\mathbf{y}. \tag{3}$$

The Bayesian OED problem then entails solving for the optimal design:

$$\mathbf{d}^* = \underset{\mathbf{d}\in\mathcal{D}}{\operatorname{argmax}}\, U(\mathbf{d}). \tag{4}$$

*2.2. Variational optimal experimental design*

We summarize below the variational OED (vOED) approach introduced by Foster *et al.* [23]. vOED replaces the posterior or marginal likelihood with an approximate PDF $q(\cdot)$ parameterized by $\boldsymbol{\lambda} \in \mathbb{R}^{n_{\boldsymbol{\lambda}}}$. When approximating the posterior, the expected utility from Eqn. (3) becomes

$$U_L(\mathbf{d};\boldsymbol{\lambda}) = \mathbb{E}_{\boldsymbol{\Theta},\mathbf{Y}|\mathbf{d}}\left[\ln\frac{q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda})}{p(\boldsymbol{\theta})}\right] = \iint p(\boldsymbol{\theta},\mathbf{y}|\mathbf{d}) \ln\left[\frac{q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda})}{p(\boldsymbol{\theta})}\right] d\boldsymbol{\theta}\, d\mathbf{y}. \tag{5}$$

Note that the outer expectation is still taken with respect to the true distribution $p(\boldsymbol{\theta},\mathbf{y}|\mathbf{d})$. $U_L(\mathbf{d};\boldsymbol{\lambda})$ is known as the Barber–Agakov bound [22]; it is a lower bound for $U(\mathbf{d})$:

$$U(\mathbf{d}) - U_L(\mathbf{d};\boldsymbol{\lambda}) = \mathbb{E}_{\mathbf{Y}|\mathbf{d}}[D_{\mathrm{KL}}(\, p_{\boldsymbol{\Theta}|\mathbf{Y},\mathbf{d}} \,||\, q_{\boldsymbol{\Theta}|\mathbf{Y};\boldsymbol{\lambda}}\,)] \geq 0. \tag{6}$$

The bound is tight if and only if $q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda}) = p(\boldsymbol{\theta}|\mathbf{y},\mathbf{d})$ for all $\boldsymbol{\theta}$ and $\mathbf{y}$. The best (tightest) lower bound at a given $\mathbf{d}$ is then

$$U_L(\mathbf{d};\boldsymbol{\lambda}^*) = \max_{\boldsymbol{\lambda}} U_L(\mathbf{d};\boldsymbol{\lambda}). \tag{7}$$

We note that the ordering of posterior approximation quality for $q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda})$ as measured by $U_L(\mathbf{d};\boldsymbol{\lambda})$ is preserved when measured by the expected KL divergence in the form of moment projection as used in expectation propagation [31] (i.e., the 'reverse' KL divergence compared to the information projection form used in variational inference). Hence, the approximate posterior resulting from a tighter lower bound is also closer to the true posterior in this expected KL sense. We make this precise in the following proposition.

*Proposition* 1. Consider probability densities $q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda}_1)$ and $q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda}_2)$ formed at variational parameter values $\boldsymbol{\lambda}_1$ and $\boldsymbol{\lambda}_2$, respectively, both as approximations to the true posterior density $p(\boldsymbol{\theta}|\mathbf{y},\mathbf{d})$. Then,

$$U_L(\mathbf{d};\boldsymbol{\lambda}_1) \leq U_L(\mathbf{d};\boldsymbol{\lambda}_2) \tag{8}$$

if and only if

$$\mathbb{E}_{\mathbf{Y}|\mathbf{d}}\left[D_{\mathrm{KL}}(p_{\boldsymbol{\theta}|\mathbf{y},\mathbf{d}} \,||\, q_{\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda}_1})\right] \geq \mathbb{E}_{\mathbf{Y}|\mathbf{d}}\left[D_{\mathrm{KL}}(p_{\boldsymbol{\theta}|\mathbf{y},\mathbf{d}} \,||\, q_{\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda}_2})\right]. \tag{9}$$

A proof of the proposition is provided in Appendix A.

With $U_L(\mathbf{d};\boldsymbol{\lambda}^*)$ being the best variational approximation to the true EIG at $\mathbf{d}$, the vOED problem looks for the design that maximizes the tightest bound by simultaneously optimizing for $\mathbf{d}$ and $\boldsymbol{\lambda}$:

$$\mathbf{d}^*, \boldsymbol{\lambda}^* = \underset{\mathbf{d}\in\mathcal{D},\boldsymbol{\lambda}}{\operatorname{argmax}}\, U_L(\mathbf{d};\boldsymbol{\lambda}). \tag{10}$$

When $U_L(\mathbf{d};\boldsymbol{\lambda})$ is differentiate with respect to $\mathbf{d}$ and $\boldsymbol{\lambda}$, their gradients are

$$\nabla_{\boldsymbol{\lambda}} U_L(\mathbf{d};\boldsymbol{\lambda}) = \mathbb{E}_{\boldsymbol{\Theta}}\{\mathbb{E}_{\mathbf{Y}|\boldsymbol{\Theta},\mathbf{d}}[\nabla_{\boldsymbol{\lambda}} \ln q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda})]\}, \tag{11}$$

$$\nabla_{\mathbf{d}} U_L(\mathbf{d};\boldsymbol{\lambda}) = \mathbb{E}_{\boldsymbol{\Theta}}\{\nabla_{\mathbf{d}} \mathbb{E}_{\mathbf{Y}|\boldsymbol{\Theta},\mathbf{d}}[\ln q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda})]\} = \mathbb{E}_{\boldsymbol{\Theta}}\left\{\int [\nabla_{\mathbf{d}} p(\mathbf{y}|\boldsymbol{\theta},\mathbf{d})] \ln q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda})\right\}$$
$$= \mathbb{E}_{\boldsymbol{\Theta}}\{\mathbb{E}_{\mathbf{Y}|\boldsymbol{\Theta},\mathbf{d}}[\ln q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda}) \nabla_{\mathbf{d}} \ln p(\mathbf{y}|\boldsymbol{\theta},\mathbf{d})]\}, \tag{12}$$

where the last equality uses the identity $\nabla_{\mathbf{d}} \ln p(\mathbf{y}|\boldsymbol{\theta},\mathbf{d}) = \frac{\nabla_{\mathbf{d}} p(\mathbf{y}|\boldsymbol{\theta},\mathbf{d})}{p(\mathbf{y}|\boldsymbol{\theta},\mathbf{d})}$. Access to these gradient formulas allows the adoption of gradient-based optimization algorithms to solve Eqn. (10).

Alternatively, one can approximate the marginal likelihood instead of the posterior as introduced in [23], and the expected utility becomes

$$U_U(\mathbf{d};\boldsymbol{\lambda}) = \mathbb{E}_{\boldsymbol{\Theta},\mathbf{Y}|\mathbf{d}}\left[\ln \frac{p(\mathbf{y}|\boldsymbol{\theta},\mathbf{d})}{q(\mathbf{y};\boldsymbol{\lambda})}\right] = \iint p(\boldsymbol{\theta},\mathbf{y}|\mathbf{d}) \ln\left[\frac{p(\mathbf{y}|\boldsymbol{\theta},\mathbf{d})}{q(\mathbf{y};\boldsymbol{\lambda})}\right] d\boldsymbol{\theta}\, d\mathbf{y}, \tag{13}$$

where the posterior-to-prior density ratio has been replaced with likelihood-to-marginal-likelihood density ratio via Bayes' rule in Eqn. (1). Again, the outer expectation remains with respect to the true distribution $p(\boldsymbol{\theta},\mathbf{y}|\mathbf{d})$. $U_U$ forms an upper bound for $U(\mathbf{d})$:

$$U_U(\mathbf{d};\boldsymbol{\lambda}) - U(\mathbf{d}) = D_{\mathrm{KL}}(\, p_{\mathbf{y}|\mathbf{d}} \,||\, q_{\mathbf{y};\boldsymbol{\lambda}}\,) \geq 0. \tag{14}$$

The bound is tight if and only if $q(\mathbf{y};\boldsymbol{\lambda}) = p(\mathbf{y}|\mathbf{d})$ for all $\mathbf{y}$. The best (tightest) upper bound at a given $\mathbf{d}$ is then

$$U_U(\mathbf{d};\boldsymbol{\lambda}^*) = \min_{\boldsymbol{\lambda}} U_U(\mathbf{d};\boldsymbol{\lambda}). \tag{15}$$

The corresponding vOED problem using $U_U(\mathbf{d};\boldsymbol{\lambda})$ involves solving for

$$\mathbf{d}^* = \operatorname*{argmax}_{\mathbf{d}\in\mathcal{D}} \min_{\boldsymbol{\lambda}} U_U(\mathbf{d};\boldsymbol{\lambda}). \tag{16}$$

When $U_U(\mathbf{d};\boldsymbol{\lambda})$ is differentiate with respect to $\boldsymbol{\lambda}$, its gradient is

$$\nabla_{\boldsymbol{\lambda}} U_U(\mathbf{d};\boldsymbol{\lambda}) = -\mathbb{E}_{\mathbf{Y}|\mathbf{d}}[\nabla_{\boldsymbol{\lambda}} \ln q(\mathbf{y};\boldsymbol{\lambda})]. \tag{17}$$

The use of $U_U(\mathbf{d};\boldsymbol{\lambda})$ can offer advantages when $n_y \ll n_\theta$ so that the variational density approximation is applied to a lower-dimensional space. However, the use of $U_U(\mathbf{d};\boldsymbol{\lambda})$ involves solving a more difficult maximin saddle-point problem in Eqn. (16), and its two optimizations cannot be combined like in Eqn. (10). In this paper, we will primarily focus on the lower bound $U_L(\mathbf{d};\boldsymbol{\lambda})$ and its corresponding vOED problem in Eqn. (10).

## 3. Computational Methods

The expected utility in Eqn. (3) is generally intractable and needs be estimated numerically. One approach is to use the NMC estimator [7]:

$$U(\mathbf{d}) \approx \widehat{U}_{\mathrm{NMC}}(\mathbf{d}) \coloneqq \frac{1}{N_{\mathrm{out}}} \sum_{i=1}^{N_{\mathrm{out}}} \left\{ \ln p(\mathbf{y}^{(i)}|\boldsymbol{\theta}^{(i)},\mathbf{d}) - \ln\left[\frac{1}{N_{\mathrm{in}}} \sum_{j=1}^{N_{\mathrm{in}}} p(\mathbf{y}^{(i)}|\boldsymbol{\theta}^{(i,j)},\mathbf{d})\right]\right\}, \tag{18}$$

where samples $\boldsymbol{\theta}^{(i)} \sim p(\boldsymbol{\theta})$, $\mathbf{y}^{(i)} \sim p(\mathbf{y}|\boldsymbol{\theta}^{(i)},\mathbf{d})$, and $\boldsymbol{\theta}^{(i,j)} \sim p(\boldsymbol{\theta})$ are drawn from the prior and likelihood. Calculating the NMC estimator $\widehat{U}_{\mathrm{NMC}}$ requires evaluating the likelihood $p(\mathbf{y}|\boldsymbol{\theta},\mathbf{d})$ and cannot directly handle implicit likelihood settings. $\widehat{U}_{\mathrm{NMC}}$ is a biased estimator to $U$ under any finite inner-loop sample size $N_{\mathrm{in}}$, but it is asymptotically unbiased as $N_{\mathrm{in}} \to \infty$. Properties of the NMC estimator has been studied extensively [7, 49, 13, 50], and we will use it for reference comparison in this work when applicable.

For vOED, the lower and upper bounds in Eqn. (5) and (13) require numerical approximations for their expectation operators, for example through standard MC estimation:

$$U_L(\mathbf{d};\boldsymbol{\lambda}) \approx \widehat{U}_L(\mathbf{d};\boldsymbol{\lambda}) \coloneqq \frac{1}{N}\sum_{i=1}^{N}\left\{\ln q(\boldsymbol{\theta}^{(i)}|\mathbf{y}^{(i)};\boldsymbol{\lambda}) - \ln p(\boldsymbol{\theta}^{(i)})\right\}, \tag{19}$$

$$U_U(\mathbf{d};\boldsymbol{\lambda}) \approx \widehat{U}_U(\mathbf{d};\boldsymbol{\lambda}) \coloneqq \frac{1}{N}\sum_{i=1}^{N}\left\{\ln p(\mathbf{y}^{(i)}|\boldsymbol{\theta}^{(i)},\mathbf{d}) - \ln q(\mathbf{y}^{(i)};\boldsymbol{\lambda})\right\}, \tag{20}$$

where $\boldsymbol{\theta}^{(i)} \sim p(\boldsymbol{\theta})$ and $\mathbf{y}^{(i)} \sim p(\mathbf{y}|\boldsymbol{\theta}^{(i)},\mathbf{d})$. In particular, calculating Eqn. (19) only requires sampling the likelihood without any likelihood evaluations; it is therefore suitable to handle implicit likelihood cases. Note that both $\widehat{U}_L$ and $\widehat{U}_U$ are not bounds themselves, but are (unbiased) estimators of the bounds $U_L$ and $U_U$, respectively. With respect to the true EIG $U$, then, $\widehat{U}_L$ and $\widehat{U}_U$ are biased estimators whose bias are, respectively, negative and positive.

We can also form MC estimators to the lower bound gradients in Eqn. (11) and (12), yielding

$$\widehat{\nabla_{\boldsymbol{\lambda}} U_L}(\mathbf{d};\boldsymbol{\lambda}) \coloneqq \frac{1}{N_{\text{batch}}}\sum_{i=1}^{N_{\text{batch}}} \nabla_{\boldsymbol{\lambda}} \ln q(\boldsymbol{\theta}^{(i)}|\mathbf{y}^{(i)};\boldsymbol{\lambda}), \tag{21}$$

$$\widehat{\nabla_{\mathbf{d}} U_L}(\mathbf{d};\boldsymbol{\lambda}) \coloneqq \frac{1}{N_{\text{batch}}}\sum_{i=1}^{N_{\text{batch}}} \ln q_{\boldsymbol{\lambda}}(\boldsymbol{\theta}^{(i)}|\mathbf{y}^{(i)},\mathbf{d})\nabla_{\mathbf{d}} \ln p(\mathbf{y}^{(i)}|\boldsymbol{\theta}^{(i)},\mathbf{d}). \tag{22}$$

Similarly, the upper bound gradient in Eqn. (17) yields

$$\widehat{\nabla_{\boldsymbol{\lambda}} U_U}(\mathbf{d};\boldsymbol{\lambda}) \coloneqq -\frac{1}{N_{\text{batch}}}\sum_{i=1}^{N_{\text{batch}}} \nabla_{\boldsymbol{\lambda}} \ln q(\mathbf{y}^{(i)};\boldsymbol{\lambda}). \tag{23}$$

In this work, we employ stochastic gradient ascent (SGA) for optimization. In order to control the total number of forward model runs, instead of sampling $(\boldsymbol{\theta}^{(i)},\mathbf{y}^{(i)})$ from their true distributions every instance, we first generate a pool of $N_{\text{opt}}$ sample pairs at each $\mathbf{d}$, and then subdivide them into mini-batches with size $N_{\text{batch}}$ for the SGA update iterations.

Foster *et al.* [23] initially proposed parameterizing $q(\,\cdot\,;\boldsymbol{\lambda})$ using relatively simple forms of distributions such as Gaussian, Bernoulli, Beta, and simple transformations of them. Below we present the use of NFs to parameterize $q(\,\cdot\,;\boldsymbol{\lambda})$, with the potential to capture a richer space of distributions and in turn to achieve tighter $U_L$ and $U_U$ bounds.

### *3.1. Normalizing flows*

An NF is an invertible mapping from a target random variable $\mathbf{X} \sim p_{\mathbf{X}}(\mathbf{x})$ to a standard normal random variable $\mathbf{Z} \sim p_{\mathbf{Z}}(\mathbf{z}) = \mathcal{N}(\mathbf{0},\mathbb{I})$ of the same dimension: $\mathbf{Z} = f(\mathbf{X})$ and $\mathbf{X} = g(\mathbf{Z})$ where $g \coloneqq f^{-1}$. In terms of transport maps, $f$ is known as the pushforward map and $g$ is the pullback map [29]. In practice, we approximate $f$ via a mapping parameterized with $\boldsymbol{\lambda}$, which produces an approximate transformation $\widetilde{\mathbf{Z}} = f(\mathbf{X};\boldsymbol{\lambda})$ and its inverse $g(\,\cdot\,;\boldsymbol{\lambda}) \coloneqq f(\,\cdot\,;\boldsymbol{\lambda})^{-1}$ produces $\mathbf{X} = g(\widetilde{\mathbf{Z}};\boldsymbol{\lambda})$. If acting on the exact standard normal $\mathbf{Z}$, then $\widetilde{\mathbf{X}} = g(\mathbf{Z};\boldsymbol{\lambda})$ and also $\mathbf{Z} = f(\widetilde{\mathbf{X}};\boldsymbol{\lambda})$.

In general, the approximate mappings used in NFs are often structured as compositions of successive simple invertible mappings: $f(\widetilde{\mathbf{X}};\boldsymbol{\lambda}) = f_n \circ f_{n-1} \circ ... \circ f_1(\widetilde{\mathbf{X}}) = f_n(f_{n-1}(...(f_1(\widetilde{\mathbf{X}}))...))$ and $g(\mathbf{Z};\boldsymbol{\lambda}) = g_1 \circ ...g_{n-1} \circ g_n(\mathbf{Z}) = g_1(g_2(...(g_n(\mathbf{Z}))...))$ with $g_i = f_i^{-1}$ and $n \geq 1$. Note that all intermediate mappings $f_i$ and $g_i$ depend on $\boldsymbol{\lambda}$, but we omit their subscripts to simplify notation. The log density of $\widetilde{\mathbf{X}}$ can be tracked via the change-of-variable formula:

$$\ln q_{\widetilde{\mathbf{X}}}(\widetilde{\mathbf{X}} = \mathbf{x};\boldsymbol{\lambda}) = \ln p_{\mathbf{Z}}(f_n \circ f_{n-1} \circ ... \circ f_1(\widetilde{\mathbf{X}} = \mathbf{x})) + \sum_{i=1}^{n} \ln \left|\det \frac{\partial f_i \circ f_{i-1} \circ ... f_1(\widetilde{\mathbf{X}})}{\partial \widetilde{\mathbf{X}}}\right|_{\widetilde{\mathbf{X}}=\mathbf{x}}, \tag{24}$$

where $\frac{\partial f_i(\widetilde{\mathbf{X}})}{\partial \widetilde{\mathbf{X}}}$is the Jacobian of $f_i$. By applying successive transformations on $\mathbf{Z}$, the PDF of the resulting variable can be highly expressive [25, 37] and effective for multi-modal, skewed, or other non-standard distribution shapes.

In the context of vOED lower bound in Eqn. (5), we use NF-based $q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda})$ to approximate $p(\boldsymbol{\theta}|\mathbf{y},\mathbf{d})$. Among a range of choices for the architecture of invertible mappings [26, 27], we adopt the coupling layers [24] as a special type of invertible neural network (INN) [51, 52] for its efficient density evaluations and sampling in both forward ($f$) and inverse ($g$) directions. Several papers have shown that composing coupling layers can create flexible flows [45, 53, 54, 51], and recent work by Draxler *et al.* [55] shows that coupling layers form a distributional universal approximator. The basic form of the coupling layer that completes one full transformation starts by partitioning the standard normal random vector $\mathbf{Z} = [\mathbf{Z}_1, \mathbf{Z}_2]^\top$ into two parts of approximately equal dimension—that is, $\mathbf{Z}_1 \in \mathbb{R}^{n_{\theta_1}}$, $\mathbf{Z}_2 \in \mathbb{R}^{n_{\theta_2}}$, and $n_{\theta_1} + n_{\theta_2} = n_\theta$—and then composes together $n = 2$ transformations that transform one part at a time. This maps $\mathbf{Z}$ to an approximate target random vector $\widetilde{\boldsymbol{\Theta}}$, i.e., $g(\mathbf{Z};\boldsymbol{\lambda}) = g_1 \circ g_2(\mathbf{Z}) = \widetilde{\boldsymbol{\Theta}}$, and is defined as:

$$g_2(\mathbf{Z}) = \begin{bmatrix} \widetilde{\boldsymbol{\Theta}}_1 = [\mathbf{Z}_1 - t_2(\mathbf{Z}_2)] \odot \exp[-s_2(\mathbf{Z}_2)] \\ \mathbf{Z}_2 \end{bmatrix}, \tag{25}$$

$$g_1(g_2(\mathbf{Z})) = \begin{bmatrix} \widetilde{\boldsymbol{\Theta}}_1 \\ \widetilde{\boldsymbol{\Theta}}_2 = [\mathbf{Z}_2 - t_1(\widetilde{\boldsymbol{\Theta}}_1))] \odot \exp(-s_1(\widetilde{\boldsymbol{\Theta}}_1)) \end{bmatrix}, \tag{26}$$

where $\odot$ denotes element-wise product, and $s_1, t_1 : \mathbb{R}^{n_{\theta_1}} \to \mathbb{R}^{n_{\theta_2}}$ and $s_2, t_2 : \mathbb{R}^{n_{\theta_2}} \to \mathbb{R}^{n_{\theta_1}}$ are arbitrary functions. The parameterizations of these functions make up $\boldsymbol{\lambda}$; for instance, if these functions are represented by neural networks, then $\boldsymbol{\lambda}$ encompasses all neural networks' weight and bias parameters.

The inverse of $g(\,\cdot\,;\boldsymbol{\lambda})$, which is $f(\widetilde{\boldsymbol{\Theta}};\boldsymbol{\lambda}) = f_2 \circ f_1(\widetilde{\boldsymbol{\Theta}}) = \mathbf{Z}$, similarly involves partitioning $\widetilde{\boldsymbol{\Theta}} = [\widetilde{\boldsymbol{\Theta}}_1, \widetilde{\boldsymbol{\Theta}}_2]^\top$ and can be shown to be:

$$f_1(\widetilde{\boldsymbol{\Theta}}) = \begin{bmatrix} \widetilde{\boldsymbol{\Theta}}_1 \\ \mathbf{Z}_2 = \widetilde{\boldsymbol{\Theta}}_2 \odot \exp(s_1(\widetilde{\boldsymbol{\Theta}}_1)) + t_1(\widetilde{\boldsymbol{\Theta}}_1) \end{bmatrix}, \tag{27}$$

$$f_2(f_1(\widetilde{\boldsymbol{\Theta}})) = \begin{bmatrix} \mathbf{Z}_1 = \widetilde{\boldsymbol{\Theta}}_1 \odot \exp(s_2(\mathbf{Z}_2)) + t_2(\mathbf{Z}_2) \\ \boldsymbol{\Theta'}_2 \end{bmatrix}. \tag{28}$$

The Jacobians of $f_1$ and $f_2$ are triangular matrices:

$$\frac{\partial f_1(\widetilde{\boldsymbol{\Theta}})}{\partial \widetilde{\boldsymbol{\Theta}}} = \begin{bmatrix} \mathbb{I}_{n_{\theta_1}} & \mathbf{0} \\ \frac{\partial \mathbf{Z}_2}{\partial \widetilde{\boldsymbol{\Theta}}_1} & \mathrm{diag}(\exp(s_1(\widetilde{\boldsymbol{\Theta}}_1))) \end{bmatrix}, \qquad \frac{\partial f_2(f_1(\widetilde{\boldsymbol{\Theta}}))}{\partial (f_1(\widetilde{\boldsymbol{\Theta}}))} = \begin{bmatrix} \mathrm{diag}(\exp(s_2(\mathbf{Z}_2)) & \frac{\partial \mathbf{Z}_1}{\partial \mathbf{Z}_2} \\ \mathbf{0} & \mathbb{I}_{n_{\theta_2}} \end{bmatrix}, \tag{29}$$

and have respective determinants $\exp(\sum_{j=1}^{n_{\theta_2}} s_1(\widetilde{\boldsymbol{\Theta}}_1)_j)$ and $\exp(\sum_{j=1}^{n_{\theta_1}} s_2(\mathbf{Z}_2)_j)$.

Multiple such complete transformations can be composed together for greater expressiveness. For example, adopting NFs with $T = 3$ sets of complete transformations entails $f(\widetilde{\boldsymbol{\Theta}};\boldsymbol{\lambda}) = (f_2 \circ f_1)^{T3} \circ (f_2 \circ f_1)^{T2} \circ (f_2 \circ f_1)^{T1}(\widetilde{\boldsymbol{\Theta}})$ and $g(\mathbf{Z};\boldsymbol{\lambda}) = (g_1 \circ g_2)^{T3} \circ (g_1 \circ g_2)^{T2} \circ (g_1 \circ g_2)^{T1}(\mathbf{Z})$.

To incorporate the $\mathbf{y}$-dependence into the approximate posteriors $q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda})$, the $s$ and $t$ functions are designed to additionally take $\mathbf{y}$ as input, leading to a form of conditional INNs (cINNs) [56]. When the dimension of $\mathbf{y}$ is large, a summary network [52] that is common to all the $s$ and $t$ functions, effectively functioning as an encoder, can be employed to compress $\mathbf{y}$ into a lower-dimensional summary statistic (i.e., latent variables) $\mathbf{y}'$. The parameters of the summary network then become part of $\boldsymbol{\lambda}$. The summary network can be particularly beneficial to contain the growth of parameters when $s$ and $t$ functions are represented by neural networks. Figure 1 illustrates the overall cINN transformation for vOED lower bound.

In the case of vOED upper bound in Eqn. (13), NF-based $q(\mathbf{y};\boldsymbol{\lambda})$ can be similarly used to approximate $p(\mathbf{y})$. The transformation $g(\mathbf{Z};\boldsymbol{\lambda})$ then mirrors Eqn. (25) and (26), while $f(\mathbf{Y};\boldsymbol{\lambda})$ mirrors Eqn. (27) and (28).

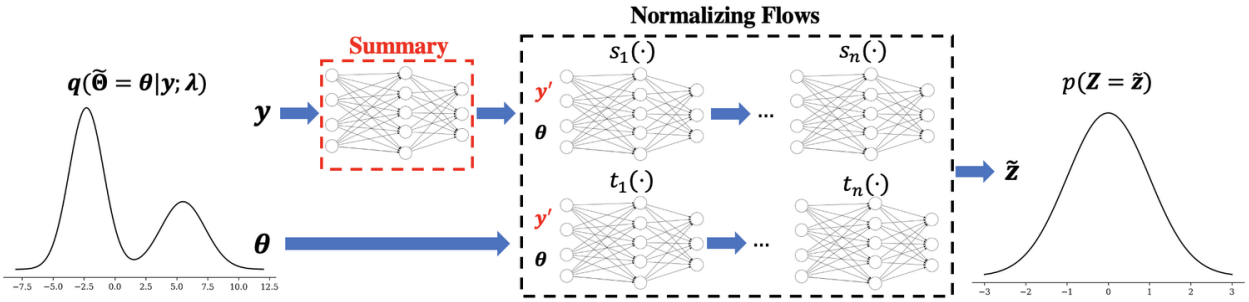



Figure 1: Diagram for the cINN NFs transformation.

## 4. Numerical Experiments

We demonstrate vOED-NFs across four design cases involving models of varying complexity: (1) a nonlinear model with 3 parameters and 1 design variable; (2) a linear model with 21 parameters and 400-dimensional design vector; (3) a PDE model for cathodic electrophoretic deposition; and (4) a stochastic model of aphid population with implicit likelihood.

### *4.1. Case 1: Low-dimensional nonlinear design*

Consider a nonlinear observation model

$$y = G(\boldsymbol{\theta}, d) + \epsilon \qquad \text{where} \qquad G(\boldsymbol{\theta}, d) = \theta_1^3 d^2 + \theta_2 e^{-|0.2-d|} + \sqrt{\theta_3^2 d^2}, \tag{30}$$

with independent prior on parameters $\theta_1 \sim \mathcal{N}(0.5, 0.3^2), \theta_2 \sim \mathcal{N}(0.3, 0.7^2), \theta_3 \sim \mathcal{N}(0.5, 0.8^2)$; $d \in [0, 1]$; and measurement noise $\epsilon$ following a Gaussian mixture distribution:

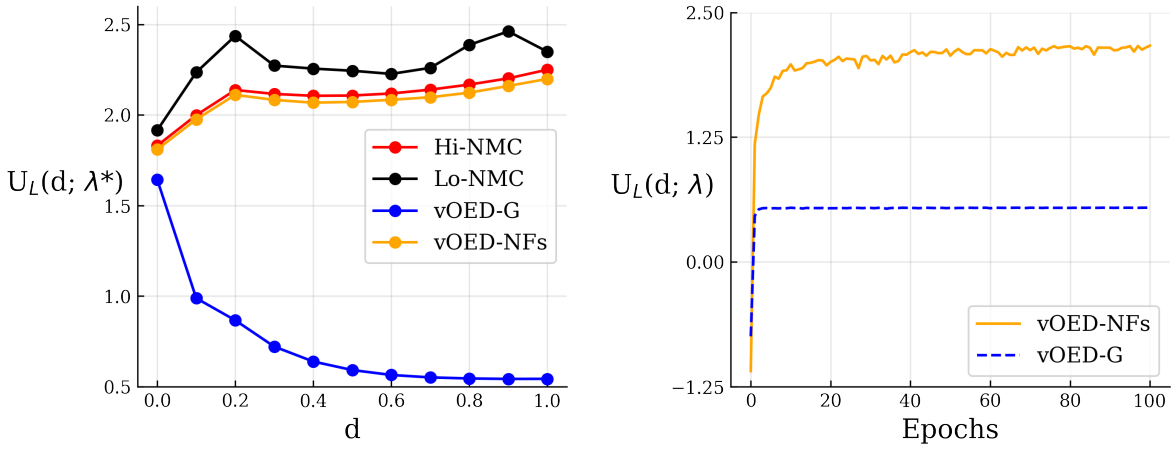
$$p(\epsilon) = \frac{1}{2\sqrt{2\pi\sigma_0^2}} \exp\left(-\frac{(\epsilon-\mu_1)^2}{2\sigma_0^2}\right) + \frac{1}{2\sqrt{2\pi\sigma_0^2}} \exp\left(-\frac{(\epsilon-\mu_2)^2}{2\sigma_0^2}\right), \tag{31}$$

where $\mu_1 = 0.1, \mu_2 = -0.1, \sigma = 0.05$.

Figure 2a presents the EIG estimates across the discretized design space using (a) (Hi-NMC) a high-quality $\widehat{U}_{\text{NMC}}$ with $N_{\text{out}} = 20000$ and $N_{\text{in}} = 20000$ samples, (b) (Lo-NMC) a low-quality $\widehat{U}_{\text{NMC}}$ with $N_{\text{out}} = 200$ and $N_{\text{in}} = 200$ samples, (c) (vOED-G) EIG lower bound $\widehat{U}_L$ using Gaussian distributions optimized with $N_{\text{opt}} = 20000$ samples, and (d) (vOED-NFs) EIG lower bound $\widehat{U}_L$ using NFs optimized with $N_{\text{opt}} = 20000$ samples. Furthermore, the lower bound estimate $\widehat{U}_L$ at each $d$ is evaluated using $N = 10000$ samples. vOED-NFs adopts $T = 5$ sets of complete transformations of the cINN described in Sec. 3.1. Additional training details and hyperparameter choices for (c) and (d) can be found in appendix Appendix B.1.

In Fig. 2a, Hi-NMC is designated as the reference solution. In comparison to this reference, Lo-NMC exhibits moderate bias while vOED-NFs performs extremely well. vOED-G, due to its much simplistic Gaussian variational distributions, deviates significantly from the expected utility trend and misidentifies the optimal design. However, the training cost for vOED-NFs is also higher than vOED-G, as illustrated through the training convergence history for $d = 1.0$ in Fig. 2b. Here vOED-G shows a very rapid convergence to $\boldsymbol{\lambda}^*$, while vOED-NFs takes longer to stabilize and experiences greater fluctuation. This timing difference is negligible when the overall computations are dominated by the forward model runs, but can become significant when the forward model is inexpensive.

To further analyze the ability of vOED-NFs in approximating the true posteriors, we compare the resulting variational posteriors to a high-quality reference posterior obtained using sequential Monte Carlo (SMC) [57] at two designs: $d = 0.2$ in Fig. 3 and $d = 1.0$ in Fig. 4. Each column plots the marginal posteriors of the three parameters $q_{\boldsymbol{\lambda}^*}(\theta_1|y, d)$, $q_{\boldsymbol{\lambda}^*}(\theta_2|y, d)$, $q_{\boldsymbol{\lambda}^*}(\theta_3|y, d)$, while each row corresponds to a different




(a) EIG estimates with different estimators

(b) Training convergence at $d = 1$

Figure 2: Case 1. EIG estimates and sample training convergence plot.

sample of observed $y$. The vOED-NFs posteriors approximate the SMC reference posteriors very well, even with multi-modal posteriors as seen in Fig. 4 for $\theta_3$.

Figure 5 investigates the impact of the number of transformations and optimization sample size $N_{\text{opt}}$ in vOED-NFs. The plots show the EIG lower bound estimates $\widehat{U}_L$ when adopting $T = \{1, 3, 5, 7\}$ sets of complete cINN transformations, and using $N_{\text{opt}} = \{5000, 10000, 50000\}$. The EIG lower bound estimates are still evaluated with $N = 10000$ samples. In general, employing 3 transformations is adequate to identify

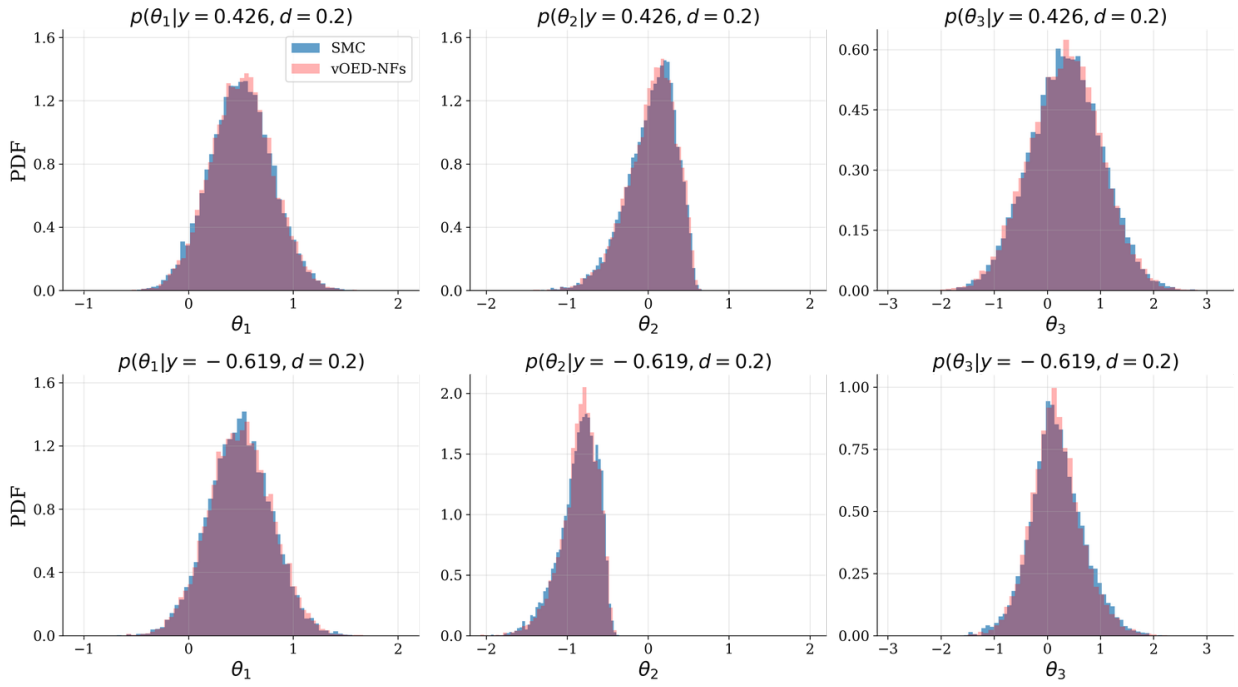



Figure 3: Case 1. Comparison of marginal posteriors obtained from SMC and vOED-NFs at $d = 0.2$.

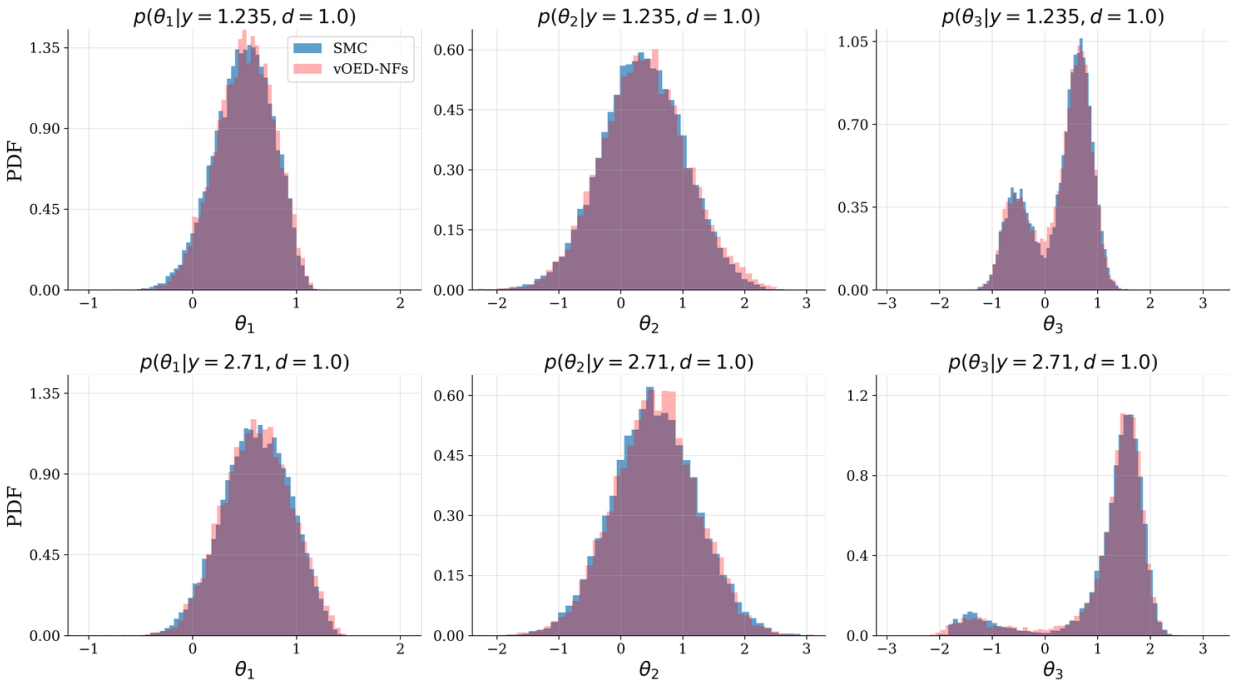



Figure 4: Case 1. Comparison of marginal posteriors obtained from SMC and vOED-NFs at $d = 1.0$.

the optimal design $d^* = 1$. However, when $N_{\text{opt}}$ is small, adding more transformations does not provide significant improvement. When $N_{\text{opt}}$ is increased, the value from adding transformations is more notable and stabilized around $T = 5$; this is consistent with previous work [52] that reported composing more than 5 transformations provides only slight additional benefits. This diminishing return is likely due to the much larger number of NFs parameters when a high number of transformations is used, making them more prone to overfitting. Additional testing and discussions for the $N_{\text{opt}} = 5000$ case can be found in Appendix B.1.

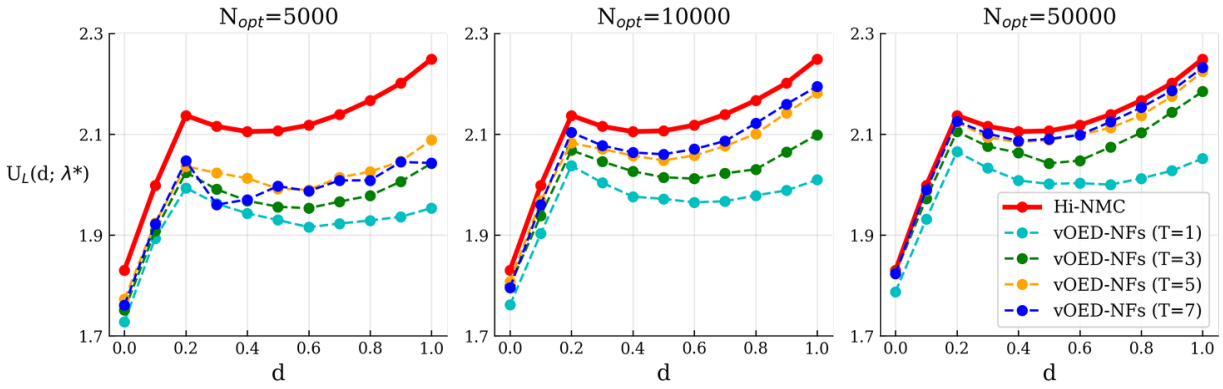



Figure 5: Case 1. EIG under different number of transformations $T$ and optimization sample sizes $N_{\text{opt}}$.

*4.2. Case 2: High-dimensional linear design*

This next example showcases the joint optimization of $\mathbf{d}$ and $\boldsymbol{\lambda}$ and in a higher dimensional setting. The setup follows the example in [14], with an observation model:

$$y_j = \mathbf{d}_j^\top \mathbf{w} + \epsilon_j, \qquad j = 1, \ldots, n, \tag{32}$$

where $y_j \in \mathbb{R}$ is the $j$th observation, $\mathbf{d}_j \in \mathbb{R}^p$ is its corresponding design vector, $\epsilon_j \sim \mathcal{N}(0, \sigma^2)$ is a Gaussian noise random variable, and $\boldsymbol{\theta} = \{\mathbf{w}, \sigma\}$ encompasses the $p$-dimensional regression coefficient vector $\mathbf{w}$ and noise standard deviation $\sigma$. We set $p = 20$ and $n = 20$, hence the parameter dimension is 20 and the total design dimension is 400. Independent priors are adopted for $\mathbf{w}_j \sim \text{Laplace}(0, 1)$ and $\sigma \sim \text{Exp}(1)$. Lastly, a design constraint $||\mathbf{d}_j||_1 = 1$ is imposed to reflect a resource budget; this is implemented through re-normalization of $\mathbf{d}$ after each SGA update in order to map it back to the feasible region. During the joint optimization of $\mathbf{d}$ and $\boldsymbol{\lambda}$, the total number of forward model runs (i.e., the total number of $(\boldsymbol{\theta}^{(i)}, \mathbf{y}^{(i)})$ pairs across all design epochs) is fixed to $10^6$. This is the same computational budget used by [14] that adopted Gaussian and Gamma variational distributions for $\mathbf{w}$ and $\sigma$, respectively; we refer to the setup in [14] as vOED-GG. Further details on the computational setup can be found in Appendix C.1.

Optimization results are presented in Table 1, with the middle column showing the mean $\pm$ one standard deviation of 10 lower bound estimates $\widehat{U}_L$ (each evaluated with $N = 10000$ samples) at the optimal design found by vOED-GG and vOED-NFs, and the right column showing the mean $\pm$ one standard deviation of 10 high-quality NMC estimates (each using $N_{\text{out}} = 10000$ and $N_{\text{in}} = 10000$) at each method's optimal design to provide an accurate EIG estimate and using a common estimation method. We observe that vOED-NFs's optimal design achieves a tighter lower bound estimate and also a higher NMC estimate, indicating that a better design has been found than vOED-GG. However, the vOED-NFs lower bound estimate is not tight, and the remaining discrepancy between $\widehat{U}_L$ and $\widehat{U}_{\text{NMC}}$ may be contributed by a number of factors: (1) the functional form of cINN; (2) the number of cINN transformations; (3) the functional representation of $s$'s and $t$'s; (4) the possibility to converge to a local optimum during optimization; and (5) the finite sample sizes.

| Method | $\widehat{U}_L(\mathbf{d}^*; \boldsymbol{\lambda}^*)$ | $\widehat{U}_{\text{NMC}}(\mathbf{d}^*)$ |
|---|---|---|
| vOED-GG | $12.07 \pm 0.11$ | $24.28 \pm 0.24$ |
| vOED-NFs | $20.94 \pm 0.19$ | $24.91 \pm 0.24$ |

Table 1: Case 2. Middle column shows the mean of 10 lower bound estimates $\widehat{U}_L$ (each is evaluated with $N = 10000$ samples) at the optimal design found by vOED-GG and vOED-NFs, and the right column shows the mean of 10 high-quality NMC estimates (each with $N_{\text{out}} = 10000$ and $N_{\text{in}} = 10000$) at the optimal designs. The $\pm$ values are one standard deviation estimates for the estimators.

In order to compare the posterior approximation capabilities of the two approaches, we look at their approximate posteriors at a common design with each $\mathbf{d}_j = \mathbf{e}_j$ set to the standard unit vector in the $j$th dimension, and at three $\mathbf{y}$ samples. Figures 6 and 7 present the marginal posteriors for $\{w_1, w_2, w_3, w_4, \sigma\}$ using No-U-Turn Hamiltonian Monte Carlo (NUTS-HMC) [58] as the reference posterior distribution, along with the approximations obtained from vOED-GG and vOED-NFs. From the figure, it is evident that the vOED-NFs posteriors can capture the non-Gaussian structures much better than vOED-GG's Gaussian/Gamma distributions.

*4.3. Case 3: Cathodic electrophoretic deposition (e-coating)*

Cathodic electrophoretic deposition, commonly known as e-coating, is a technique for applying protective coatings to automobile surfaces. During the procedure, an anode and a cathode (the car body) are placed in a colloidal bath in which the electric current triggers an electrochemical reaction resulting in the deposition of material on the car body. A proper coating of the car body is achieved when a sufficient thickness of coating is reached. Here we apply vOED-NFs to design e-coating experiments, and employ both the lower and upper bound setups respectively in Eqn. (6) and (14).

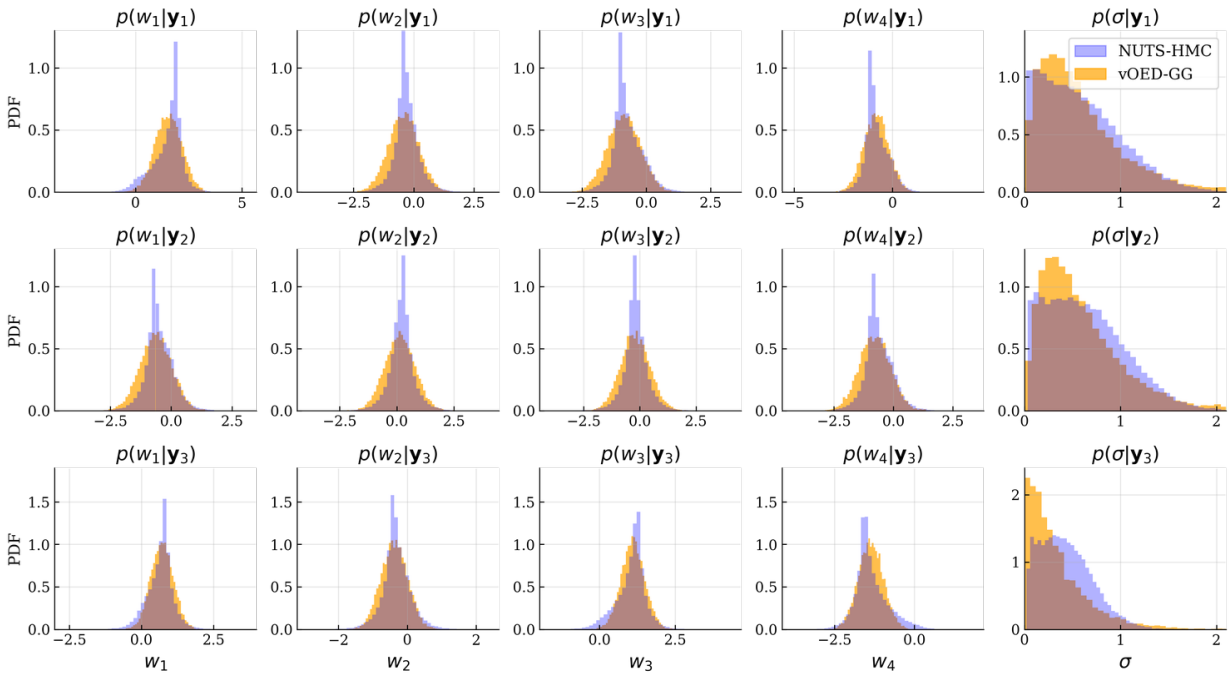



Figure 6: Case 2. Comparison of marginal posteriors obtained from NUTS-HMC and vOED-GG [23] at design where $\mathbf{d}_j = \mathbf{e}_j$.

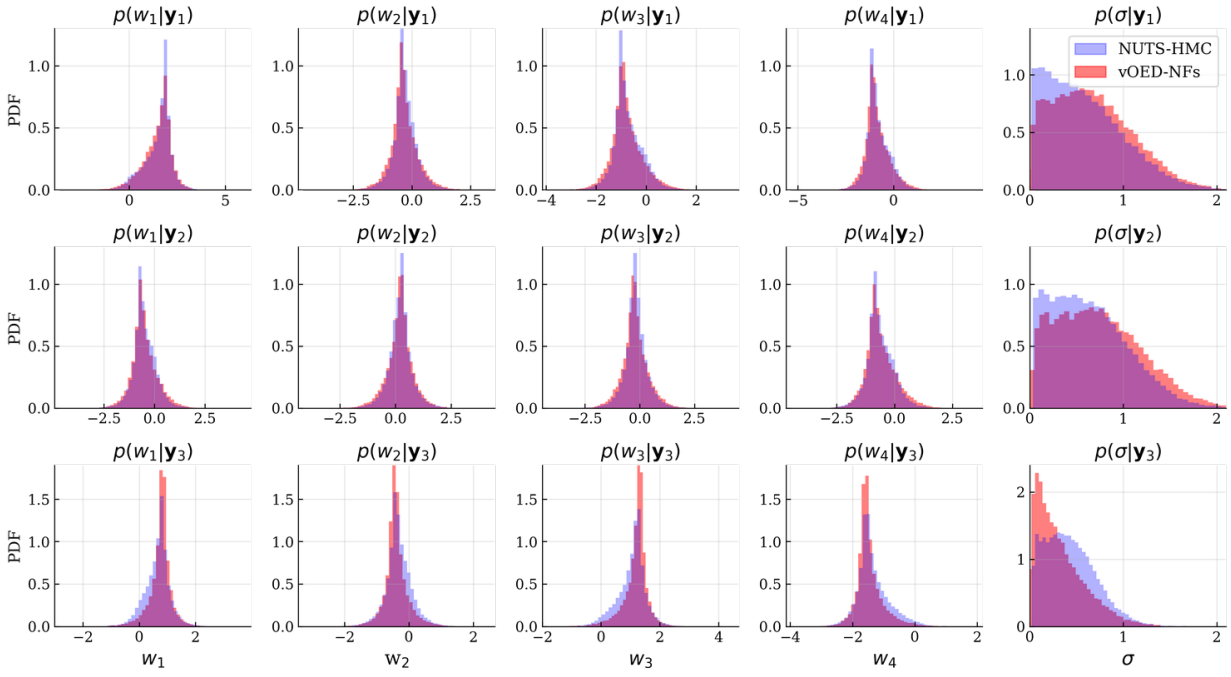



Figure 7: Case 2. Comparison of marginal posteriors obtained from NUTS-HMC and vOED-NFs [23] at design where $\mathbf{d}_j = \mathbf{e}_j$.

The physical model adopted for the e-coating process, consisting of a PDE, is the baseline model from [59] and also summarized in Appendix D.1. The parameters of interest, $\boldsymbol{\theta} = \{C_v, j_{\min}, Q_{\min}\}$, are the volumetric

coulombic efficiency ($m^3/C$), minimum current ($A/m^2$), and minimum charge ($C/m^2$), respectively. Each parameter is endowed with a truncated normal prior, with detailed prior parameters provided in Appendix D.1. A measurement on the current (mA) can be obtained at time $t$ (s), modeled as

$$y_t = j(\boldsymbol{\theta}, j_0, t)(1 + \epsilon), \tag{33}$$

where $j(\boldsymbol{\theta}, j_0, t)$ is the forward model mapping from parameters to observables at time $t$ and under constant current boundary condition $j_0$ (mA), and $\epsilon \sim \mathcal{N}(0, 0.1^2)$ represents a relative measurement noise. The design variable is $j_0$, and three candidate designs are considered: (1) $j_0 = 10$ mA with measurements made at $t = \{10, 20, \ldots, 100\}$ s; (2) $j_0 = 7.5$ mA with measurements made at $t = \{20, 40, \ldots, 200\}$ s; and (3) $j_0 = 5.0$ mA with measurements made at $t = \{30, 60, \ldots, 300\}$ s. Hence, all designs always involve $n_y = 10$ measurements at regular intervals; the lower $j_0$ designs have slower decays of the current (e.g., see Figure D.14 in [59]) and thus longer intervals are adopted.

Figure 8 presents the convergence of the upper and lower bound estimates from Eqn. (19) and (20), along with a reference EIG value, when optimizing over $\boldsymbol{\lambda}$ at design $j_0 = 5.0$ mA. The reference EIG is obtained from a high-quality NMC with the 'reuse' technique described in [8] in order to accommodate the relatively high computational cost for each forward model evaluation (i.e., a PDE solve). Thus, even with $N_{\text{out}} = N_{\text{in}} = 10^5$, the reuse technique only requires $10^5$ total forward model evaluations (instead of $10^{10}$, which would be impractical). We further provide comparison of the bounds between low-quality optimization that uses $N_{\text{opt}} = 10000$ versus high-quality estimation that uses $N_{\text{opt}} = 80000$ (the evaluation of the bounds are always performed with $N = 10000$). Results indicate that increasing $N_{\text{opt}}$ tightens both bounds. Under the same $N_{\text{opt}}$, the upper bound appears to be tighter than the lower bound. To illustrate the ability of $q(\mathbf{y}; \boldsymbol{\lambda})$ in representing the marginal likelihood $p(\mathbf{y}|\mathbf{d})$, their marginal distributions are shown in Fig. 9, where we see excellent agreement between the approximation and the true distributions obtained from direct Monte Carlo.

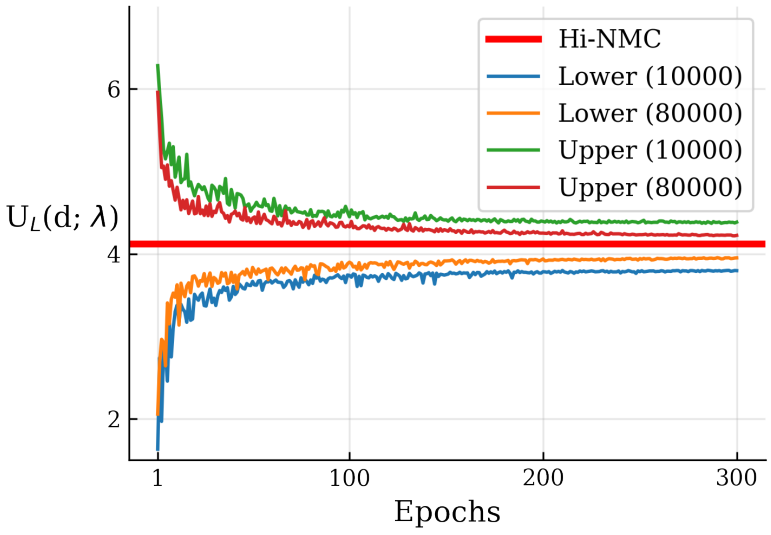



Figure 8: Case 3. Convergence history of the lower and upper bounds at $d = j_0 = 5$ mA and 10 observations when optimizing $\boldsymbol{\lambda}$ using different $N_{\text{opt}}$. Hi-NMC is the high-quality reference EIG estimate. Here the upper bound estimates appear to be tighter than the lower bound estimates.

In scenarios where the dimension of $\boldsymbol{y}$ is high, the input size to the $s$ and $t$ networks of the cINN becomes large. To combat this high dimensionality, summary networks are employed to convert $\mathbf{y}$ to a lower-dimensional $\mathbf{y}'$, specifically through a long short-term memory (LSTM) network [60] as recommended in [52] to accommodate the sequential nature of $\mathbf{y}$. Figure 10 presents the same convergence of bound

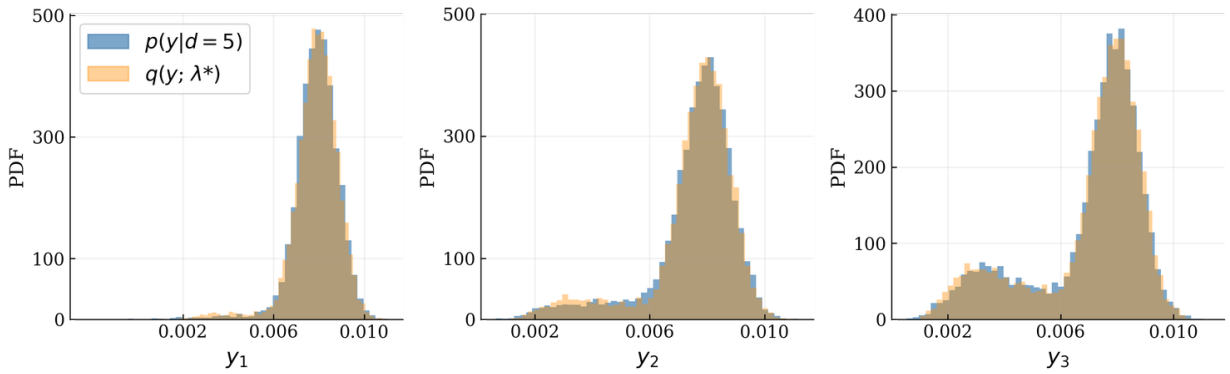



Figure 9: Case 3. Marginal distributions $p(y_1|d), p(y_2|d), p(y_3|d)$ at $d = j_0 = 5$ mA obtained using MC sampling and vOED-NFs with $N_{\text{opt}} = 10000$.

estimates but now with $n_y = 50$ measurements taken in each experiment, i.e., $\boldsymbol{y} = \{y_{t_1}, \ldots, y_{t_{50}}\}$, and with a summary network that compresses it to a 10-dimensional $\mathbf{y}'$. The figure indicates that the lower bound estimates are now tighter than their upper bound counterparts. Together with Fig. 8, these observations suggest that when the dimension of $\mathbf{y}$ significantly exceeds that of $\boldsymbol{\theta}$, EIG may be more accurately estimated though the lower bound estimate that approximates the posterior, than the upper bound estimate that approximates the marginal likelihood.

Various lower bound estimates, using vOED-G, vOED-NFs, vOED-NFs-LSTM, and the high-quality NMC, are computed for all three candidate designs and shown in Fig. 11. All results indicate $j_0 = 0.75$ mA achieving the best design, although vOED-G incurs a notable error in estimating the EIG compared to the rest. The vOED-NFs lower bound estimate is further tightened when the LSTM summary network is used. The vOED-NFs-LSTM using different $N_{\text{opt}}$ also further supports the benefit when optimizing using a larger sample size.

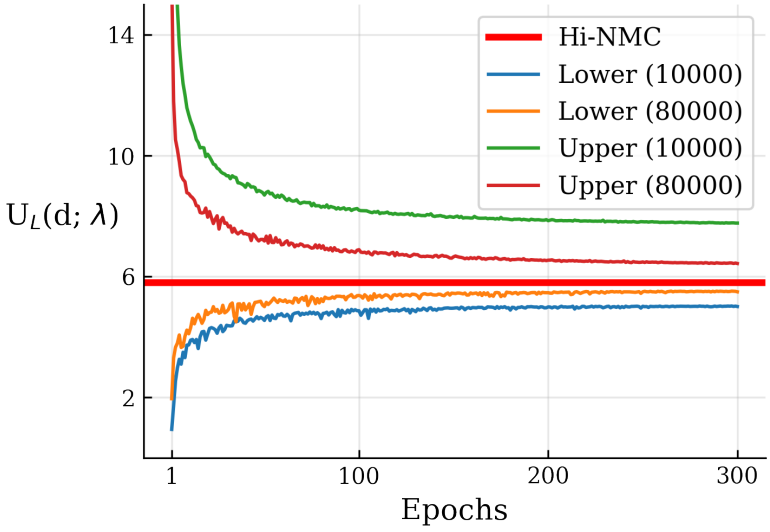



Figure 10: Case 3. Convergence history of the lower and upper bounds at $d = j_0 = 5$ mA and 50 observations when optimizing $\boldsymbol{\lambda}$ using different $N_{\text{opt}}$. Hi-NMC is the high-quality reference EIG estimate. Here the lower bound estimates appear to be tighter than the upper bound estimates.

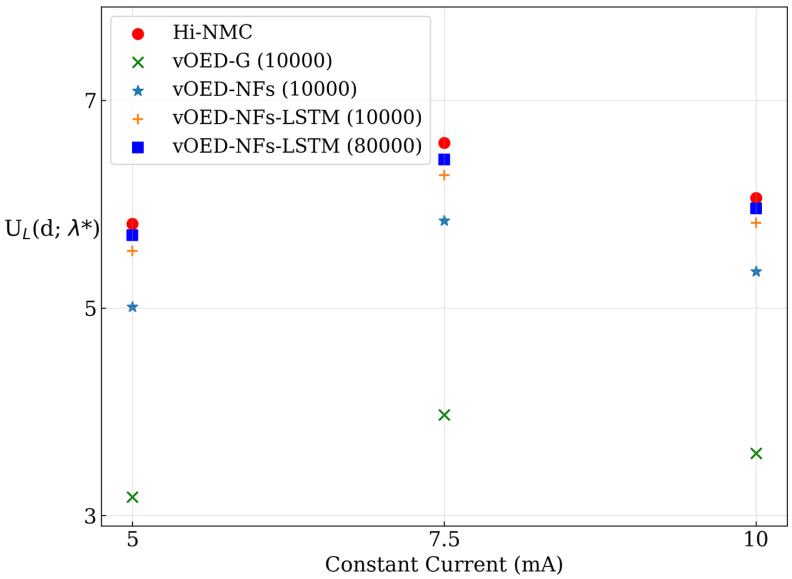
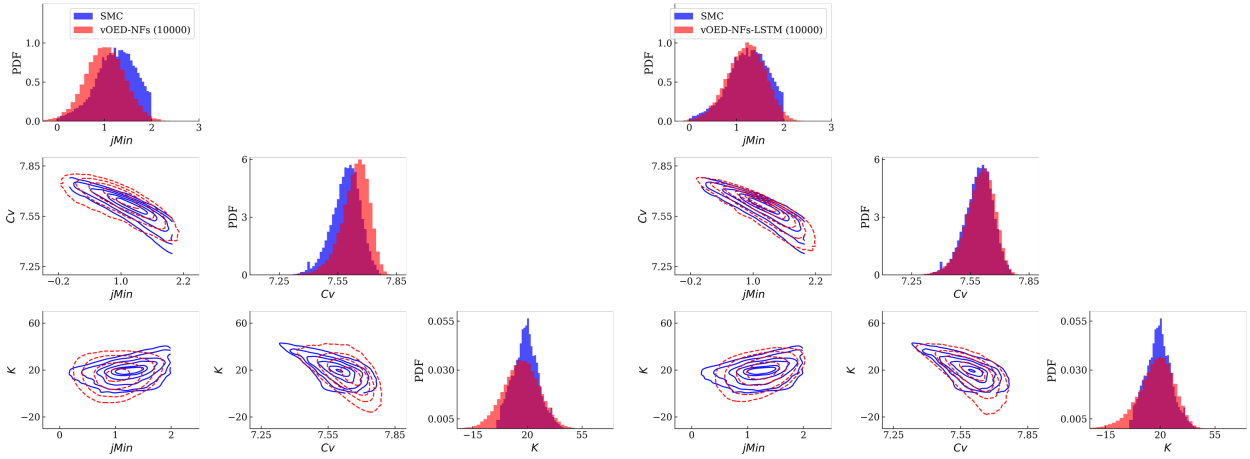



Figure 11: Case 3. Lower bound estimates for all designs using different methods. Hi-NMC is the high-quality reference EIG estimate.

Figure 12 presents the marginal and pairwise posterior distributions obtained from using vOED-NFs and vOED-NFs-LSTM lower bound estimates, along with the reference posteriors obtained from SMC. We observe that the posteriors obtained from vOED-NFs-LSTM in Fig. 12b offering improved approximation to the SMC reference than those from vOED-NFs in Fig. 12a. This improvement supports the effectiveness of employing the LSTM summary network for dimension reduction in this example.

### *4.4. Case 4: aphid population*

The last case involves a stochastic model depicting the growth of aphid population [61], which results in an implicit likelihood scenario where samples may be drawn from the likelihood but probability density




(a) vOED-NFs (b) vOED-NFs-LSTM

Figure 12: Case 3. Marginal and pairwise joint posterior distributions obtained using SMC (high-quality reference) and vOED-NFs-LSTM.

cannot be evaluated. Denote $M(t)$ and $C(t)$ respectively the current and accumulative sizes of the aphid population, the probability that a birth or death occurs in a small time period $\delta_t$ is [62]:

$$\mathbb{P}\{M(t+\delta_t) = m+1, C(t+\delta_t) = c+1) \,|\, M(t) = m, C(t) = c\} = \alpha m \delta_t + o(\delta_t), \tag{34}$$

$$\mathbb{P}\{M(t+\delta_t) = m-1, C(t+\delta_t) = c \,|\, M(t) = m, C(t) = c\} = \beta m c \delta_t + o(\delta_t), \tag{35}$$

where $\boldsymbol{\theta} = \{\alpha, \beta\}$ are the birth and death model parameters, respectively. In practice a non-zero $\delta_t$ must be used, leading to a discretized approximation to the continuum limit. We adopt a prior distribution following [62]:

$$\begin{pmatrix} \alpha \\ \beta \end{pmatrix} \sim \mathcal{N}\left[ \begin{pmatrix} 0.246 \\ 0.000136 \end{pmatrix}, \begin{pmatrix} 0.0079^2 & 5.8 \times 10^{-8} \\ 5.8 \times 10^{-8} & 0.00002^2 \end{pmatrix} \right], \tag{36}$$

and an initial condition of $M(0) = C(0) = 28$. In each experiment, only $M(t)$ is observed at a particular observation time $t$ and $C(t)$ is not observed. An exact likelihood evaluation of $\mathbb{P}(M(t) = m \,|\, \boldsymbol{\theta})$ thus requires the summation of probabilities along all paths that end at $M(t) = m$ and then marginalizing out $C(t)$; this is generally intractable to compute.

The OED problem is then to determine the optimal $k$ measurement times, $\mathbf{d} = \{t_1, t_2, ..., t_k\}$, $t_k \in [0, 50]$. Following [15], we solve the design optimization for $k = 1$ and 2 (i.e., designing for respectively 1 and 2 measurement times) through grid search, while for $k = 3$ and 4 we use the simultaneous perturbation stochastic approximation (SPSA) algorithm [63]. Table 2 presents the design results obtained using vOED-NFs and the LB-KLD method that was used in [15]. LB-KLD constructs another lower bound to EIG based on the entropy power inequality and entropy's concavity property, and adopts the nearest neighbor based entropy estimator within the lower bound estimator. Note that LB-KLD does not involve tightening the lower bound and therefore is not a variational lower bound, but instead directly estimates the lower bound, here using 30000 samples. For vOED-NFs, we set $N_{\text{opt}} = 20000$ and $N = 10000$ such that the total number of samples is the same. Implementation details can be found in Appendix E.1.

| $k$ | Method | $\mathbf{d}^*$ | $\widehat{U}_{\text{LB-KLD}}(\mathbf{d}^*)$ | $\widehat{U}_L(\mathbf{d}^*; \boldsymbol{\lambda}^*)$ |
|---|---|---|---|---|
| 1 | LB-KLD | (21) | 1.18 (0.012) | 1.22 (0.003) |
| | vOED-NFs | (21) | 1.18 (0.012) | 1.22 (0.003) |
| 2 | LB-KLD | (17, 28) | 1.86 (0.021) | 1.86 (0.018) |
| | vOED-NFs | (17, 27) | 1.88 (0.008) | 1.89 (0.004) |
| 3 | LB-KLD | (15.7, 22.7, 32.0) | 2.00 (0.019) | 2.08 (0.004) |
| | vOED-NFs | (14.6, 20.7, 28.6) | 2.00 (0.018) | 2.10 (0.005) |
| 4 | LB-KLD | (13.8, 19.1, 24.5, 30.6) | 2.05 (0.017) | 2.20 (0.006) |
| | vOED-NFs | (13.7, 19.2, 24.8, 32.7) | 2.06 (0.014) | 2.19 (0.006) |

Table 2: Case 4. The third column shows the optimal design $\mathbf{d}^*$ found by LB-KLD and vOED-NFs. The fourth column shows the mean of 10 $\widehat{U}_{\text{LB-KLD}}$ estimates with different random seeds (each with 30000 samples for Monte Carlo estimation) at the optimal designs, and $(\cdot)$ indicates the standard deviation originates from different random partitions of the same 30000 samples used for entropy estimation. The last column shows the mean of 10 $\widehat{U}_L$ estimates (each with $N_{\text{opt}} = 20000$ and $N = 10000$) at the optimal designs but different $\boldsymbol{\lambda}^*$. Here $(\cdot)$ indicates the standard deviation due to 10 random seeds for different initialization when optimizing $\boldsymbol{\lambda}^*$, and evaluation is conducted on the same $N = 10000$ samples.

Table 2 indicates that both methods generally propose comparable optimal designs for various $k$ values and similar lower bound estimates. Notably at $k = 3$ and 4, vOED-NFs produce tighter bound estimators than LB-KLD. Samples from the approximate posterior in vOED-NFs for a simulated $\mathbf{y}$ at $k = 4$ are illustrated in Fig. 13, which agree well with reference posterior samples obtained from approximate Bayesian computation (ABC) [64]. Note that posterior information is only available owing to vOED-NFs's use of

$q$ to approximate the posterior density; in contrast, other lower bound approaches not based on posterior approximation, for example LB-KLD, do not offer any posterior information.

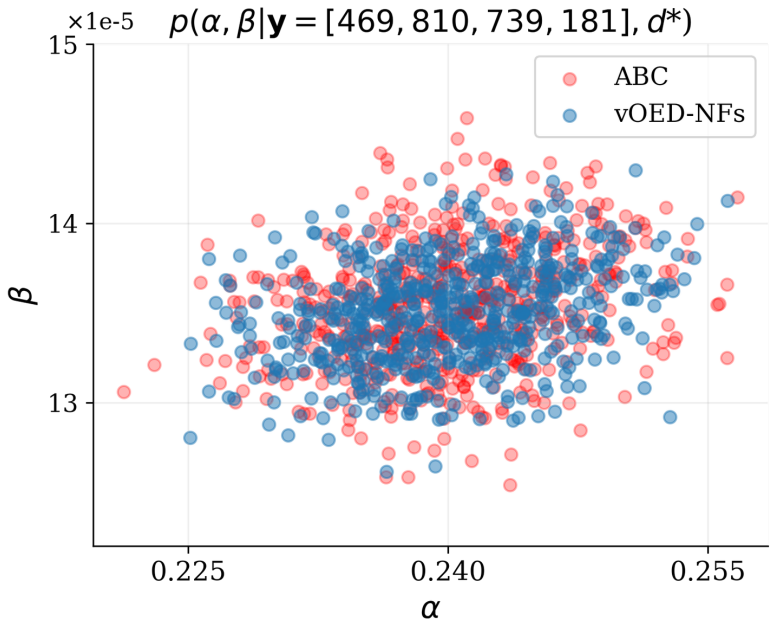



Figure 13: Case 4. Comparison of joint posteriors samples from ABC and vOED-NFs at the optimal design.

## 5. Conclusions

This paper introduced vOED-NFs, a method to use normalizing flows to represent variational distributions in the context of Bayesian OED. When the expected utility of OED is chosen to be the EIG in the model parameters, the Barber–Agakov lower bound may be used to estimate the EIG and the bound tightened by optimizing its variational parameters. We presented Monte Carlo estimators to both lower and upper bound versions along with their gradient expressions. We then detailed the use of NFs, in particular with cINN architecture involving a composition of coupling layers and together with a summary network for dimension reduction, to approximate the posterior or marginal likelihood distributions.

We validated vOED-NFs against established methods in two benchmark problems, and demonstrated vOED-NFs on a PDE-governed application of cathodic electrophoretic deposition and stochastic modeling of aphid population with implicit likelihood. The findings suggested that 4–5 compositions of the coupling layers were adequate to achieve a lower bias compared to previous approaches. Furthermore, we illustrated that vOED-NFs produced approximate posteriors that matched very well with the true posteriors, able to capture non-Gaussian and multi-modal features effectively.

A limitation of vOED-NFs is the lack of analytical results connecting bound estimator quality to design optimization. For example, even if the lower bounds are not tight but the bound gap is consistent across $\mathcal{D}$, the design maximizer may still be similar to the true optimal design. Another limitation is that, to retain good accuracy under high-dimensional $\boldsymbol{\theta}$, complex NFs architectures with high-dimensional $\boldsymbol{\lambda}$ may be needed, making the optimization problem challenging. Future research exploring efficient and scalable transport map architectures in vOED, such as with rectification operator [36] and probability flow ODEs [65], will be useful. The adoption of vOED-NFs into other OED structures, such as goal-oriented OED [66] and sequential OED [67], will also be important.

More broadly, OED approaches that rely on EIG bounds can only be used when the expected utility is the EIG (possibly with other terms that do not depend on the posterior) and cannot be applied to other choices of design criteria. Moreover, the bounds (i.e., Eqn. (5) and (13)) cannot be computed in closed form, but only estimated through, for example, Monte Carlo sampling (i.e., Eqn. (19) and (20)). Consequently, the bound estimators can no longer provide inequality guarantees due to the sampling variance. Thus, as a candidate for future exploration, there is a great need for sampling efficiency for bound-based strategies, to keep the bound estimator variance lower than the bias (i.e., the bound gap).

**Acknowledgement**

This work was supported at the University of Michigan by Ford Motor Company under the grant "Hybrid Physics-Machine Learning Models for Electrodeposition".

## References


[1] F. Pukelsheim, Optimal Design of Experiments, Society for Industrial and Applied Mathematics, 2006.

[2] K. Chaloner, I. Verdinelli, Bayesian experimental design: A review, Statistical Science 10 (3) (1995) 273–04. doi:10.1214/ss/1177009939.

[3] A. Atkinson, A. Donev, R. Tobias, Optimum Experimental Designs, with SAS, Oxford University Press, 2007. doi:10.1093/oso/9780199296590.001.0001.

[4] E. G. Ryan, C. C. Drovandi, J. M. McGree, A. N. Pettitt, A review of modern computational algorithms for Bayesian optimal design, International Statistical Review 84 (1) (2016) 128–154. doi:10.1111/insr.12107.

[5] T. Rainforth, A. Foster, D. R. Ivanova, F. B. Smith, Modern Bayesian experimental design, Statistical Science 39 (1) (2023) 100–114. doi:10.1214/23-STS915.

[6] D. V. Lindley, On a measure of the information provided by an experiment, The Annals of Mathematical Statistics 27 (4) (1956) 986–1005. doi:10.1214/aoms/1177728069.

[7] K. J. Ryan, Estimating expected information gains for experimental designs with application to the random fatigue-limit model, Journal of Computational and Graphical Statistics 12 (3) (2003) 585–603. doi:10.1198/1061860032012.

[8] X. Huan, Y. M. Marzouk, Simulation-based optimal Bayesian experimental design for nonlinear systems, Journal of Computational Physics 232 (1) (2013) 288–317. doi:10.1016/j.jcp.2012.08.013.

[9] X. Huan, Y. Marzouk, Gradient-based stochastic optimization methods in Bayesian experimental design, International Journal for Uncertainty Quantification 4 (6) (2014). doi:10.1615/Int.J.UncertaintyQuantification.2014006730.

[10] Q. Long, M. Scavino, R. Tempone, S. Wang, Fast estimation of expected information gains for Bayesian experimental designs based on Laplace approximations, Computer Methods in Applied Mechanics and Engineering 259 (2013) 24–39. doi:10.1016/j.cma.2013.02.017.

[11] J. Yu, V. M. Zavala, M. Anitescu, A scalable design of experiments framework for optimal sensor placement, Journal of Process Control 67 (2018) 44–55. doi:10.1016/j.jprocont.2017.03.011.

[12] K. Wu, P. Chen, O. Ghattas, A fast and scalable computational framework for large-scale high-dimensional Bayesian optimal experimental design, SIAM/ASA Journal on Uncertainty Quantification 11 (1) (2023) 235–261. doi:10.1137/21M1466499.

[13] J. Beck, B. M. Dia, L. F. Espath, Q. Long, R. Tempone, Fast Bayesian experimental design: Laplace-based importance sampling for the expected information gain, Computer Methods in Applied Mechanics and Engineering 334 (2018) 523–553. doi:10.1016/j.cma.2018.01.053.

[14] A. Foster, M. Jankowiak, M. O'Meara, Y. W. Teh, T. Rainforth, A unified stochastic gradient approach to designing Bayesian-optimal experiments, in: Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics, Vol. 108 of Proceedings of Machine Learning Research, PMLR, 2020, pp. 2959–2969.

[15] Z. Ao, J. Li, An approximate KLD based experimental design for models with intractable likelihoods, in: Proceedings of the Twenty Third International Conference on Artificial Intelligence and Statistics, Vol. 108 of Proceedings of Machine Learning Research, PMLR, 2020, pp. 3241–3251.

[16] B. Poole, S. Ozair, A. Van Den Oord, A. Alemi, G. Tucker, On variational bounds of mutual information, in: Proceedings of the 36th International Conference on Machine Learning, Vol. 97 of Proceedings of Machine Learning Research, PMLR, 2019, pp. 5171–5180.

[17] X. Nguyen, M. J. Wainwright, M. I. Jordan, Estimating divergence functionals and the likelihood ratio by convex risk minimization, IEEE Transactions on Information Theory 56 (11) (2010) 5847–5861. doi:10.1109/TIT.2010.2068870.

[18] M. I. Belghazi, A. Baratin, S. Rajeswar, S. Ozair, Y. Bengio, A. Courville, R. D. Hjelm, Mine: mutual information neural estimation, arXiv preprint arXiv:1801.04062 (2018).

[19] S. Kleinegesse, M. U. Gutmann, Bayesian experimental design for implicit models by mutual information neural estimation, in: Proceedings of the 37th International Conference on Machine Learning, Vol. 119 of Proceedings of Machine Learning Research, PMLR, 2020, pp. 5316–5326.

[20] A. van den Oord, Y. Li, O. Vinyals, Representation learning with contrastive predictive coding, arXiv preprint arXiv:1807.03748 (2018).

[21] D. R. Ivanova, A. Foster, S. Kleinegesse, M. U. Gutmann, T. Rainforth, Implicit deep adaptive design: Policy-based experimental design without likelihoods, Advances in Neural Information Processing Systems 34 (2021) 25785–25798.

[22] D. Barber, F. Agakov, The IM Algorithm: A variational approach to information maximization, in: Advances in Neural Information Processing Systems 16, 2003, p. 201–208.

[23] A. Foster, M. Jankowiak, E. Bingham, P. Horsfall, Y. W. Teh, T. Rainforth, N. Goodman, Variational Bayesian optimal experimental design, in: Advances in Neural Information Processing Systems 32, 2019, p. 14036–14047.

[24] L. Dinh, J. Sohl-Dickstein, S. Bengio, Density estimation using Real NVP, arXiv preprint arXiv:1605.08803 (2016).

[25] D. Rezende, S. Mohamed, Variational inference with normalizing flows, in: Proceedings of the 32nd International Conference on Machine Learning, Vol. 37 of Proceedings of Machine Learning Research, PMLR, 2015, pp. 1530–1538.

[26] G. Papamakarios, E. T. Nalisnick, D. J. Rezende, S. Mohamed, B. Lakshminarayanan, Normalizing flows for probabilistic modeling and inference, Journal of Machine Learnine Research 22 (57) (2021) 1–64.

[27] I. Kobyzev, S. J. Prince, M. A. Brubaker, Normalizing flows: An introduction and review of current methods, IEEE Transactions on Pattern Analysis and Machine Intelligence 43 (11) (2020) 3964–3979. doi:10.1109/TPAMI.2020.2992934.

[28] C. Villani, Optimal Transport: Old and New, Springer-Verlag Berlin Heidelberg, Berlin, Germany, 2008.

[29] Y. Marzouk, T. Moselhy, M. Parno, A. Spantini, Sampling via measure transport: An introduction, in: Handbook of Uncertainty Quantification, Springer International Publishing, Cham, 2016, pp. 1–41. doi:10.1007/978-3-319-11259-6_23-1.

[30] A. Spantini, D. Bigoni, Y. Marzouk, Inference via low-dimensional couplings, Journal of Machine Learning Research 19 (66) (2018) 1–71.

[31] T. P. Minka, Expectation Propagation for approximate Bayesian inference, in: Proceedings of the Seventeenth Conference on Uncertainty in Artificial Intelligence (UAI 2001), 2001, pp. 362–369.

[32] V. I. Bogachev, A. V. Kolesnikov, K. V. Medvedev, Triangular transformations of measures, Sbornik: Mathematics 196 (3) (2005) 309.

[33] T. A. El Moselhy, Y. M. Marzouk, Bayesian inference with optimal maps, Journal of Computational Physics 231 (23) (2012) 7815–7850. doi:10.1016/j.jcp.2012.07.022.

[34] T. Cui, S. Dolgov, Deep composition of tensor-trains using squared inverse Rosenblatt transports, Foundations of Computational Mathematics 22 (6) (2021) 1863–1922. doi:10.1007/s10208-021-09537-5.

[35] Z. O. Wang, R. Baptista, Y. Marzouk, L. Ruthotto, D. Verma, Efficient neural network approaches for conditional optimal transport with applications in Bayesian inference, arXiv preprint: arxiv:2310.16975 (2023).

[36] R. Baptista, Y. Marzouk, O. Zahm, On the representation and learning of monotone triangular transport maps, Foundations of Computational Mathematics (2023) 1–46doi:10.1007/s10208-023-09630-x.

[37] E. G. Tabak, E. Vanden-Eijnden, Density estimation by dual ascent of the log-likelihood, Communications in Mathematical Sciences 8 (1) (2010) 217–233.

[38] E. G. Tabak, C. V. Turner, A family of nonparametric density estimation algorithms, Communications on Pure and Applied Mathematics 66 (2) (2013) 145–164.

[39] J. M. Tomczak, M. Welling, Improving variational auto-encoders using Householder flow, arXiv preprint arXiv:1611.09630 (2016).

[40] C. Louizos, M. Welling, Multiplicative normalizing flows for variational Bayesian neural networks, in: Proceedings of the 34th International Conference on Machine Learning, Vol. 70 of Proceedings of Machine Learning Research, PMLR, 2017, pp. 2218–2227.

[41] R. van den Berg, L. Hasenclever, J. M. Tomczak, M. Welling, Sylvester normalizing flows for variational inference, arXiv preprint arXiv:1803.05649 (2018).

[42] L. Dinh, D. Krueger, Y. Bengio, Nice: Non-linear independent components estimation, arXiv preprint arXiv:1410.8516 (2014).

[43] D. P. Kingma, T. Salimans, R. Jozefowicz, X. Chen, I. Sutskever, M. Welling, Improved variational inference with inverse autoregressive flow, in: Advances in Neural Information Processing Systems 29, 2016.

[44] G. Papamakarios, T. Pavlakou, I. Murray, Masked autoregressive flow for density estimation, in: Advances in Neural Information Processing Systems 30, 2017.

[45] D. P. Kingma, P. Dhariwal, Glow: Generative flow with invertible 1x1 convolutions, in: Advances in Neural Information Processing Systems 31, 2018.

[46] C. Durkan, A. Bekasov, I. Murray, G. Papamakarios, Cubic-spline flows, arXiv preprint arXiv:1906.02145 (2019).

[47] R. T. Q. Chen, J. Behrmann, D. K. Duvenaud, J.-H. Jacobsen, Residual flows for invertible generative modeling, in: Advances in Neural Information Processing Systems 32, 2019.

[48] R. T. Q. Chen, Y. Rubanova, J. Bettencourt, D. K. Duvenaud, Neural ordinary differential equations, in: Advances in Neural Information Processing Systems 31, 2018.

[49] C. Feng, Y. M. Marzouk, A layered multiple importance sampling scheme for focused optimal Bayesian experimental design, arXiv preprint arXiv:1903.11187 (2019).

[50] T. Rainforth, R. Cornish, H. Yang, A. Warrington, F. Wood, On nesting monte carlo estimators, in: Proceedings of the 35th International Conference on Machine Learning, Vol. 80 of Proceedings of Machine Learning Research, PMLR, 2018, pp. 4267–4276.

[51] J. Kruse, G. Detommaso, U. Köthe, R. Scheichl, HINT: Hierarchical invertible neural transport for density estimation and Bayesian inference, in: Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 35, 2021, pp. 8191–8199. doi:10.1609/aaai.v35i9.16997.

[52] S. T. Radev, U. K. Mertens, A. Voss, L. Ardizzone, U. Köthe, BayesFlow: learning complex stochastic models with invertible neural networks, IEEE Transactions on Neural Networks and Learning Systems 33 (4) (2020) 1452–1466. doi:10.1109/TNNLS.2020.3042395.

[53] R. Prenger, R. Valle, B. Catanzaro, Waveglow: a flow-based generative network for speech synthesis, in: ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE, 2019, pp. 3617–3621. doi:10.1109/ICASSP.2019.8683143.

[54] L. Ardizzone, J. Kruse, C. Rother, U. Köthe, Analyzing inverse problems with invertible neural networks, in: International Conference on Learning Representations, 2019.

[55] F. Draxler, S. Wahl, C. Schnörr, U. Köthe, On the universality of coupling-based normalizing flows, arXiv preprint arXiv:2402.06578 (2024).

[56] G. A. Padmanabha, N. Zabaras, Solving inverse problems using conditional invertible neural networks, Journal of Computational Physics 433 (2021) 110194. doi:10.1016/j.jcp.2021.110194.

[57] A. Doucet, N. Freitas, K. Murphy, S. Russell, Sequential Monte Carlo Methods in Practice, Springer New York, 2013. doi:10.1007/978-1-4757-3437-9.

[58] M. D. Hoffman, A. Gelman, The No-U-Turn sampler: adaptively setting path lengths in Hamiltonian Monte Carlo, Journal of Machine Learning Research 15 (47) (2014) 1593–1623.

[59] C. Jacobsen, J. Dong, M. Khalloufi, X. Huan, K. Duraisamy, M. Akram, W. Liu, Enhancing dynamical system modeling through interpretable machine learning augmentations: a case study in cathodic electrophoretic deposition, arXiv preprint arXiv:2401.08414 (2024).

[60] F. Gers, J. Schmidhuber, F. Cummins, Learning to forget: continual prediction with LSTM, Neural Computation 12 (2000) 2451–2471. doi:10.1162/089976600300015015.

[61] J. Matis, T. Kiffe, T. Matis, D. Stevenson, Nonlinear stochastic modeling of aphid population growth, Mathematical Biosciences 198 (2) (2006) 148–168. doi:10.1016/j.mbs.2005.07.009.

[62] C. S. Gillespie, A. Golightly, Bayesian inference for generalized stochastic population growth models with application to aphids, Journal of the Royal Statistical Society Series C: Applied Statistics 59 (2) (2010) 341–357. doi:10.1111/j.1467-9876.2009.00696.x.

[63] J. C. Spall, Implementation of the simultaneous perturbation algorithm for stochastic optimization, IEEE Transactions on Aerospace and Electronic Systems 34 (3) (1998) 817–823. doi:10.1109/7.705889.

[64] J.-M. Marin, P. Pudlo, C. P. Robert, R. J. Ryder, Approximate Bayesian computational methods, Statistics and Computing 22 (2011) 1167–1180. doi:10.1007/s11222-011-9288-2.

[65] Y. Song, C. Durkan, I. Murray, S. Ermon, Maximum likelihood training of score-based diffusion models, in: Advances in Neural Information Processing Systems 34, 2021, pp. 1415–1428.

[66] S. Zhong, W. Shen, T. Catanach, X. Huan, Goal-oriented Bayesian optimal experimental design for nonlinear models using Markov Chain Monte Carlo, arXiv preprint arXiv:2403.18072 (2024).

[67] W. Shen, X. Huan, Bayesian sequential optimal experimental design for nonlinear models using policy gradient reinforcement learning, Computer Methods in Applied Mechanics and Engineering 416 (2023) 116304. doi:10.1016/j.cma.2023.116304.

## Appendix A. Proof for Proposition 1

*Proof.* Starting from Eqn. (8), we have

$$
\begin{aligned}
& U_L(\mathbf{d};\boldsymbol{\lambda}_1) \le U_L(\mathbf{d};\boldsymbol{\lambda}_2) \\
\iff & \mathbb{E}_{\boldsymbol{\Theta},\mathbf{Y}|\mathbf{d}}\left[\ln \frac{q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda}_1)}{p(\boldsymbol{\theta})}\right] \le \mathbb{E}_{\boldsymbol{\Theta},\mathbf{Y}|\mathbf{d}}\left[\ln \frac{q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda}_2)}{p(\boldsymbol{\theta})}\right] \\
\iff & \mathbb{E}_{\boldsymbol{\Theta},\mathbf{Y}|\mathbf{d}}\left[\ln q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda}_1) - \ln p(\boldsymbol{\theta})\right] \le \mathbb{E}_{\boldsymbol{\Theta},\mathbf{Y}|\mathbf{d}}\left[\ln q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda}_2) - \ln p(\boldsymbol{\theta})\right] \\
\iff & \mathbb{E}_{\boldsymbol{\Theta},\mathbf{Y}|\mathbf{d}}\left[\ln q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda}_1) - \ln p(\boldsymbol{\theta}|\mathbf{y},\mathbf{d})\right] \le \mathbb{E}_{\boldsymbol{\Theta},\mathbf{Y}|\mathbf{d}}\left[\ln q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda}_2) - \ln p(\boldsymbol{\theta}|\mathbf{y},\mathbf{d})\right] \\
\iff & \mathbb{E}_{\boldsymbol{\Theta},\mathbf{Y}|\mathbf{d}}\left[\ln \frac{q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda}_1)}{p(\boldsymbol{\theta}|\mathbf{y},\mathbf{d})}\right] \le \mathbb{E}_{\boldsymbol{\Theta},\mathbf{Y}|\mathbf{d}}\left[\ln \frac{q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda}_2)}{p(\boldsymbol{\theta}|\mathbf{y},\mathbf{d})}\right] \\
\iff & -\mathbb{E}_{\mathbf{Y}|\mathbf{d}}\mathbb{E}_{\boldsymbol{\Theta}|\mathbf{Y},\mathbf{d}}\left[\ln \frac{p(\boldsymbol{\theta}|\mathbf{y},\mathbf{d})}{q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda}_1)}\right] \le -\mathbb{E}_{\mathbf{Y}|\mathbf{d}}\mathbb{E}_{\boldsymbol{\Theta}|\mathbf{d}}\left[\ln \frac{p(\boldsymbol{\theta}|\mathbf{y},\mathbf{d})}{q(\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda}_2)}\right] \\
\iff & \mathbb{E}_{\mathbf{Y}|\mathbf{d}}\left[D_{\mathrm{KL}}(p_{\boldsymbol{\theta}|\mathbf{y},\mathbf{d}} \,||\, q_{\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda}_1})\right] \ge \mathbb{E}_{\mathbf{Y}|\mathbf{d}}\left[D_{\mathrm{KL}}(p_{\boldsymbol{\theta}|\mathbf{y},\mathbf{d}} \,||\, q_{\boldsymbol{\theta}|\mathbf{y};\boldsymbol{\lambda}_2})\right],
\end{aligned}
$$

ending at Eqn. (9). The fourth line is from adding $\mathbb{E}_{\boldsymbol{\Theta},\mathbf{Y}|\mathbf{d}}\left[\ln p(\boldsymbol{\theta}) - \ln p(\boldsymbol{\theta}|\mathbf{y},\mathbf{d})\right]$, which is a constant, on both sides to the preceeding line. □

## Appendix B. Case 1

### *Appendix B.1. Hyperparameters*

The hyperparameters for Sec. 4.1 are given in Tables B.3 and B.4. The number of vOED-NFs transformation is also compared across $T = \{1, 3, 5, 7\}$.

| Hyperparameter | vOED-G |
|---|---|
| $N_{\mathrm{opt}}$ | 20000 |
| $N$ | 10000 |
| $N_{\mathrm{batch}}$ | 1000 |
| Initial learning rate | $10^{-2}$ |
| Learning rate decay | 0.99 |
| Network structure | {32, 32} |
| Training epochs | 301 |
| Activation | ReLU |

Table B.3: Case 1. Hyperparameters for vOED-G.

| Hyperparameter | vOED-NFs |
|---|---|
| $N_{\text{opt}}$ | 5000 / 10000 / 20000 / 50000 |
| $N$ | 10000 |
| $N_{\text{batch}}$ | 250 / 500 / 1000 / 2500 |
| Initial learning rate | $5 \times 10^{-3}$ / $5 \times 10^{-3}$ / $10^{-2}$ / $10^{-2}$ |
| Learning rate decay | 0.99 |
| Network structure (s & t) | {32, 32} |
| Training epochs | 301 |
| Activation | ELU |

Table B.4: Case 1. Hyperparameters for vOED-NFs.

*Appendix B.2. EIG estimated on training sample*

Figure B.14 shows the EIG evaluated using the same samples it used for optimization (i.e., it is 'testing' on the same samples that it used for 'training'). We see that in this case, smaller $N_{\text{opt}}$ can actually yield higher EIG estimates than Hi-NMC that uses $N_{\text{out}} = N_{\text{in}} = 20000$ samples, indicating an overfitting phenomenon where $\widehat{U}_L(d; \boldsymbol{\lambda}^*)$ could overestimate the EIG. Though in this case it does alter the location of the optimal design, the observation suggests the importance of evaluating EIG on a separate sample set than those used for optimization in order to reduce the bias in estimating the lower bound, especially when $N_{\text{opt}}$ is small.

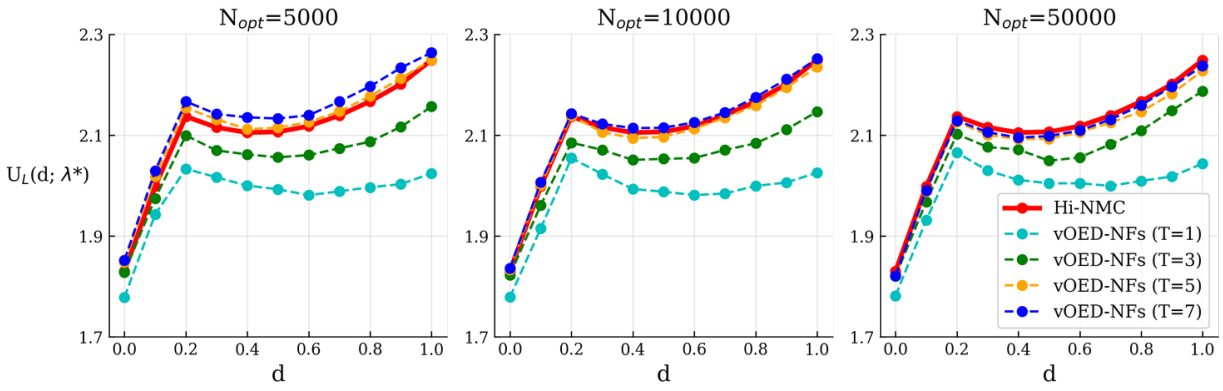



Figure B.14: Case 1. EIG under different number of transformations T and optimization sample sizes $N_{\text{opt}}$, evaluated using the same samples that it used for performing optimization.

## Appendix C. Case 2

*Appendix C.1. Hyperparameters*

The hyperparameters for Sec. 4.2 are given in Tables C.5 and C.6.

| Hyperparameter | vOED-GG |
|---|---|
| $N_{\text{opt}}$ | 50000 |
| $N$ | 10000 |
| $N_{\text{batch}}$ | 2048 |
| Initial learning rate | $5 \times 10^{-3}$ |
| Learning rate decay | 0.99 |
| Network structure | {64, 64} |
| Training epochs | 201 (Table 1) / 301 (Fig. 6) |
| Activation for vOED-GG | ReLU |

Table C.5: Case 2. Hyperparameters for vOED-GG.

| Hyperparam | vOED-NFs |
|---|---|
| $N_{\text{opt}}$ | 50000 |
| $N$ | 10000 |
| $N_{\text{batch}}$ | 2048 |
| Initial learning rate | $5 \times 10^{-3}$ |
| Learning rate decay | 0.99 |
| Network structure vOED-NFs (s & t) | {32, 32} |
| $T$ (numebr of transformations) | 4 |
| Training epochs | 201 (Table 1) / 301 (Fig. 7) |
| Activation for vOED-NFs | ELU |

Table C.6: Case 2. Hyperparameters for vOED-NFs.

## Appendix D. Case 3

*Appendix D.1. Forward model*

This section introduces the forward model used in Sec. 4.3. The model solve for e-coating can be described by three steps. First, the electric field within the bath is computed using the conservation of current density. For a constant current e-coating, a Poisson PDE with Robin boundary condition at the interface bath/film and Neumann condition at the anode is solved and described by the following equations:

$$\nabla \cdot \mathbf{j} = 0 \tag{D.1}$$

$$\mathbf{j} = \sigma_{\text{bath}} \nabla \phi \tag{D.2}$$

$$\phi|_{\Gamma} = R_{\text{film}}\, j_n \qquad \text{at the interface film-bath} \tag{D.3}$$

$$j_n = j_0 \qquad \text{at the anode} \tag{D.4}$$

where $\mathbf{j}$ is the current density, $j_n = \mathbf{j} \cdot \mathbf{n}$ is the normal component of the current density, $j_0$ is the prescribed density current at the anode, $\phi$ is the electrical potential, $\sigma_{\text{bath}}$ is the bath conductivity, $R_{\text{film}}$ is the coating film resistance, and $\Gamma$ represents the interface between the coating film and the bath. Second, the film deposition rate is computed using

$$\frac{dh}{dt} = C_v j_n, \tag{D.5}$$

where $h$ is the film thickness and $C_v$ is the Coulombic efficiency. Third, the film resistance is found by solving the following equation:

$$\frac{dR_{\text{film}}}{dt} = \rho(\mathbf{j})\frac{dh}{dt}, \tag{D.6}$$

where $\rho(\mathbf{j})$ is the film resistivity. The coulombic efficiency $C_v$ is assumed to be constant and the film resistivity $\rho(\mathbf{j})$ is a decreasing function of the current density.

The onset condition is critical for the prediction of the film deposition. Two criteria are used to evaluate the deposition onset. The first one considers a minimal value of the current density $j_{\min}$ that will trigger the deposition:

$$\frac{dh}{dt} = C_v\, j_n \text{ for } j > j_{\min}. \tag{D.7}$$

The second condition is a minimum charge condition $Q_{\min}$ and assumes that the deposition starts when the accumulative charge on the cathode reaches a minimum value as follows:

$$\frac{dh}{dt} = C_v\, j_n \text{ for } Q > Q_{\min}, \tag{D.8}$$

where the electric charge $Q$ is defined as $Q(t) = \int_t j_n dt$. The minimum charge can be expressed as a function of the constant current $j_0$, with $K$ a constant for a given bath and materials:

$$Q_{\min} = \frac{K^2}{j_0}. \tag{D.9}$$

The truncated normal priors for $\{C_v, j_{\min}, Q_{\min}\}$ are:

$$p(C_v) \sim p_{TN}(\mu = 7,\ \sigma = 0.5,\ l = 6,\ u = 8), \tag{D.10}$$

$$p(j_{\min}) \sim p_{TN}(\mu = 1,\ \sigma = 0.5,\ l = 0,\ u = 2), \tag{D.11}$$

$$p(Q_{\min}) \sim p_{TN}(\mu = 50, \sigma = 25,\ l = 0,\ u = 100), \tag{D.12}$$

where

$$p_{TN}(x;, \mu, \sigma, l, u) = \frac{1}{\sigma}\frac{\phi(\frac{x-\mu}{\sigma})}{\Phi(\frac{u-x}{\sigma}) - \Phi(\frac{l-x}{\sigma})}$$

for $x \in [l, u]$, and equals 0 otherwise. Here $\phi(\cdot)$ denotes the PDF for a standard normal distribution and $\Phi(\cdot)$ is its cumulative distribution function.

*Appendix D.2. Hyperparameters for the 10-dimensional case*

For $\mathbf{y}$ with dimension 10, Table D.7 lists the hyperparameters used for the lower and upper bound estimators in vOED-NFs.

| Hyperparam | vOED-NFs |
|---|---|
| $N_{\text{opt}}$ | 10000 / 80000 |
| $N$ | 10000 |
| $N_{\text{batch}}$ | 512 / 2048 |
| Initial learning rate | $5 \times 10^{-3}$ |
| Learning rate decay | 0.99 |
| Network structure (s & t) | {16, 16} |
| $T$ | 5 |
| Training epochs | 301 |
| Activation for vOED-NFs | ELU |

Table D.7: Case 3. Hyperparameters for the 10-dimensional case, both lower and upper bound estimators.

*Appendix D.3. Hyperparameters for the 50-dimensional case*

For $\mathbf{y}$ with dimension 50, Table D.8 lists the hyperparameters used for the lower bound estimators in vOED-G, and Table D.9 lists the hyperparameters used for the lower and upper bound estimators in vOED-NFs, without the LSTM summary network.

| Hyperparameter | vOED-G |
|---|---|
| $N_{\text{opt}}$ | 10000 |
| $N$ | 10000 |
| $N_{\text{batch}}$ | 256 |
| Initial learning rate | $1 \times 10^{-3}$ |
| Learning rate decay | 0.99 |
| Network structure | {32, 32} |
| Training epochs | 301 |
| Activation for vOED-G | ReLU |

Table D.8: Case 3. Hyperparameters for vOED-G for the 50-dimensional case, for the lower bound estimator.

| Hyperparameter | vOED-NFs |
|---|---|
| $N_{\text{opt}}$ | 10000 / 80000 |
| $N$ | 10000 |
| $N_{\text{batch}}$ | 512 / 2048 |
| Initial learning rate | $1 \times 10^{-3}/5 \times 10^{-3}$ |
| Learning rate decay | 0.99 |
| Network structure (s & t) | {16, 16} |
| $T$ | 5 |
| Training epochs | 301 |
| Activation for vOED-NFs | ELU |

Table D.9: Case 3. Hyperparameters for vOED-NFs for the 50-dimensional case, for both lower and upper bound estimators without LSTM summary network.

Table D.10 lists the hyperparameters for vOED-NFs with the LSTM summary network. The initial learning rates reported in both tables with or without LSTM empirically yield the highest lower bound estimates among the learning rate values $\{5 \times 10^{-4}, 1 \times 10^{-3}, 5 \times 10^{-3}, 1 \times 10^{-2}\}$. To get a dimension reduced 10-dimensional $\mathbf{y}'$, we first feed $\mathbf{y}$ as input to a bidirectional LSTM network, and then concatenate the resulting context vectors, which is further fed into the final embedding network emanating $\mathbf{y}'$.

*Appendix D.4. Posteriors from vOED-NFs with LSTM summary network and $N_{opt} = 80000$*

Figure D.15 plots vOED-NFs posteriors when using LSTM summary network and $N_{\text{opt}} = 80000$. The posteriors match the SMC posteriors (reference) better compared to those trained with smaller $N_{\text{opt}}$ and when $\mathbf{y}$ is higher dimensional (i.e., from Fig. 12).

| Hyperparameters | vOED-NFs-LSTM |
|---|---|
| $N_{\text{opt}}$ | 10000 / 80000 |
| $N$ | 10000 |
| $N_{\text{batch}}$ | 512 / 2048 |
| Initial learning rate | $5 \times 10^{-3}/1 \times 10^{-2}$ |
| Learning rate decay | 0.99 |
| Network structure (s & t) | {16, 16} |
| $T$ | 5 |
| Training epochs | 301 |
| Activation for vOED-NFs-LSTM | ELU |
| LSTM-n_feature | 1 |
| LSTM-hidden_dim | 20 |
| LSTM-num_layers | 1 |
| Embedding network structure | 40, 20 |
| Activation for embedding network | ELU |

Table D.10: Case 3. Hyperparameters for vOED-NFs for the 50-dimensional case, for lower bound estimator with LSTM summary network.

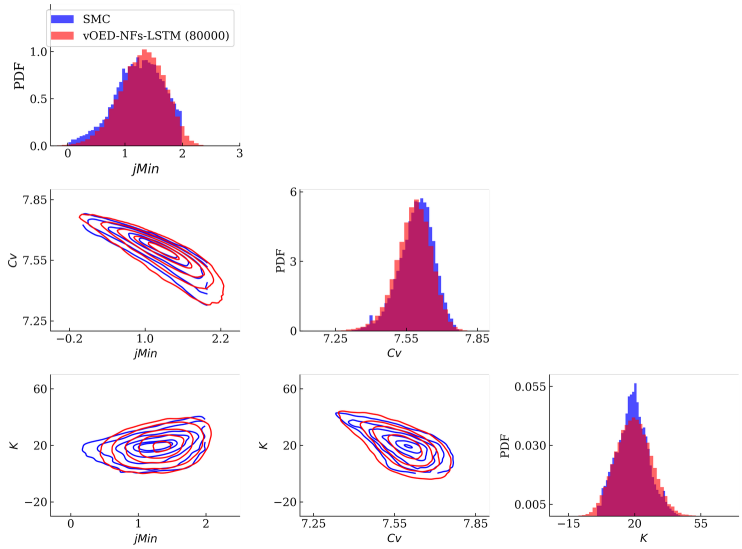



Figure D.15: Case 3. Marginal and pairwise joint posterior distributions obtained using SMC (high-quality reference) and vOED-NFs-LSTM with $N_{\text{opt}} = 80000$.

## Appendix E. Case 4

### *Appendix E.1. Hyperparameters*

The hyperparameters for Sec. 4.4 are given in Table E.11. Note LB-KLD [15] is computed using 10000 prior samples, each associated with 3 data samples such that the overall number of forward model runs is 30000. For vOED-NFs, we keep the total number of forward model runs at 30000 ($N_{\text{opt}} = 20000$ and $N = 10000$).

| Hyperparameters | vOED-NFs |
|---|---|
| $N_{\text{opt}}$ | 20000 |
| $N$ | 10000 |
| $N_{\text{batch}}$ | 2048 |
| Initial learning rate | $1 \times 10^{-2}$ |
| Learning rate decay | 0.99 |
| Network structure (s & t) | {16, 16} |
| $T$ | 4 |
| Training epochs | 51 |
| Activation for vOED-NFs | ELU |

Table E.11: Case 4. Hyperparameters for vOED-NFs.