HIGH-DIMENSIONAL BAYESIAN INVERSION WITH BLACK-BOX SIMULATORS

Dhruv V. Patel
Stanford University
Stanford, CA 94305, USA
dvpatel@stanford.edu

Jonghyun Lee
University of Hawaii at Manoa
Honolulu, HI 96822, USA
jonghyun.harry.lee@hawaii.edu

Matthew W. Farthing
U.S. Army ERDC
Vicksburg, MS 39183, US
matthew.w.farthing@erdc.dren.mil

Peter K. Kitanidis
Stanford University
Stanford, CA 94305, USA
peterk@stanford.edu

Eric F. Darve
Stanford University
Stanford, CA 94305, USA
darve@stanford.edu

ABSTRACT

Efficient black-box forward model simulators are fundamental to various applications in hydrology and subsurface characterization due to their accuracy in mimicking complex subsurface processes. However, their inability to accommodate differentiation through them poses a significant challenge, rendering them incompatible with current state-of-the-art gradient-based Bayesian inference techniques such as Hamiltonian Monte Carlo or Variational Inference. In this study, we address this critical issue by introducing a modular approach that combines black-box variational inference (BBVI) with deep generative priors, enabling efficient and accurate high-dimensional Bayesian inversion in a black-box setting. Our methodology introduces a novel gradient correction term and sampling strategy tailored for BBVI, which collectively diminish gradient errors by several orders of magnitude, even with minimal batch sizes. This results in significant reduction in number of forward model evaluations required. Furthermore, integrating our method with Generative Adversarial Network (GAN)-based priors enables the solution of high-dimensional inverse problems. We validate our algorithm's effectiveness on multiple physics-based inverse problems using both simulated and experimental data (includng the problem of hydraulic tomography). In comparison to Markov Chain Monte Carlo (MCMC) methods, our approach consistently delivers superior accuracy and substantial improvements in both statistical and computational efficiency, often by an order of magnitude.

***K*eywords** Bayesian inference · Inverse problems · Variational inference · Black-box simulators · Uncertainty quantification · Hydraulic tomography

1 Introduction

Scientific simulators play a pivotal role in water resources research by offering powerful tools for modeling and simulating complex phenomena in surface and subsurface hydrology [1, 2, 3, 4]. Their importance lies in their ability to bridge theoretical insights with real-world observations from laboratory-scale experiments, field sites, and remote sensing, allowing researchers to explore and test hypotheses in a controlled and reproducible environment. These simulators facilitate the investigation of complex hydrological systems across various scales, from molecular interactions [5, 6], Darcy-scale flow and transport [7], multi-phase flow in regional aquifers [8, 9], rainfall runoff in watersheds [10], to global climate patterns [11, 12], enabling scientists to gain deep insights into underlying flow and transport mechanisms and behaviors. Moreover, they serve as invaluable platforms for scenario testing, risk assessment, and decision-making in various fields associated with water resources management ranging from agriculture engineering and environmental science to healthcare [13] and beyond. By providing a virtual sandbox for experimentation and analysis, scientific simulators drive innovation, advance knowledge, and contribute significantly to solving pressing real-world challenges.

The ever-increasing computational power of modern hardware, coupled with the expressiveness of advanced programming languages, has facilitated the development of more and more complex, high-fidelity simulators. These forward model simulators excel in accurately modeling multi-scale, multi-physics, and multi-phase phenomena of interest, finding applications across a spectrum of scientific domains, from computational fluid dynamics [14] to surface-subsurface flow and transport [15, 8] and beyond [16, 17, 18, 19]. Their versatility equips researchers with the means to comprehensively describe and predict system behaviors across a wide array of conditions.

While these sophisticated "black-box simulators" deliver high-fidelity solutions, unfortunately, they are ill-suited when it comes to integrating with subsequent simulation-based applications such as sensitivity analysis [20], parameter calibration and uncertainty quantification [21, 22] and optimization [23] that require efficient gradient computation, due to their inherent black-box nature [24]. To make ideas more concrete consider a representative problem in modeling groundwater flow of predicting hydraulic head from hydraulic conductivity. This prediction problem entails solving partial differential equation(s). There are many sophisticated numerical solvers developed over the years to solve this problem. The MODFLOW software [25] developed by the U.S. Geological Survey is one promiment example of this. This open-source software code was developed in 1980s in Fortran and is considered an international standard for simulating and predicting groundwater conditions and groundwater/surface-water interactions and it is still routinely being used by hydrogeologists to simulate the flow of groundwater through aquifers. Unfortunately, the production version of it does not support easy integration of modern automatic differentiation libraries for fast gradient computation.

Here, we refer to such forward model simulator or computer program as "black-box simulator". These forward model simulators (denoted as $\boldsymbol{f}$) take various problem parameters $\boldsymbol{x}$ as input, including geometric descriptions of the domain, boundary conditions, initial conditions, and hydraulic conductivity distributions, and produce corresponding solutions (hydraulic head) $\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x})$. However, their limitations become apparent in their inability to easily provide gradients of the simulator's output with respect to its inputs, i.e., $\partial \boldsymbol{y}/\partial \boldsymbol{x}$ remains unavailable. This constraint primarily arises from the complex and optimized nature of these simulators, which are often developed over years by multiple researchers, utilizing legacy programming languages and heavily optimized for forward computations, sometimes with parallel implementations. These characteristics make it challenging to seamlessly integrate these simulators with modern automatic differentiation libraries or require substantial manual intervention to incorporate gradient functionality in them.

The inability to perform efficient Bayesian inference with such high-fidelity black-box simulators poses a significant barrier to scientific progress in hydrology [26, 27, 24]. Bayesian inference, particularly when applied to high-fidelity models, plays a pivotal role in facilitating precise parameter estimation [28], conducting robust uncertainty quantification[29], guiding informed decision-making processes and risk assessment [30], supporting rigorous hypothesis testing, validating complex models, and optimizing experimental designs [31] across a wide spectrum of applications in geophysics[32]. Therefore, the development of specialized inference algorithms tailored to handle the intricacies of such black-box simulators is key to unlocking numerous possibilities in these crucial applications.

Furthermore, in these applications the parameter to be inferred, $\boldsymbol{x}$, (the nodal values of hydraulic conductivity) can often be very high-dimensional (10^4–10^6). This high dimensionality emanates from the fine spatio-temporal discretization required by numerical methods (such as finite element or finite difference) to accurately represent the highly heterogeneous hydraulic conductivity fields commonly found in practice. Current state-of-the-art inference techniques such as Hamiltonian Monte Carlo (HMC) [33, 34], Langevin Dynamics [35], or variational inference (VI) [36] struggle and often fail to converge in such high-dimensional parameter space. This is popularly known as the '*curse of dimensionality*'. The difficulty manifests as long mixing times and larger auto-correlations between successive samples of Markov chains [37, 38] for different MCMC-based inference methods. While variational inference has been shown to scale better than MCMC [39, 40], the curse of dimensionality is still a challenge. The number of parameters that must be optimized, when approximating posterior distributions, proliferates as the dimensionality of the inverse problem increases [41]. This '*curse of dimensionality*' further constrains the applicability of advanced inference algorithms in practical science and engineering contexts, thereby impeding scientific progress.

In addressing the challenges posed by high-dimensional Bayesian inversion with black-box forward model simulators, there arises a critical need for innovative inference algorithms that can efficiently navigate the complexities of these models while leveraging their black-box nature. Our approach aims to achieve accurate and efficient Bayesian inversion by integrating seamlessly with the intrinsic characteristics of such simulators, all while maintaining computational efficiency.

To tackle this challenge, in this manuscript, we propose a strategy that proposes improvements to traditional black-box variational inference (BBVI) and combines it with deep generative priors. BBVI, initially introduced by Ranganath et al. [42], allows for valuable information extraction from black-box forward models without requiring direct access to their gradients. This capability is crucial for handling complex "black-box" simulators efficiently without requiring its gradient. However, it is well known that the gradient obtained using the traditional BBVI algorithm suffers from high

variance [43]. This problem can be overcome by using more (thousands to tens of thousands of) samples to approximate the gradient. While this may be an acceptable solution strategy for some computer vision and machine learning-based applications, it is unacceptable for PDE-based inverse problems (like the ones considered in this manuscript). This is because each sample means one PDE solve and at each iteration of the optimization of the BBVI algorithm one needs to approximate the gradient and if one needs (tens of) thousands of samples to accurately approximate the true gradient then the total number of samples required could be in millions (since the total number of samples required for performing Bayesian inversion = number of samples required to approximate the gradient at each iteration $\times$ the number of iterations). This is prohibitively expensive for subsurface characterization applications limiting the direct use of BBVI for such applications.

To overcome this challenge, we propose a novel gradient correction term to the BBVI gradient. As shown in Section Section 4, this significantly reduces the variance of the gradient estimate and accurately approximates the gradient with a minimal number of samples required. This is particularly promising for PDE-based problems due to its low computational cost. This gradient correction term, however, requires computing finite difference gradient approximation which scales with the dimensionality of the problems. Since many physics-based problems are high-dimensional in nature due to fine spatio-temporal discretization, this eventually increases the total computational cost. To tackle this issue, we concurrently leverage recently proposed generative adversarial network (GAN)-based priors. As demonstrated in studies by Laloy et al. [44] and Patel et al. [45], this data-driven prior enables the inference algorithm to operate within a lower-dimensional latent space of the pre-trained generator of a GAN. This strategic combination of finite difference-based gradient correction and GAN-based prior not only reduces the overall computational cost but also speeds up the inference process, paving the way for more effective scientific analyses and decision-making processes for hydrologists based on the solution of complex high-dimensional Bayesian inverse problem.

Specifically, our contributions are outlined as follows:

- We propose a modular methodology that harnesses modified BBVI and GAN-based priors to tackle *high-dimensional* Bayesian inverse problems associated with *black-box forward model simulators* in an efficient manner.
- Recognizing the sample inefficiency of traditional BBVI gradients, initially introduced by Ranganath et al. [42], we introduce a novel gradient correction term and sampling strategy. This substantially reduces gradient errors across problems of varying dimensionalities, even with very small batch sizes, resulting in huge computational savings.
- We validate the efficacy of our proposed method through extensive numerical experiments on both synthetic and real-world datasets. Our approach consistently outperforms traditional Markov Chain Monte Carlo (MCMC) algorithms on both accuracy as well as efficiency, for the same computational budget.

The remainder of this paper is organized as follows. Section 2 sets up the problem of interest, and provides a brief background on Bayesian inference, variational inference, and Generative Adversarial Networks (GANs). We introduce Black-box Variational Inference (BBVI) and proposed improvements in Section 3. In Section 4, we apply the proposed algorithm to solve different inverse problems involving both synthetic and experimental data. Finally, we conclude in Section 5.

2 Background

2.1 Bayesian inference

We begin with the definition of the forward/direct model as follows:

$$\boldsymbol{f} : \boldsymbol{x} \mapsto \boldsymbol{y}, \quad \boldsymbol{x} \in \Omega_{\mathcal{X}} \subseteq \mathbb{R}^{N_{\mathcal{X}}}, \quad \boldsymbol{y} \in \Omega_{\mathcal{Y}} \subseteq \mathbb{R}^{N_{\mathcal{Y}}}, \tag{1}$$

where $\boldsymbol{y}$ represents the response/solution corresponding to input parameters $\boldsymbol{x}$ for the given forward model $\boldsymbol{f}$. For example, a representative groundwater example is that of predicting hydraulic pressure. In this case, the input parameters $\boldsymbol{x}$ may include the nodal values of hydraulic conductivity or boundary conditions and the response/solution $\boldsymbol{y}$ represents the nodal values of the hydraulic head. Forward model $\boldsymbol{f}$ represents the discretized solution operator of a PDE (Darcy's flow). So, for given input parameters $\boldsymbol{x}$ one can accurately compute $\boldsymbol{y}$ using any reasonable numerical methods for solving such PDE. This problem is well defined and is relatively easy to solve. However, in many applications we are interested in the task of subsurface characterization where we are interested in inferring the nodal values of hydraulic conductivity ($\boldsymbol{x}$) in entire domain from sparse and noisy measurement of hydraulic head ($\boldsymbol{y}$). This is in general an ill-posed problem in the sense of Hadamard. This means either (i) the solution does not exist or (ii) the solution is not unique or (iii) the solution does not depend continuously on data. Bayesian inference offers a systematic approach to solving such inverse problems while providing quantified uncertainty estimates.

Within the Bayesian framework, we model the inferred field and observations as realizations of random variables $\mathcal{X}$ and $\mathcal{Y}$, respectively. We assume that measurements are corrupted by additive noise, i.e., $\tilde{\boldsymbol{y}} = \boldsymbol{y} + \boldsymbol{\eta}$, where $\boldsymbol{\eta}$ represents the modeling/measurement error and p_η its distribution. Additionally, we introduce a prior distribution $p^{\text{prior}}_{\mathcal{X}}(\boldsymbol{x})$, which encapsulates our knowledge about $\mathcal{X}$ *prior* to observing $\tilde{\boldsymbol{y}}$. Alongside this, we define the likelihood of observing the measurement $\tilde{\boldsymbol{y}}$ given $\mathcal{X} = \boldsymbol{x}$ as $p^{\text{like}}_{\mathcal{Y}}(\tilde{\boldsymbol{y}}|\boldsymbol{x})$. Applying Bayes' rule yields the posterior distribution of $\mathcal{X}$ given the observed measurement $\tilde{\boldsymbol{y}}$:

$$p^{\text{post}}_{\mathcal{X}}(\boldsymbol{x}|\tilde{\boldsymbol{y}}) = \frac{p^{\text{like}}_{\mathcal{Y}}(\tilde{\boldsymbol{y}}|\boldsymbol{x}) p^{\text{prior}}_{\mathcal{X}}(\boldsymbol{x})}{p_{\mathcal{Y}}(\tilde{\boldsymbol{y}})} \propto p_\eta(\tilde{\boldsymbol{y}} - \boldsymbol{f}(\boldsymbol{x})) p^{\text{prior}}_{\mathcal{X}}(\boldsymbol{x}). \tag{2}$$

Here, $p_{\mathcal{Y}}(\tilde{\boldsymbol{y}})$ represents the evidence. The likelihood term, $p^{\text{like}}_{\mathcal{Y}}(\tilde{\boldsymbol{y}}|\boldsymbol{x}) = p_\eta(\tilde{\boldsymbol{y}} - \boldsymbol{f}(\boldsymbol{x}))$, incorporates physical knowledge through the forward operator $\boldsymbol{f}$. Statistical/Bayesian inference involves characterizing the posterior distribution by computing the expectation of a quantity of interest (QoI) $s(\boldsymbol{x})$ with respect to this posterior:

$$\mathop{\mathbb{E}}_{\boldsymbol{x} \sim p^{\text{post}}_{\mathcal{X}}(\boldsymbol{x}|\tilde{\boldsymbol{y}})} [s(\boldsymbol{x})] = \int_{\Omega_{\mathcal{X}}} s(\boldsymbol{x}) p^{\text{post}}_{\mathcal{X}}(\boldsymbol{x}|\tilde{\boldsymbol{y}}) d\boldsymbol{x}. \tag{3}$$

For example, if we are interested in inferring the mean of the posterior distribution then we set $s(\boldsymbol{x}) = \boldsymbol{x}$. For the subsurface characterization problem, the estimation of such posterior mean can give us an idea of what the mean of all the possible value of hydraulic conductivity look like for a given (noisy) measurement of hydraulic head.

2.2 Generative Adversarial Networks

In order to solve high-dimensional Bayesian inverse problems, like the one encountered in subsurface characterization, we will leverage recent advancements of deep generative priors in Bayesian inference. Specifically, we will be using deep generative adversarial network (GAN)-based priors. Here, we provide a brief overview of GAN before showcasing how we use them in our framework in Section 3.

Generative adversarial networks [46] are generative models consisting of two sub-networks: a generator and a discriminator (also known as the critic). GANs are trained *adversarially*: the generator tries to deceive the discriminator while the discriminator tries to distinguish between '*fake*' samples generated from the generator and '*true*' samples available from the target distribution. The generator and critic play an adversarial '*game*' between them with the ultimate goal of generating new realizations from an underlying distribution, the prior distribution $p^{\text{prior}}_{\mathcal{X}}$ in this case. Let the generator network G, parameterized by $\boldsymbol{\theta}$, map the *latent* variable $\boldsymbol{z} \in \Omega_{\mathcal{Z}} \subseteq \mathbb{R}^{N_{\mathcal{Z}}}$ to the target variable $\boldsymbol{x}$, i.e., $G(\cdot, \boldsymbol{\theta}) : \Omega_{\mathcal{Z}} \to \Omega_{\mathcal{X}}$. Herein, we refer to $\boldsymbol{z}$ as the *latent* variable. Typically, $\boldsymbol{z}$ is sampled from a simple distribution $p_{\mathcal{Z}}$, like the multivariate standard normal distribution. Moreover, the latent dimension $N_{\mathcal{Z}}$ is typically chosen to be much smaller than the ambient dimension $N_{\mathcal{X}}$, i.e., $N_{\mathcal{Z}} \ll N_{\mathcal{X}}$. Thus, GANs are endowed with dimension reduction capabilities and the generator G serves as a map from the low-dimensional latent space to the high-dimensional ambient space. On the other hand, the discriminator D, parameterized by $\boldsymbol{\phi}$ such that $D(\cdot, \boldsymbol{\phi}) : \Omega_{\mathcal{X}} \to \mathbb{R}$, tries to differentiate between realizations drawn from $p^{\text{prior}}_{\mathcal{X}}$ and those generated by the generator.

The parameters $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ of the generator and the discriminator networks, respectively, are obtained through the min-max optimization of an appropriate loss function, say $\mathcal{L}_{\text{GAN}}$, i.e.

$$(\boldsymbol{\theta}^*, \boldsymbol{\phi}^*) = \arg\min_{\boldsymbol{\theta}} \Big(\arg\max_{\boldsymbol{\phi}} \mathcal{L}_{\text{GAN}}(\boldsymbol{\theta}, \boldsymbol{\phi}) \Big). \tag{4}$$

Different types of GANs will use different loss functions $\mathcal{L}_{\text{GAN}}$; interested readers may refer to [47, 48, 49] for an overview. It is important to note that training a GAN requires realizations of $p^{\text{prior}}_{\mathcal{X}}$, therefore, we assume that n_{data} independent and identically distributed (iid) realizations of $\boldsymbol{x}$ from $p^{\text{prior}}_{\mathcal{X}}$ are available, which we herein denote using $\mathcal{S} = \left\{ \boldsymbol{x}^{(i)} \right\}_{i=1}^{n_{\text{data}}}$ and refer to $\mathcal{S}$ as the prior dataset.

2.3 Variational Inference

In typical inverse problems encountered in subsurface characterization, the dimensionality ($N_{\mathcal{X}}$) of the parameter vector $\boldsymbol{x}$ is very high. This is due to the fact that fine spatio-temporal resolution is required to resolve highly heterogenous subsurface fields. This results in $N_{\mathcal{X}}$ to be of the order of (10^3–10^6). This poses significant challenges while performing Bayesian inference. This is because for such high-dimensional problems, computing the denominator of Eq. (2) requires computing high-dimensional integral over $N_{\mathcal{X}}$-dimensional space, which is intractable both analytically and numerically for most practical (non-conjugate) problems. Furthermore, once the posterior is obtained, computing posterior QoIs using Eq. (3) is also intractable for the same reasons.

To overcome this challenge of high-dimensional integration in Bayesian inference, various techniques have been developed. Broadly speaking, these techniques can be classified as (1) Sampling/Markov-chain Monte Carlo (MCMC)-based

techniques and (2) Approximation/Variational inference-based techniques. While MCMC-based techniques provide helpful theoretical convergence guarantees and also provide unbiased estimates, they suffer from slow convergence and for many high-dimensional practical problems require an unacceptably large number of samples rendering it impractical.

Variational inference (VI) on the other hand offers an elegant solution for Bayesian inference. It sidesteps the problem of high-dimensional integration (and hence intractable denominator—evidence term) by directly approximating the posterior distribution with a tractable variational distribution. In VI, we define a variational family of distributions $q_\lambda(\boldsymbol{x})$ and determine the optimal values of variational parameters $\boldsymbol{\lambda}$ such that the variational distribution closely approximates the true posterior distribution. This "closeness" is typically defined via some divergence between the true posterior distribution $p_{\mathcal{X}}^{\text{post}}(\boldsymbol{x}|\tilde{\boldsymbol{y}})$ and the variational distribution $q_\lambda(\boldsymbol{x})$:

$$\boldsymbol{\lambda}^* = \arg\min_{\boldsymbol{\lambda}} d(q_\lambda(\boldsymbol{x}) \| p_{\mathcal{X}}^{\text{post}}(\boldsymbol{x}|\tilde{\boldsymbol{y}})) \tag{5}$$

The Kullback-Leibler (KL) divergence is a popular choice for d but other divergence measures have also been used [50]. Thus, variational Bayesian inference converts the problem of posterior sampling into an equivalent optimization problem. Once $\boldsymbol{\lambda}^*$ has been determined, $q_{\boldsymbol{\lambda}^*}(\boldsymbol{x})$ serves as an approximation to $p_{\mathcal{X}}^{\text{post}}(\boldsymbol{x}|\tilde{\boldsymbol{y}})$ and can be repeatedly sampled without additional likelihood evaluations to obtain as many posterior samples as required — unlike MCMC-based methods. As a result, variational Bayesian inference offers a computationally efficient alternative to MCMC sampling in many cases. The performance of variational Bayesian inference relies on the *a priori* chosen parameterized family of distribution $q_\lambda(\boldsymbol{x})$ being capable of approximating the posterior distributions, which can have a complex shape. This approximation may be difficult to achieve using standard distribution families like mixture models. Moreover, the computational effort of the optimization problem in Eq. (5) increases as the dimension of $\boldsymbol{\lambda}$ increases, which is expected to happen as the dimensionality of the inferred field ($N_{\mathcal{X}}$) of the inverse problem grows.

Unfortunately, finding $\boldsymbol{\lambda}^*$ by directly optimizing Eq. (5) is not possible. As computing the divergence (objective function) itself requires the very posterior ($p_{\mathcal{X}}^{\text{post}}(\boldsymbol{x}|\tilde{\boldsymbol{y}})$) we are trying to approximate. Hence, instead, an evidence lower bound (ELBO) of the above divergence is computed and is maximized to find the optimal variational parameters. For example, while using KL divergence in Eq. (5) we can find the optimal variational parameters by maximizing the following ELBO

$$\boldsymbol{\lambda}^* = \arg\max_{\boldsymbol{\lambda}} \mathcal{L}(\boldsymbol{\lambda}), \qquad \text{where } \mathcal{L}(\boldsymbol{\lambda}) = \text{ELBO} = \mathop{\mathbb{E}}_{q_\lambda} \left[\log p(\boldsymbol{x}, \tilde{\boldsymbol{y}}) - \log q_\lambda(\boldsymbol{x})\right]. \tag{6}$$

Here $p(\boldsymbol{x}, \tilde{\boldsymbol{y}}) = p(\tilde{\boldsymbol{y}}|\boldsymbol{x})p(\boldsymbol{x}) = p_\eta(\tilde{\boldsymbol{y}} - \boldsymbol{f}(\boldsymbol{x}))p_{\mathcal{X}}^{\text{prior}}(\boldsymbol{x})$. For a detailed derivation of an ELBO from Eq. (5) and other relevant information regarding variational inference please refer to [36].

We acknowledge that Laplace approximation [51, 52] is another widely-used method that transforms the inference problem into an optimization problem. Specifically, Laplace approximation involves creating a local Gaussian approximation of the posterior near the MAP (maximum-a-posteriori) point. While this method is relatively simple and direct, it is unsuitable for our scenario due to our operation within a black-box setting. Traditional Laplace approximation algorithms require differentiation through the forward model, making them impractical for our purposes.

Furthermore, even if differentiation through the forward model were possible, Laplace approximation struggles with approximating multi-modal distributions. It is highly sensitive to the initial guess and the effectiveness of the optimizer due to its local nature. In cases of multi-modal posterior distributions, Laplace approximation tends to only approximate one of the modes. In contrast, the proposed BBVI method offers a more comprehensive approximation of the entire target posterior distribution. By minimizing the KL divergence between the variational distribution and the target distribution, BBVI can better approximate the entire posterior distribution. We visually demonstrate this in the results section in Section 4.1.

2.4 Black-box Variational Inference (BBVI)

Variational inference (VI) in Bayesian inference involves optimizing the evidence lower bound (ELBO) (Eq. (6)). This involves computing its gradient of ELBO with respect to the variational parameters $\nabla_{\boldsymbol{\lambda}}\mathcal{L}$ and updating the variational parameters with gradient ascent with any appropriate gradient-based optimizers.

However, when dealing with non-differentiable simulators, traditional VI methods face significant challenges. As for such simulators since forward model $\boldsymbol{f}$ is non-differentiable, the resulting likelihood term is non-differentiable and as a result the joint distribution $p(\boldsymbol{x}, \tilde{\boldsymbol{y}})$ in Eq. (6) cannot be differentiated, thus limiting the use of any gradient-based optimizers. To address this, we turn to black-box variational inference (BBVI), initially introduced by Ranganath et al. [42] for latent variable models, and adapt it for physics-based inverse problems. BBVI allows for inference without the need to differentiate through the forward model.

The core idea of the BBVI is the do integration by parts of the gradient equation of the ELBO. This results in the following BBVI gradient:

$$\nabla_{\boldsymbol{\lambda}}^{\text{BBVI}}\mathcal{L} = \mathbb{E}_{\boldsymbol{x}\sim q_\lambda(\boldsymbol{x})}\nabla_{\boldsymbol{\lambda}}\log q_\lambda(\boldsymbol{x})\Big\{\log p_{\mathcal{X}}^{\text{prior}}(\boldsymbol{x}) + \log p_{\eta}^{\text{like}}(\tilde{\boldsymbol{y}} - \boldsymbol{f}(\boldsymbol{x})) - \log q_\lambda(\boldsymbol{x})\Big\} \tag{7}$$

The detailed derivation of this equation is provided in Appendix A.

An important thing to note here is that this gradient computation does not require direct access to the gradient of the forward model $\boldsymbol{f}$, making it suitable for black-box scenarios. Thus, with BBVI the gradient of the expectation (in Eq. (6)) is converted to the expectation of the gradient (in Eq. (7)). To approximate this expectation, via Monte Carlo (MC)-based sampling, i.e., $\nabla_{\boldsymbol{\lambda}}^{\text{BBVI}}\mathcal{L}$ is approximated as follows:

$$\nabla_{\boldsymbol{\lambda}}^{\text{BBVI}}\mathcal{L} \approx \frac{1}{N}\sum_{i=1}^{N}\nabla_{\boldsymbol{\lambda}}\log q_\lambda(\boldsymbol{x})\Big\{\log p_{\mathcal{X}}^{\text{prior}}(\boldsymbol{x}_i) + \log p_{\eta}^{\text{like}}(\tilde{\boldsymbol{y}} - \boldsymbol{f}(\boldsymbol{x}_i)) - \log q_\lambda(\boldsymbol{x}_i)\Big\}, \qquad \boldsymbol{x}_i \sim q_\lambda(\boldsymbol{x}) \tag{8}$$

where N is the batch size (number of samples used to approximate the expectation in Eq. (7).

It's worth noting that the approach employed in deriving the BBVI gradient, known as the "log-derivative trick", finds applications across various machine learning contexts under different names such as REINFORCE in reinforcement learning and the likelihood ratio method. The central idea remains consistent: approximating the gradient of an expectation as an expectation of a gradient [12, 53, 43].

3 Improved BBVI for High-Dimensional Physics-based Bayesian Inversion

The effectiveness of the Black-Box Variational Inference (BBVI) gradient estimator for the Evidence Lower Bound (ELBO), as shown in Eq. (7), is well-recognized, yet it is also known to suffer from high variance [43, 54, 55]. This high variance characteristic necessitates a substantial (hundreads to thousands) number of samples (N) to achieve gradients with acceptable accuracy. In the context of high-dimensional physics-based inverse problems, this poses a significant challenge as it demands solving a large number of computationally intensive Partial Differential Equations (PDEs) representing the forward model at each optimization iteration. Consequently, this leads to substantial computational overhead as the total number of PDE solves required = batch size (N) $\times$ number of iterations. Since, for a typical BBVI problem, the batch size could be $\mathcal{O}(10^2 - 10^3)$ and the number of iterations could be $\mathcal{O}(10^3)$. The total number of PDE solves required is very high. This is one of the main reason impeding the practical adoption of BBVI for PDE-based Bayesian inversion tasks.

The limitations of BBVI in this domain highlight the critical need for strategies to mitigate its computational burden and enhance its scalability for high-dimensional problems. Addressing these challenges not only improves the efficiency of Bayesian inference but also expands the applicability of sophisticated simulation tools in tackling real-world scientific and engineering challenges. In the subsequent sections, we present novel approaches and optimizations tailored to alleviate the computational demands of BBVI in high-dimensional physics-based Bayesian inversion, paving the way for more accessible and efficient probabilistic inference in complex systems.

3.1 Gradient correction

By exploiting the structure of the assumed variational distribution, we can estimate this gradient more accurately by reducing the variance. We do this by first assuming the variational distribution to be multivariate Gaussian, i.e., $q_\lambda(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{I\sigma^2})$ with $\boldsymbol{\lambda} = [\boldsymbol{\mu}, \boldsymbol{\gamma}]$; where, $\boldsymbol{x}, \boldsymbol{\mu}, \boldsymbol{\gamma} \in N_{\mathcal{X}}$, $\boldsymbol{\gamma} = [\gamma_i = \log(\sigma_i)]_{i=1}^{N_{\mathcal{X}}}$ and $\boldsymbol{I\sigma^2}$ indicates covariance matrix with non-zero diagonal entries and zero off-diagonal entries. Further, we introduce finite difference gradient correction term (and a special sampling strategy complementing this).

Let $l(\boldsymbol{x}) = \log p_{\mathcal{X}}^{\text{prior}}(\boldsymbol{x}) + \log p_{\eta}^{\text{like}}(\tilde{\boldsymbol{y}} - \boldsymbol{f}(\boldsymbol{x})) - \log q_\lambda(\boldsymbol{x})$. We can now rewrite Eq. (7) as

$$\nabla_{\boldsymbol{\lambda}}^{\text{BBVI}}\mathcal{L} = \mathop{\mathbb{E}}_{q_\lambda(\boldsymbol{x})}\big[\nabla_{\boldsymbol{\lambda}}\log q_\lambda(\boldsymbol{x})l(\boldsymbol{x}) - l'_{FD}(\boldsymbol{\lambda})^T(\boldsymbol{x} - \boldsymbol{\lambda})\big] + \mathop{\mathbb{E}}_{q_\lambda(\boldsymbol{x})}\big[\nabla_{\boldsymbol{\lambda}}\log q_\lambda(\boldsymbol{x})l'_{FD}(\boldsymbol{\lambda})^T(\boldsymbol{x} - \boldsymbol{\lambda})\big], \tag{9}$$

where $\boldsymbol{l'_{FD}}(\boldsymbol{\lambda})$ represents a finite difference approximation of the derivative of $l(\boldsymbol{\lambda})$. Notably, the computation of $\boldsymbol{l'_{FD}}(\boldsymbol{\lambda})$ does not require differentiation through the forward model, making it computationally tractable using standard finite difference schemes such as central differences. To understand the effect of this correction term, let us first focus on $\boldsymbol{\lambda} = \boldsymbol{\mu}$. With this Eq. (9) simplifies to

$$\nabla_{\boldsymbol{\mu}}^{\text{BBVI}}\mathcal{L} = \mathop{\mathbb{E}}_{q_\lambda(\boldsymbol{x})}\big[\nabla_{\boldsymbol{\mu}}\log q_\lambda(\boldsymbol{x})\{l(\boldsymbol{x}) - l'_{FD}(\boldsymbol{\mu})^T(\boldsymbol{x} - \boldsymbol{\mu})\}\big] + l'_{FD}(\boldsymbol{\mu}). \tag{10}$$

The detailed derivation of Eq. (10) is provided in Appendix B. Here, since the second term is independent of $\boldsymbol{x}$ it has effectively zero variance, and it approximates the gradient of the ELBO quite accurately, whereas the first term acts as

an error correction term. In practice, the expectation in Eq. (10) is approximated with MC-sampling.

$$\nabla_{\boldsymbol{\mu}}^{\text{BBVI}}\mathcal{L} \approx \frac{1}{N}[\nabla_{\boldsymbol{\mu}} \log q_\lambda(\boldsymbol{x}_i)\{l(\boldsymbol{x}_i) - l'_{FD}(\boldsymbol{\mu})^T(\boldsymbol{x}_i - \boldsymbol{\mu})\} + l'_{FD}(\boldsymbol{\mu})], \qquad \boldsymbol{x}_i \sim q_\lambda(\boldsymbol{x}). \tag{11}$$

Next, with $\boldsymbol{\lambda} = \boldsymbol{\gamma}$, Eq. (9) simplifies to

$$\nabla_{\boldsymbol{\gamma}}^{\text{BBVI}}\mathcal{L} = \mathop{\mathbb{E}}_{q_\lambda(\boldsymbol{x})}\left[\nabla_{\boldsymbol{\gamma}} \log q_\lambda(\boldsymbol{x})\{l(\boldsymbol{x}) - l'_{FD}(\boldsymbol{\gamma})^T(\boldsymbol{x} - \boldsymbol{\gamma})\}\right]. \tag{12}$$

The detailed derivation of Eq. (12) is provided in Appendix C. Here, the term $l'_{FD}(\boldsymbol{\gamma})^T(\boldsymbol{x} - \boldsymbol{\gamma})$ acts as a control variate reducing the effective variance of the BBVI gradient. Again the expectation in Eq. (12) is approximated with MC sampling as follows:

$$\nabla_{\boldsymbol{\gamma}}^{\text{BBVI}}\mathcal{L} \approx \frac{1}{N}[\nabla_{\boldsymbol{\gamma}} \log q_\lambda(\boldsymbol{x}_i)\{l(\boldsymbol{x}_i) - l'_{FD}(\boldsymbol{\gamma})^T(\boldsymbol{x}_i - \boldsymbol{\gamma})\}], \qquad \boldsymbol{x}_i \sim q_\lambda(\boldsymbol{x}). \tag{13}$$

Because of its reduced variance both Eq. (10) and Eq. (12) can approximate the true gradient with very few samples compared to the gradient estimate shown in Eq. (7). This has been shown numerically in Section 4 where our proposed gradient estimate incurs orders of magnitude less error compared to traditional BBVI gradient at all batch sizes and problem dimensions.

3.2 Deep Generative Prior

While the finite difference correction significantly enhances gradient accuracy, its computational cost scales with the dimensionality of the problem. In high-dimensional physics-based inverse problems, where the parameter space $\boldsymbol{x}$ can be substantial, this computational overhead can counterbalance the gains achieved through the gradient correction term. To overcome this challenge, we turn to recent developments of Generative Adversarial Network (GAN)-based priors [44, 56] to reduce the dimensionality of the inferred field and as a result that of the variational parameters.

GAN-priors offer a sample-based approach that re-formulates the posterior distribution $p_{\mathcal{X}}^{\text{post}}(\boldsymbol{x}|\tilde{\boldsymbol{y}})$ within a lower-dimensional latent space, denoted as $\boldsymbol{z}$, derived from a pre-trained GAN generator. Although GAN-priors have conventionally been coupled with Markov Chain Monte Carlo (MCMC) methods, they can seamlessly integrate with Variational Inference (VI) techniques. This integration is achieved by defining an appropriate variational distribution in the latent space $q_\lambda(\boldsymbol{z})$ of the generator and optimizing the corresponding variational parameters $\boldsymbol{\lambda}$ to maximize the associated Evidence Lower Bound (ELBO). This significantly reduces the complexity of the problem as the dimension of the variational parameter N_λ directly depends upon the dimension of the inferred parameter and since with GAN-prior our inferred parameter is $\boldsymbol{z}$, which is of much smaller dimension than $\boldsymbol{x}$, (i.e., $N_{\mathcal{Z}} \ll N_{\mathcal{X}}$), the overall dimension of N_λ is reduced drastically. This, in turn, helps with faster convergence and a relatively easier optimization process with improved computational and memory efficiency.

We note that a recent investigation [41] successfully incorporates GAN-priors within a VI framework in what is termed as a "white-box" setting. In contrast, our proposed methodology confronts the challenges posed by the "black-box" setting. By leveraging the strengths of GAN-priors within a VI context, we navigate the complexities of high-dimensional inference problems with computational efficiency and robustness, thereby advancing the state-of-the-art in probabilistic inference for complex systems.

For learning the prior $p_{\mathcal{X}}^{\text{prior}}(\boldsymbol{x})$ from data we use Wasserstein GAN with Gradient Penalty (WGAN-GP) [57, 58] due to its stable training property. For a WGAN-GP, the loss function $\mathcal{L}_{\text{GAN}}$ is given as

$$\mathcal{L}_{\text{GAN}}(\boldsymbol{\theta}, \boldsymbol{\phi}) = \mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_{\mathcal{X}}^{\text{prior}}}[D(\boldsymbol{x}, \boldsymbol{\phi})] - \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_{\mathcal{Z}}^{\text{prior}}}[D(\boldsymbol{G}(\boldsymbol{z}, \boldsymbol{\theta}), \boldsymbol{\phi})], \tag{14}$$

and the min-max optimization problem in Eq. (4) is solved under the constraint that $D(\boldsymbol{z}, \boldsymbol{\phi})$ lies in the space of 1-Lipschitz functions. This constraint is satisfied by enforcing a soft penalty on the gradients of the critic D with respect to $\boldsymbol{z}$ [58]. The resulting maximization problem that is solved to optimize the parameters of the discriminator is

$$\boldsymbol{\phi}^* = \arg\max_{\boldsymbol{\phi}} \mathcal{L}_{\text{GAN}}(\boldsymbol{\theta}, \boldsymbol{\phi}) - \alpha \mathop{\mathbb{E}}_{\tilde{\boldsymbol{x}}\sim \mathrm{p}_{\tilde{\boldsymbol{X}}}(\tilde{\boldsymbol{x}})}\left[(\|\nabla_{\tilde{\boldsymbol{x}}} D(\tilde{\boldsymbol{x}}, \boldsymbol{\phi})\|_2 - 1)^2\right], \tag{15}$$

where α is the gradient penalty parameter, and $\mathrm{p}_{\tilde{\boldsymbol{X}}}(\tilde{\boldsymbol{x}})$ is the uniform distribution over the straight line joining two pairs of points sampled from $p_{\mathcal{X}}^{\text{prior}}$ and the pushforward of $p_{\mathcal{Z}}^{\text{prior}}$ by $\boldsymbol{G}$ (i.e. the distribution induced by the "fake" samples generated by the generator), respectively. The loss function in Eq. (15) minimizes the Wasserstein-1 distance between $p_{\mathcal{X}}^{\text{prior}}$ and the pushforward distribution of $p_{\mathcal{Z}}^{\text{prior}}$ due to $\boldsymbol{G}$ [56].

Now, let $\boldsymbol{G}^*$ denote the generator $\boldsymbol{G}$ with optimally chosen parameters $\boldsymbol{\theta}^*$. For a perfectly trained generator $\boldsymbol{G}^*$,

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_{\mathcal{X}}^{\text{prior}}}[m(\boldsymbol{x})] = \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_{\mathcal{Z}}^{\text{prior}}}[m(\boldsymbol{G}^*(\boldsymbol{z}))] \quad \forall\, m \in C_b(\Omega_{\mathcal{X}}), \tag{16}$$

where $C_b(\cdot)$ is the space of continuously bounded functions. Eqs. (3) and (16) can be combined to compute

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim p_{\mathcal{X}}^{\text{post}}(\boldsymbol{x}|\tilde{\boldsymbol{y}})}[s(\boldsymbol{x})] = \mathop{\mathbb{E}}_{\boldsymbol{z}\sim p_{\mathcal{Z}}^{\text{post}}(\boldsymbol{z}|\tilde{\boldsymbol{y}})}[s(\boldsymbol{G}^*(\boldsymbol{z}))] \quad \forall\, \ell \in C_b(\Omega_{\mathcal{X}}), \tag{17}$$

by choosing

$$m(\boldsymbol{z}) = \frac{s(\boldsymbol{z})p_{\mathcal{Y}}^{\text{like}}(\tilde{\boldsymbol{y}}|\boldsymbol{z})}{p_{\mathcal{Y}}(\tilde{\boldsymbol{y}})}, \tag{18}$$

where

$$p_{\mathcal{Z}}^{\text{post}}(\boldsymbol{z}|\tilde{\boldsymbol{y}}) = \frac{p_{\mathcal{Y}}^{\text{like}}(\tilde{\boldsymbol{y}}|\boldsymbol{z})p_{\mathcal{Z}}^{\text{prior}}(\boldsymbol{z})}{p_{\mathcal{Y}}(\tilde{\boldsymbol{y}})} \tag{19}$$

is the posterior distribution of the latent variable $\boldsymbol{z}$.

Now in order to approximate this posterior distribution with the proposed improved BBVI scheme we directly approximate $p_{\mathcal{Z}}^{\text{post}}(\boldsymbol{z}|\tilde{\boldsymbol{y}})$ with variational distribution $q_\lambda(\boldsymbol{z})$ and maximize the ELBO as indicated in Eq. (6), albeit in $\boldsymbol{z}$-space with $\boldsymbol{x} = \boldsymbol{G}^*(\boldsymbol{z})$. Similarly, we compute the gradient of this ELBO with the gradient correction term as shown in Eq. (9) with $\boldsymbol{x} = \boldsymbol{G}^*(\boldsymbol{z})$. As mentioned before since the dimensionality of the $\boldsymbol{z}$ is much smaller than $\boldsymbol{x}$, computation of $\boldsymbol{l'_{FD}}$ is much cheaper thus alleviating the issue of high computation cost raised by the introduction of the gradient correction term.

3.3 Sampling Strategy

Our proposed finite difference corrected gradient in Eqs. (10) and (12) assumes that the variational distribution q_λ is Gaussian. Moreover, the expectation in these equations is typically approximated with a Monte Carlo sum. The inherent reason for the variance in gradient estimates is the randomness introduced during this approximation of expectation with a Monte Carlo sum. So, if we can remove/restrict this randomness then we can reduce this variance, which in turn could help us approximate the gradient with very few samples.

Since, in Eqs. (10) and (12) the variational distribution is assumed (Gaussian), we can exploit the structure of this assumed variational distribution to reduce the effective variance. Specifically, we re-parameterize samples from Gaussian using a standard normal distribution as $\boldsymbol{z} = \boldsymbol{\xi} \odot \boldsymbol{\sigma} + \boldsymbol{\mu}$ with $\boldsymbol{\xi} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I})$. The set $\{\boldsymbol{\xi}\}_{i=1}^N$ can be obtained through various methods to approximate the expectation in the gradient. Here, we explore three strategies: (i) Normal: drawing N points from a normal distribution, (ii) Symmetric Normal: sampling $N/2$ points from a normal distribution and selecting the remaining $N/2$ points by changing the sign of the first $N/2$ points, (iii) Non-uniform Deterministic: an importance sampling strategy with uniform weights but a non-uniform distribution of points. These points are distributed such that their spacing is inversely proportional to the normal or Gaussian distribution.

4 Results

In this section, we provide a series of numerical results demonstrating the effectiveness of the proposed improved BBVI method for different problems. These can be classified based on the following criteria,

1. Inverse problem: we consider the initial condition inversion problem and the permeability inversion problem in the context of hydraulic tomography.
2. Problem dimensionality: we consider problems with varying dimensionality ranging from 2 to 3200.
3. Prior distribution: we consider two different priors: Gaussian and GAN-based.
4. Likelihood distribution: we consider two different scenarios commonly encountered in practice while solving inference problems:
   (a) The likelihood distribution (Gaussian $\mathcal{N}(\boldsymbol{0}, \sigma^2\mathbb{1})$) and its parameters ($\sigma$) are known. This corresponds to the inverse heat conduction problem (Section 4.3).
   (b) The likelihood distribution is not known and must be guessed, which corresponds to a real-world experimental situation wherein the underlying noise model is completely unknown. This refers to the hydraulic tomography problem (Section 4.4) in this manuscript, where the measurement is obtained from a laboratory experiment.

Whenever possible we use a Markov Chain Monte Carlo (MCMC) algorithm as a baseline method to compare (both qualitatively and quantitatively) the relative performance of the proposed method. While HMC, and VI are efficient inference algorithms, they are not amenable to black-box forward models and hence are not considered here as a baseline. All numerical experiments were carried out on a desktop with 32 GB RAM and RTX3090 GPU.

4.1 1D Pedagogical toy problem

We present an illustrative pedagogical example to showcase the application of the proposed improved Black-Box Variational Inference (BBVI) in a one-dimensional (1D) setting, emphasizing visualization for clarity. In this example, both the prior and likelihood distributions are modeled as Gaussian distributions, while the forward model $f(x) = 0.2(x-2)^3 \sin(x-2)$ introduces non-linearity, leading to a non-convex posterior distribution. Specifically, we set the prior and likelihood distributions as $\mathcal{N}(0,1)$ and choose the observation as $\tilde{y} = 2$. To compute accurate posterior statistics, we utilize a Monte Carlo estimate with one million samples. For comparison purposes, we select baseline methods that do not require the gradient of the forward model and hence are amenable to black-box solvers. One of the baselines is the vanilla MCMC method and the other is the least-square/quadratic approximation approach. Figure 1

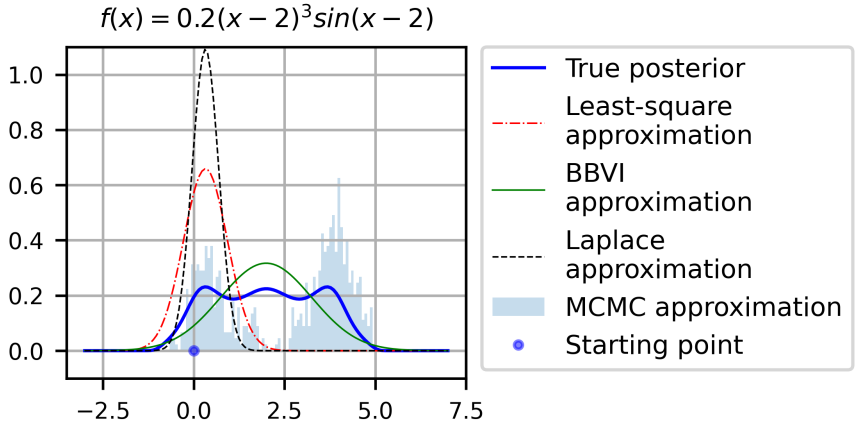


Figure 1: 1D pedagogical example: visual comparison of proposed improved BBVI with Laplace approximation, least-square approximation, and MCMC for a non-linear forward model $f(x) = 0.2(x-2)^3 \sin(x-2)$ with Gaussian prior and likelihood.

presents a qualitative comparison of BBVI with Laplace approximation, quadratic approximation, and Markov Chain Monte Carlo (MCMC) techniques for the non-linear forward model. The term "least-square approximation" refers to fitting the logarithm of the posterior using three points in a least-square sense, where the central point approximates the mean and the two side points are one standard deviation away from the central point. Note that the Laplace approximation method is not compatible with the black-box forward models as it requires computing gradient of the posterior. We include it here nonetheless since it is one of the popular methods for physics-based Bayesian inversion where the posterior is approximated with a Gaussian around the local MAP point. As can be seen from Figure 1 both Laplace and least-square approximation methods locally approximate the posterior but they are observed to be quite sensitive to the initial guess (marked with Yellow circle). In contrast, the BBVI method is robust to the initialization. Moreover, since its objective function is to minimize the KL divergence between the true posterior and the variational distribution (Gaussian in this example), it finds the Gaussian distribution which maximally covers the target density. MCMC does reasonable job covering most of the regions of the target density but it fails to capture the true density for the given fixed compute budget. The Figure 2 provides quantitative comparisons of these methods for this example. The results depicted in Figure 2 demonstrate the superior performance of BBVI over the other methods in this simple 1D toy problem with a multi-modal posterior distribution. Notably, since BBVI exhibits a broader exploration of the posterior distribution, whereas the Laplace and quadratic approximation method tends to focus on specific modes, it results in high relative error in mean and variance, as well as the Kullback-Leibler (KL) divergence between the target and approximated distributions for these local approximation methods, whereas BBVI has much smaller error and KL-divergence value. While MCMC does a reasonable job for a given compute budget, it still lags BBVI for this example. It is important to highlight that while we have utilized a Gaussian distribution as our variational family in this example, more expressive variational distributions can be chosen to closely capture the target density's characteristics, potentially yielding further improvements in quantitative metrics.

4.2 Bayesian Inversion with Conjugate Priors

In this study, we delve into a Bayesian inverse problem characterized by Gaussian prior and Gaussian likelihood distributions, leading to an analytically tractable Gaussian posterior. This controlled setting allows us to systematically analyze different aspects of the proposed algorithm. Our approach involves approximating this Gaussian posterior using a Gaussian variational distribution, $q_\lambda(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{I\sigma^2})$ with $\boldsymbol{\lambda} = [\boldsymbol{\mu}, \boldsymbol{\gamma}]$; where, $\boldsymbol{x}, \boldsymbol{\mu}, \boldsymbol{\gamma} \in N_{\mathcal{X}}$, $\boldsymbol{\gamma} = [\gamma_i =$

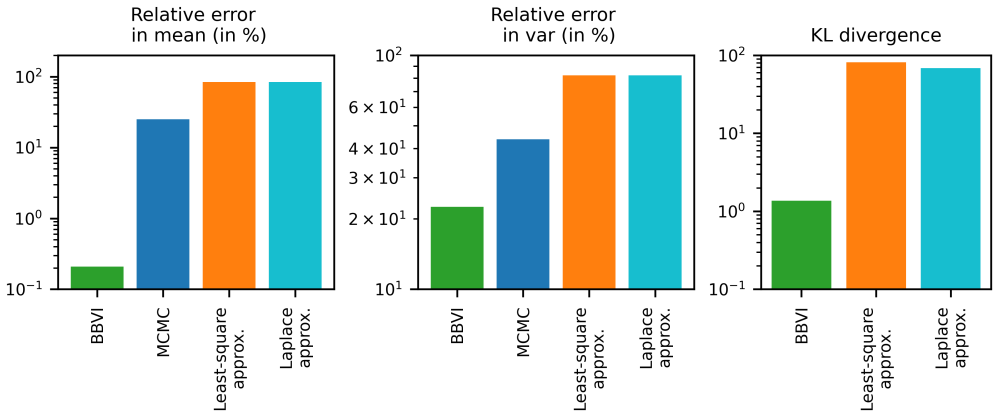


Figure 2: 1D pedagogical example: quantitative comparison of BBVI with Laplace approximation, quadratic approximation, and MCMC.

$\log(\sigma_i)]_{i=1}^{N_\mathcal{X}}$ as indicated in Section 3.1. We explore problems across various dimensionalities ($N_\mathcal{X} = 2, 10, 50$) to understand the behavior and scalability of our method. Furthermore, we analyze different sampling strategy proposed in Section 3.3. For each of these case, we compute the gradient using four different schemes: (i) vanilla BBVI gradient (Eq. (7)), (ii) BBVI gradient with finite difference gradient correction (Eq. (10)), (iii) BBVI gradient with control variate, (iv) BBVI gradient with gradient correction and control variate. We approximate the expectations in each of these gradients with Monte Carlo sum and the number of samples used in approximating this expectation (also called batch size) is treated as a hyper-parameter. We compare each of these four gradients with the true gradient (which can be derived analytically for this conjugate prior case) and compute the relative error in percentage and show how these error changes with batch size for different types of gradients in Figure 3 for different dimensionalities and sampling strategies. Each row in the figure corresponds to a different sampling scheme, while each column represents a distinct problem dimension.Different markers distinguish between different types of gradients used in our analysis. We can make following observations from Figure 3:

- Our proposed gradient (BBVI+FD+CV) method incurs up to four orders of magnitude less error compared to vanilla BBVI method for all problem dimensions and sampling strategies.
- The error in vanilla BBVI gradient goes down with the batch-size, but even with a very large batch-size (10^5) the relative error with this method is quite high across all dimensionalities and sampling strategies. As compared to this with the proposed gradient method (BBVI+FD+CV) the relative error is less than 10^{-5} for all cases even with a minimal batch size of 10. This highlights significant computational advantage of the proposed method over the traditional method.
- BBVI+FD and BBVI+CV seems to perform equal or better than the vanilla BBVI gradient in most of the cases. Further, for the most of the cases both of them seem to perform poorer compared to BBVI+FD+CV. The only exception being the symmetric normal sampling strategy, where BBVI+FD seems to perform equally well as BBVI+FD+CV.
- The proposed gradient has excellent scaling behavior as even with dimension=50 it incurs relative error of around $\mathcal{O}(10^{-5})$ for most cases even with batch size of 10.
- Symmetric Normal appears to be the best sampling strategy, as with this strategy the BBVI+FD gradient follows the exact same trend as BBVI+FD+CV. This means we can use BBVI+FD gradient without requiring additional CV term computations. For this reason, symmetric normal is our sampling strategy of choice and is used in the rest of the numerical experiments.

4.3 Physics-based inversion with synthetic data (initial condition inversion)

In our study, we direct our attention towards practical physics-based Bayesian inverse problems. Specifically, we tackle the challenging task of inferring the initial condition in the time-dependent heat conduction equation, omitting a source term, and employing a constant conductivity field with a diffusivity of $\kappa = 0.064$ units. We next consider the problem of inferring the initial condition for the transient heat conduction problem given the noisy temperature measurement at

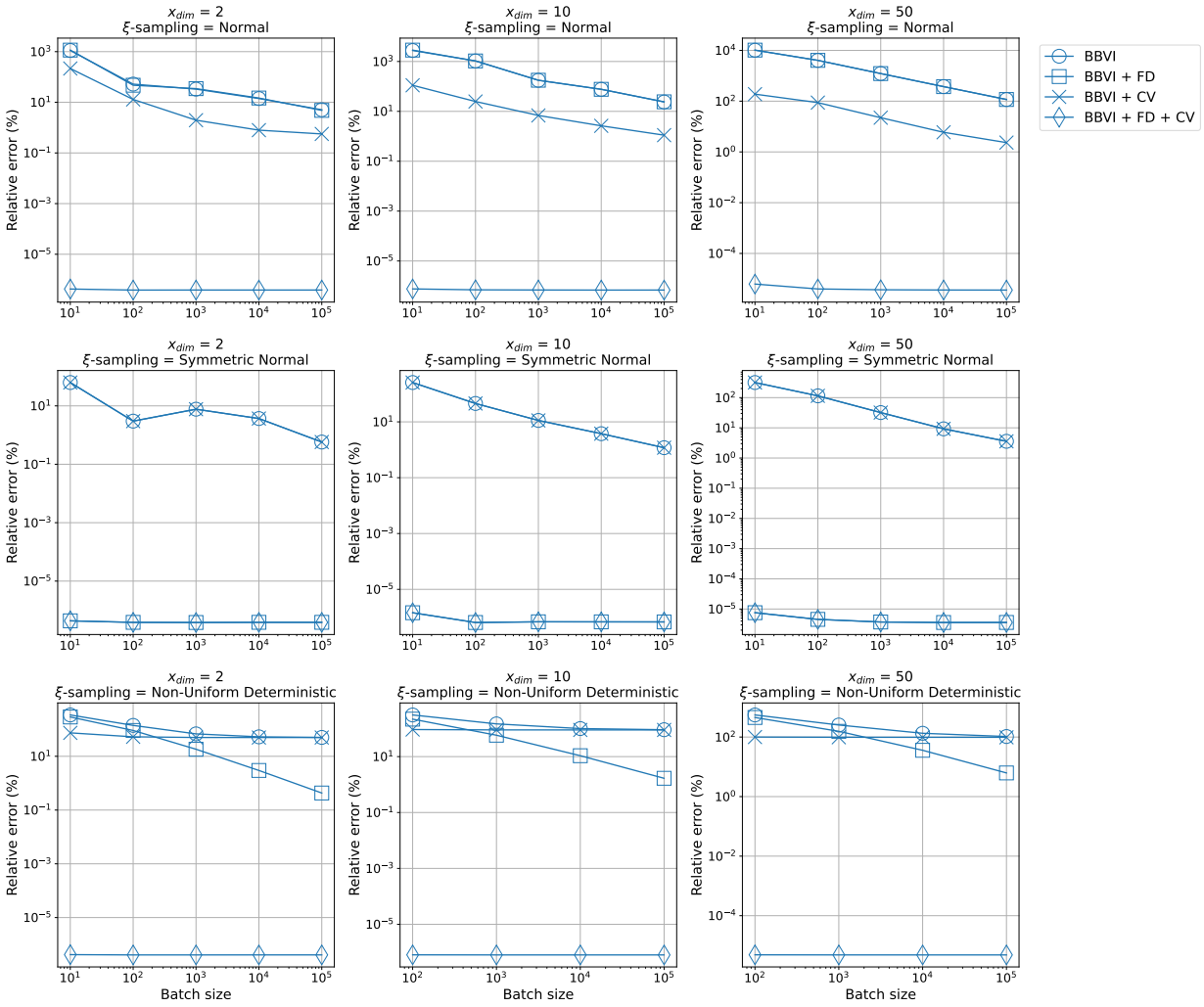


Figure 3: Relative error in gradient computation as a function of batch size (N). Each row represents a different sampling scheme, and each column represents the problem dimension. Different colors in each subplot indicate different evaluation points for the gradient, and various markers represent different gradient estimates. Circle mark represents vanilla BBVI gradient (Eq. (8)); cross mark indicates BBVI gradient with control variate; square represents BBVI gradient with finite difference (FD) gradient correction (Eq. (11)); diamond denotes BBVI gradient with FD gradient correction and control variate.

some later time.

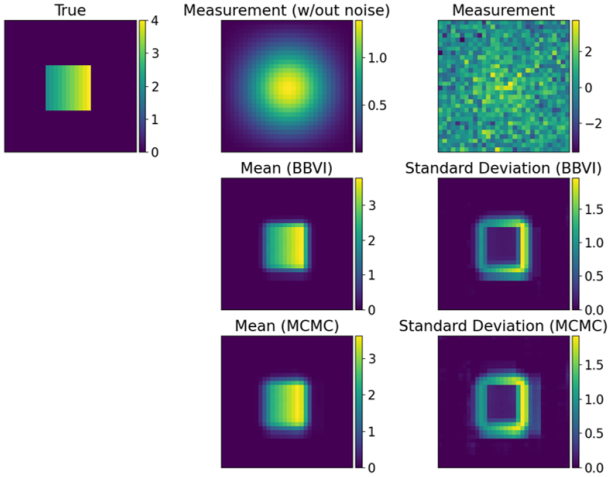
$$\nabla \cdot (\alpha \nabla u(\boldsymbol{s}, t)) = \partial u(\boldsymbol{s}, t)/\partial t \qquad \boldsymbol{s} \in \Omega, t \in (0, T] \tag{20}$$

$$u(\boldsymbol{s}, 0) = m(\boldsymbol{s}) \qquad \boldsymbol{s} \in \Omega \tag{21}$$

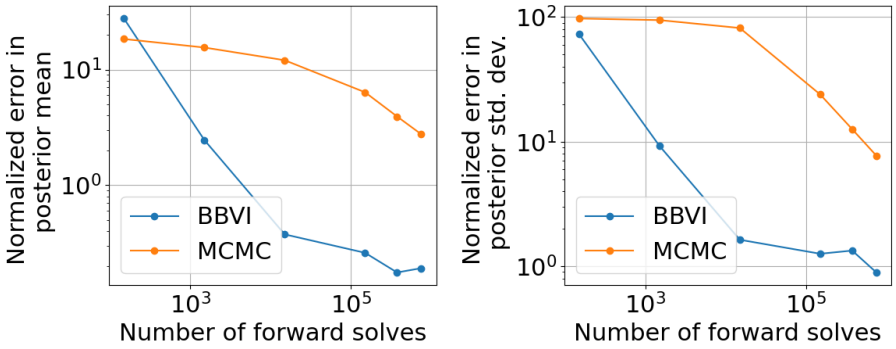
$$u(\boldsymbol{s}, t) = g(\boldsymbol{s}, t) \qquad \boldsymbol{s} \in \partial\Omega, t \in (0, T] \tag{22}$$

where $\Omega \subset \mathbb{R}^2$ is a square domain with length = 2π units and α is thermal diffusivity. $u(\boldsymbol{s}, t)$ is temperature at location $\boldsymbol{s}$ at time t, and $m(\boldsymbol{s})$ is the initial condition for temperature. Here, the parameter to infer $\boldsymbol{x}$ is the nodal values of initial condition $m(\boldsymbol{s})$ and the measurement $\boldsymbol{y}$ is the nodal values of final temperature $u(\boldsymbol{s}, T)$.

The corresponding inverse problem is given noisy (and possibly sparse) temperature-field measurements $\tilde{\boldsymbol{y}}$ at some later time $t = T$ infer the posterior distribution corresponding to the initial condition of temperature $\boldsymbol{x}$. This is an ill-posed problem as significant information is lost via diffusion as we move forward in time. Here, the initial condition



(a)



(b)

Figure 4: (a) Initial condition inversion. *Top row*: true inferred field $\boldsymbol{x}$, final temperature $\boldsymbol{y} = \boldsymbol{f}(\boldsymbol{x})$, measured temperature field $\tilde{\boldsymbol{y}} = \boldsymbol{y} + \boldsymbol{\eta}$. *Second and third row*: posterior mean and standard deviation obtained using BBVI and MCMC, respectively. (b) Normalized error in posterior statistics as a function of the number of forward solves for BBVI and MCMC.

is prescribed as zero everywhere inside the domain except in a rectangle region, where it varies linearly from left (with the value of 2 units) to right (with the value of 4 units), as can be seen from Figure 4(a). Given a noisy temperature field at $t = 1$, characterized by a noise variance of 1, our goal is to recover the initial condition. This problem presents a severe ill-posed nature due to information loss during the diffusion process and the presence of substantial measurement noise. We represent the discretized initial and final temperature fields as $\boldsymbol{x}$ and $\boldsymbol{y}$, respectively, on a 32×32 Cartesian grid. Our approach leverages a GAN-prior, trained on a parametric rectangular dataset utilized in prior studies [45, 59],

within the 5-dimensional latent space of the GAN. For optimization, we employ the Adam optimizer [60] in conjunction with the Normal Symmetric sampling scheme for BBVI. As a baseline, we compare our method against the random walk Markov Chain Monte Carlo (MCMC) approach, often used with black-box simulators in physics-based inverse problems. We optimally tune the proposal variance for MCMC.

Qualitative results, shown in Figure 4(a), indicate slightly superior reconstruction by BBVI. Figure 4(b) presents a quantitative analysis. We solve the inverse problem with a fixed number of forward solves for both methods and assess the error in posterior statistics. Notably, BBVI consistently demonstrates smaller errors and significantly faster convergence rates compared to MCMC, underscoring its superiority in addressing this challenging problem.

4.4 Physics-based inversion with experimental data (hydraulic tomography)

In this section, we demonstrate the efficacy of our proposed algorithm using a real-world experimental dataset. Our focus is on a laboratory-scale hydraulic tomography experiment where pressure/hydraulic head changes are recorded during a series of pumping tests to reconstruct the spatially distributed hydraulic conductivity field of a lab-scale sandbox. This is a critical yet challenging inverse problem in geophysics and petroleum engineering. The experiments were conducted at the University of Iowa under the supervision of Walter Illman and colleagues, and the dataset has been utilized in prior studies [61, 62]. Detailed information regarding the sandbox flow-through tests can be found in [61].

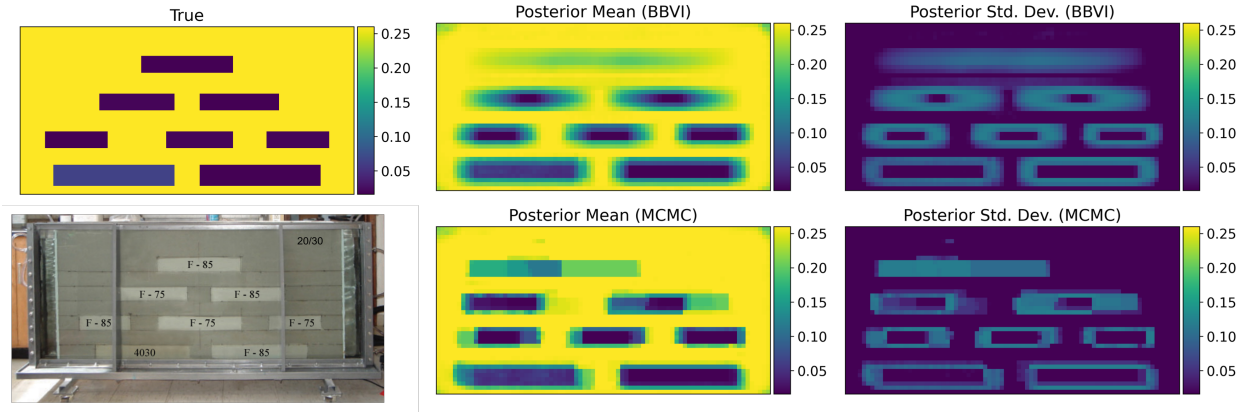


Figure 5: Hydraulic tomography using experimental data. *First column*: (top) true permeability field, (bottom) photograph of the sandbox used during the experiment. Figure reused from [62]. The details of the sandbox flow-through tests can be found in [61]. *Second column*: posterior mean obtained using BBVI and MCMC method. *Third column*: posterior standard deviation obtained using BBVI and MCMC method.

The picture in the bottom left panel of Figure 5 illustrates the front view of the sandbox utilized in this experiment. Four distinct commercially available sands (labeled as F-85, F-75, 4030, and 20/30 in Figure 5 (left panel)) were employed in constructing the sandbox. The box comprises finer sand, with the background composed of coarser sand. Notably, eight rectangular slots (identified as F-85, F-75, 4030) were filled with low-permeability sand, while the background (20/30) was filled with higher-permeability sand. The true permeability values at various locations are depicted in Figure 5 (top left panel). On the rear side of the sandbox, 48 pressure sensors were installed to measure hydraulic head. Nine distinct experiments were conducted for hydraulic surveying, resulting in a total of 9 x 47 = 423 measurements. Data from all nine experiments were utilized for inference purposes.

The governing equation for the forward model for this hydraulic survey is represented by:

$$\begin{aligned} -\nabla \cdot (\kappa(\boldsymbol{s})\nabla u(\boldsymbol{s})) &= f_i \delta(\boldsymbol{s}-\boldsymbol{a}), && \boldsymbol{s}=(s_1,s_2)\in\Omega \\ u(\boldsymbol{s}) &= g_i, && \boldsymbol{s}=(s_1,s_2)\in\partial\Omega_g \\ \kappa(\boldsymbol{s})\nabla u(\boldsymbol{s}) &= h_i, && \boldsymbol{s}=(s_1,s_2)\in\partial\Omega_h \end{aligned} \tag{23}$$

Here, κ, u, and $\boldsymbol{a}$ represent hydraulic conductivity, hydraulic head, and the locations of circled ports, respectively. f_i, g_i, and h_i denote the source, Dirichlet boundary condition, and flux boundary condition for the i^{th} experiment ($i = 1, 2, \ldots, 9$).

For the inference process, we employ GAN-based priors. This prior was trained by varying the horizontal and vertical locations of the eight sand blocks. A CNN-based surrogate model serves as the forward model, mapping the hydraulic

conductivity image to 432 hydraulic head measurements corresponding to 47 head measurements for 9 experiments. Additionally, we utilize MCMC as a baseline method for comparison. We conduct both MCMC and improved BBVI under two different scenarios: (i) fixed number of forward model evaluations and (ii) fixed CPU wall-clock time. Again, we use MCMC method as a baseline. Table 1 shows the quantitative results for both these cases. Further, second and

Table 1: Quantitative comparison of MCMC and BBVI

For a fixed number of forward solves

QoI	MCMC	**BBVI**
Error ↓	31.58%	**26.06%**
Coverage (95% CI) ↑	0.760	**0.903**
Time (in seconds) ↓	108.25	**11.07**

For fixed CPU wall-clock time

QoI	MCMC	**BBVI**
Error ↓	39.81%	**26.09%**
Coverage (95% CI) ↑	0.274	**0.900**
No. of forward solves ↑	3000	**72000**

third column of Figure 5 shows the qualitative comparison of the posterior mean and standard deviation obtained using the proposed BBVI method and the baseline MCMC method for the fixed number of forward model evaluation case. As can be seen from the figure, BBVI does much better job at faithfully reconstructing the permeability field whereas MCMC faces hard time reconstructing some of the rectangular blocks. The same is reflected in Table 1 (for a fixed number of forward solves case), where BBVI consistently outperforms MCMC in relative error in posterior mean as well as coverage while taking significantly less time due to parallel processing of batch as opposed to sequential nature of MCMC. Table 1 also indicates that for the case when we compare BBVI and MCMC for the fixed CPU wall-clock time, BBVI still outperforms MCMC in all metrics. This is because, for a given time BBVI can allow multiple forward evaluation due to its parallel (batch-wise) processing, whereas MCMC can accommodate very limited number of forward model evaluations resulting in poor overall performance.

5 Conclusion

n conclusion, our study presents a modular framework integrating black-box variational inference (BBVI) with deep generative priors to effectively address high-dimensional Bayesian inversion challenges in black-box forward model simulators. By introducing a novel gradient correction term and tailored sampling strategy for BBVI, we mitigate gradient errors across dimensions, ensuring scalability and efficiency even with limited batch sizes. Leveraging Generative Adversarial Network (GAN)-based priors enables the solution of complex high-dimensional inverse problems, surpassing traditional Markov Chain Monte Carlo (MCMC) methods in accuracy and convergence speed. Our validation on diverse datasets underscores the practical significance of our approach, offering a promising avenue for advancing Bayesian inference capabilities across scientific and engineering disciplines.

While our proposed method demonstrates significant advancements in high-dimensional Bayesian inversion, it is important to acknowledge certain limitations that pave the way for exciting future research directions. Firstly, our current approach relies on a multivariate Gaussian variational distribution, which may not be adequate for certain applications where the posterior distribution is complex and multi-modal. Future work will explore more expressive and complex variational distributions to address this challenge effectively. Additionally, extending our method to tackle more exotic inverse problems involving real-world data, such as those in geophysical or environmental sciences, presents an exciting avenue for further exploration and application of our framework. These efforts aim to enhance the versatility and applicability of our methodology across a wide range of scientific and engineering fields with the overarching goal of enabling the seamless integration of powerful black-box simulators for conducting efficient high-dimensional Bayesian inference.

References

[1] Anna Kauffeldt, Fredrik Wetterhall, Florian Pappenberger, Peter Salamon, and Jutta Thielen. Technical review of large-scale hydrological models for implementation in operational flood forecasting schemes on continental level. *Environmental Modelling & Software*, 75:68–76, 2016.

[2] Valentina Krysanova, Tobias Vetter, Stephanie Eisner, Shaochun Huang, Ilias Pechlivanidis, Michael Strauch, Alexander Gelfan, Rohini Kumar, Valentin Aich, Berit Arheimer, et al. Intercomparison of regional-scale hydrological models and climate change impacts projected for 12 large river basins worldwide—a synthesis. *Environmental Research Letters*, 12(10):105002, 2017.

[3] Roland Barthel and Stefan Banzhaf. Groundwater and surface water interaction at the regional-scale–a review with focus on regional integrated models. *Water resources management*, 30(1):1–32, 2016.

[4] John S McCartney, Marcelo Sánchez, and Ingrid Tomac. Energy geotechnics: Advances in subsurface energy recovery, storage, exchange, and waste management. *Computers and Geotechnics*, 75:244–256, 2016.

[5] A. P. Thompson, H. M. Aktulga, R. Berger, D. S. Bolintineanu, W. M. Brown, P. S. Crozier, P. J. in 't Veld, A. Kohlmeyer, S. G. Moore, T. D. Nguyen, R. Shan, M. J. Stevens, J. Tranchida, C. Trott, and S. J. Plimpton. LAMMPS - a flexible simulation tool for particle-based materials modeling at the atomic, meso, and continuum scales. *Comp. Phys. Comm.*, 271:108171, 2022.

[6] Richard J Gowers, Max Linke, Jonathan Barnoud, Tyler John Edward Reddy, Manuel N Melo, Sean L Seyler, Jan Domanski, David L Dotson, Sébastien Buchoux, Ian M Kenney, et al. Mdanalysis: a python package for the rapid analysis of molecular dynamics simulations. Technical report, Los Alamos National lab.(LANL), Los Alamos, NM (United States), 2019.

[7] Pierre Horgue, Cyprien Soulaine, Jacques Franc, Romain Guibert, and Gérald Debenest. An open-source toolbox for multiphase flow in porous media. *Computer Physics Communications*, 187:217–226, 2015.

[8] Glenn E Hammond, Peter C Lichtner, and RT Mills. Evaluating the performance of parallel subsurface simulators: An illustrative example with pflotran. *Water resources research*, 50(1):208–228, 2014.

[9] Yoojin Jung, George Shu Heng Pau, Stefan Finsterle, and Ryan M Pollyea. Tough3: A new efficient version of the tough suite of multiphase flow and transport simulators. *Computers & Geosciences*, 108:2–7, 2017.

[10] Joseph J Hamman, Bart Nijssen, Theodore J Bohn, Diana R Gergel, and Yixin Mao. The variable infiltration capacity model version 5 (vic-5): Infrastructure improvements for new applications and reproducibility. *Geoscientific Model Development*, 11(8):3481–3496, 2018.

[11] Thomas L Delworth, Anthony J Broccoli, Anthony Rosati, Ronald J Stouffer, V Balaji, John A Beesley, William F Cooke, Keith W Dixon, John Dunne, KA Dunne, et al. Gfdl's cm2 global coupled climate models. part i: Formulation and simulation characteristics. *Journal of Climate*, 19(5):643–674, 2006.

[12] Ronald J Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8:229–256, 1992.

[13] René P Schwarzenbach, Thomas Egli, Thomas B Hofstetter, Urs Von Gunten, and Bernhard Wehrli. Global water pollution and human health. *Annual review of environment and resources*, 35:109–136, 2010.

[14] Cameron W. Smith, Eroma Abeysinghe, Suresh Marru, and Kenneth E. Jansen. Phasta science gateway for high performance computational fluid dynamics. In *Proceedings of the Practice and Experience on Advanced Research Computing*, PEARC '18, New York, NY, USA, 2018. Association for Computing Machinery.

[15] Reed M Maxwell. A terrain-following grid transform and preconditioner for parallel, large-scale, integrated hydrologic modeling. *Advances in Water Resources*, 53:109–117, 2013.

[16] David E Keyes, Lois C McInnes, Carol Woodward, William Gropp, Eric Myra, Michael Pernice, John Bell, Jed Brown, Alain Clo, Jeffrey Connors, Emil Constantinescu, Don Estep, Kate Evans, Charbel Farhat, Ammar Hakim, Glenn Hammond, Glen Hansen, Judith Hill, Tobin Isaac, Xiangmin Jiao, Kirk Jordan, Dinesh Kaushik, Efthimios Kaxiras, Alice Koniges, Kihwan Lee, Aaron Lott, Qiming Lu, John Magerlein, Reed Maxwell, Michael McCourt, Miriam Mehl, Roger Pawlowski, Amanda P Randles, Daniel Reynolds, Beatrice Rivière, Ulrich Rüde, Tim Scheibe, John Shadid, Brendan Sheehan, Mark Shephard, Andrew Siegel, Barry Smith, Xianzhu Tang, Cian Wilson, and Barbara Wohlmuth. Multiphysics simulations: Challenges and opportunities. *The International Journal of High Performance Computing Applications*, 27(1):4–83, 2013.

[17] Gang Lei, Jianguo Zhu, Youguang Guo, Chengcheng Liu, and Bo Ma. A review of design optimization methods for electrical machines. *Energies*, 10(12):1962, 2017.

[18] Kyle Cranmer, Johann Brehmer, and Gilles Louppe. The frontier of simulation-based inference. *Proceedings of the National Academy of Sciences*, 117(48):30055–30062, 2020.

[19] Mohammad Nabi Omidvar, Xiaodong Li, and Xin Yao. A review of population-based metaheuristics for large-scale black-box global optimization—part i. *IEEE Transactions on Evolutionary Computation*, 26(5):802–822, 2021.

[20] Xiaomeng Song, Jianyun Zhang, Chesheng Zhan, Yunqing Xuan, Ming Ye, and Chonggang Xu. Global sensitivity analysis in hydrological modeling: Review of concepts, methods, theoretical framework, and applications. *Journal of hydrology*, 523:739–757, 2015.

[21] Niklas Linde, David Ginsbourger, James Irving, Fabio Nobile, and Arnaud Doucet. On uncertainty quantification in hydrogeology and hydrogeophysics. *Advances in Water Resources*, 110:166–181, 2017.

[22] Peter K Kitanidis. Quasi-linear geostatistical theory for inversing. *Water resources research*, 31(10):2411–2419, 1995.

[23] Efi Foufoula-Georgiou and Peter K Kitanidis. Gradient dynamic programming for stochastic optimal control of multidimensional water resources systems. *Water resources research*, 24(8):1345–1359, 1988.

[24] Hojat Ghorbanidehno, Amalia Kokkinaki, Jonghyun Lee, and Eric Darve. Recent developments in fast and scalable inverse modeling and data assimilation methods in hydrology. *Journal of Hydrology*, 591:125266, 2020.

[25] Arlen W Harbaugh. *MODFLOW-2005, the US Geological Survey modular ground-water model: the ground-water flow process*, volume 6. US Department of the Interior, US Geological Survey Reston, VA, USA, 2005.

[26] Jonghyun Lee and Peter K Kitanidis. Large-scale hydraulic tomography and joint inversion of head and tracer data using the principal component geostatistical approach (pcga). *Water Resources Research*, 50(7):5410–5427, 2014.

[27] Tan Bui-Thanh, Omar Ghattas, James Martin, and Georg Stadler. A computational framework for infinite-dimensional bayesian inverse problems part i: The linearized case, with application to global seismic inversion. *SIAM Journal on Scientific Computing*, 35(6):A2494–A2523, 2013.

[28] Mingjie Chen, Azizallah Izady, Osman A Abdalla, and Mansoor Amerjeed. A surrogate-based sensitivity quantification and bayesian inversion of a regional groundwater flow model. *Journal of Hydrology*, 557:826–837, 2018.

[29] Yefang Jiang, Allan D Woodbury, and Scott Painter. Full-bayesian inversion of the edwards aquifer. *Groundwater*, 42(5):724–733, 2004.

[30] Ji-Chun Wu, Le Lu, and Tian Tang. Bayesian analysis for uncertainty and risk in a groundwater numerical model's predictions. *Human and Ecological Risk Assessment: An International Journal*, 17(6):1310–1331, 2011.

[31] Jiangjiang Zhang, Weixuan Li, Lingzao Zeng, and Laosheng Wu. An adaptive gaussian process-based method for efficient bayesian experimental design in groundwater contaminant source identification problems. *Water Resources Research*, 52(8):5971–5984, 2016.

[32] P. K. Kitanidis. Bayesian and geostatistical approaches to inverse problems. In *Large-Scale Inverse Problems and Quantification of Uncertainty*, pages 71–85. John Wiley & Sons, Ltd, 2010.

[33] Simon Duane, A.D. Kennedy, Brian J. Pendleton, and Duncan Roweth. Hybrid monte carlo. *Physics Letters B*, 195(2):216–222, 1987.

[34] Andrew Gelman, John B B Carlin, Hal S S Stern, and Donald B B Rubin. Bayesian Data Analysis, Third Edition (Texts in Statistical Science). *Book*, page 675, 2014.

[35] Gareth O Roberts and Jeffrey S Rosenthal. Optimal scaling of discrete approximations to langevin diffusions. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 60(1):255–268, 1998.

[36] David M. Blei, Alp Kucukelbir, and Jon D. McAuliffe. Variational inference: A review for statisticians. *Journal of the American Statistical Association*, 112:859 – 877, 2016.

[37] Tiangang Cui, James Martin, Youssef M Marzouk, Antti Solonen, and Alessio Spantini. Likelihood-informed dimension reduction for nonlinear inverse problems. *Inverse Problems*, 30(11):114015, 2014.

[38] Michael Betancourt. A conceptual introduction to Hamiltonian Monte Carlo. *arXiv preprint arXiv:1701.02434*, 2017.

[39] Tarek A El Moselhy and Youssef M Marzouk. Bayesian inference with optimal maps. *Journal of Computational Physics*, 231(23):7815–7850, 2012.

[40] Anna Andrle, Nando Farchmin, Paul Hagemann, Sebastian Heidenreich, Victor Soltwisch, and Gabriele Steidl. Invertible neural networks versus MCMC for posterior reconstruction in grazing incidence X-ray fluorescence. In *International Conference on Scale Space and Variational Methods in Computer Vision*, pages 528–539. Springer, 2021.

[41] Agnimitra Dasgupta, Dhruv V Patel, Deep Ray, Erik A Johnson, and Assad A Oberai. A dimension-reduced variational approach for solving physics-based inverse problems using generative adversarial network priors and normalizing flows. *arXiv preprint arXiv:2310.04690*, 2023.

[42] Rajesh Ranganath, Sean Gerrish, and David Blei. Black Box Variational Inference. In Samuel Kaski and Jukka Corander, editors, *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics*, volume 33 of *Proceedings of Machine Learning Research*, pages 814–822, Reykjavik, Iceland, 22–25 Apr 2014. PMLR.

[43] Shakir Mohamed, Mihaela Rosca, Michael Figurnov, and Andriy Mnih. Monte carlo gradient estimation in machine learning. *The Journal of Machine Learning Research*, 21(1):5183–5244, 2020.

[44] Eric Laloy, Romain Hérault, Diederik Jacques, and Niklas Linde. Training-image based geostatistical inversion using a spatial generative adversarial neural network. *Water Resources Research*, 54(1):381–406, Jan 2018.

[45] Dhruv V Patel, Deep Ray, and Assad A Oberai. Solution of physics-based bayesian inverse problems with deep generative priors. *Computer Methods in Applied Mechanics and Engineering*, 400:115428, 2022.

[46] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[47] Yongjun Hong, Uiwon Hwang, Jaeyoon Yoo, and Sungroh Yoon. How generative adversarial networks and their variants work: An overview. *ACM Computing Surveys*, 52(1):1–43, 2019.

[48] Xin Yi, Ekta Walia, and Paul Babyn. Generative adversarial network in medical imaging: A review. *Medical Image Analysis*, 58:101552, 2019.

[49] Abdul Jabbar, Xi Li, and Bourahla Omar. A survey on generative adversarial networks: Variants, applications, and training. *ACM Computing Surveys*, 54(8):1–49, 2021.

[50] Akash Kumar Dhaka, Alejandro Catalina, Manushi Welandawe, Michael R Andersen, Jonathan Huggins, and Aki Vehtari. Challenges and opportunities in high dimensional variational inference. *Advances in Neural Information Processing Systems*, 34:7787–7798, 2021.

[51] Zhenming Shun and Peter McCullagh. Laplace approximation of high dimensional integrals. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 57(4):749–760, 1995.

[52] David JC MacKay. Choice of basis for laplace approximation. *Machine learning*, 33:77–86, 1998.

[53] Peter W Glynn. Likelihood ratio gradient estimation for stochastic systems. *Communications of the ACM*, 33(10):75–84, 1990.

[54] Francesco Locatello, Gideon Dresdner, Rajiv Khanna, Isabel Valera, and Gunnar Rätsch. Boosting black box variational inference. *Advances in Neural Information Processing Systems*, 31, 2018.

[55] Rajesh Ranganath, Dustin Tran, and David Blei. Hierarchical variational models. In *International conference on machine learning*, pages 324–333. PMLR, 2016.

[56] Dhruv V. Patel and Assad A. Oberai. Gan-based priors for quantifying uncertainty in supervised learning. *SIAM/ASA Journal on Uncertainty Quantification*, 9(3):1314–1343, 2021.

[57] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *International Conference on Machine Learning*, pages 214–223. PMLR, 2017.

[58] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *Advances in neural information processing systems*, pages 5767–5777, 2017.

[59] Dhruv V Patel, Deep Ray, Harisankar Ramaswamy, and Assad Oberai. Bayesian inference in physics-driven problems with adversarial priors. In *NeurIPS 2020 Workshop on Deep Learning and Inverse Problems*, 2020.

[60] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[61] Walter A Illman, Xiaoyi Liu, and Andrew Craig. Steady-state hydraulic tomography in a laboratory aquifer with deterministic heterogeneity: Multi-method and multiscale validation of hydraulic conductivity tomograms. *Journal of Hydrology*, 341(3-4):222–234, 2007.

[62] X. Liu and P. K. Kitanidis. Large-scale inverse modeling with an application in hydraulic tomography. *Water Resources Research*, 47(2):2501, feb 2011.

Appendix

A Derivation of Eq. (7)

In this section we provide the derivation of the BBVI gradient first proposed in [42] for completeness.

We know that the ELBO for variational inference is given by (Eq. (6))

$$\mathcal{L}(\boldsymbol{\lambda}) = \text{ELBO} = \mathop{\mathbb{E}}_{\boldsymbol{x}\sim q_\lambda(\boldsymbol{x})} \left[\log p(\boldsymbol{x}, \tilde{\boldsymbol{y}}) - \log q_\lambda(\boldsymbol{x})\right]. \tag{24}$$

In variational inference we are interested in finding the optimal values of $\boldsymbol{\lambda}^*$ such that it maximizes ELBO, i.e., $\boldsymbol{\lambda}^* = \arg\max_{\boldsymbol{\lambda}} \mathcal{L}(\boldsymbol{\lambda})$. In practice, we do this by using gradient-based optimizers and miximize ELBO using gradient

ascent. This requires computing $(\nabla_{\boldsymbol{\lambda}}\mathcal{L}$. This can be simplified as follows:

$$\begin{aligned}
\nabla_{\boldsymbol{\lambda}}^{\text{BBVI}}\mathcal{L} &= \nabla_{\boldsymbol{\lambda}} \mathop{\mathbb{E}}_{\boldsymbol{x}\sim q_\lambda(\boldsymbol{x})}[\log p(\boldsymbol{x},\tilde{\boldsymbol{y}}) - \log q_\lambda(\boldsymbol{x})] \\
&= \nabla_{\boldsymbol{\lambda}} \int_{\Omega_x} \{\log p(\boldsymbol{x},\tilde{\boldsymbol{y}}) - \log q_\lambda(\boldsymbol{x})\} q_\lambda(\boldsymbol{x}) d\boldsymbol{x} \\
&= \int_{\Omega_x} \nabla_{\boldsymbol{\lambda}} [\{\log p(\boldsymbol{x},\tilde{\boldsymbol{y}}) - \log q_\lambda(\boldsymbol{x})\} q_\lambda(\boldsymbol{x})] d\boldsymbol{x} \\
&= \underbrace{\int_{\Omega_x} \nabla_{\boldsymbol{\lambda}} [\log p(\boldsymbol{x},\tilde{\boldsymbol{y}}) - \log q_\lambda(\boldsymbol{x})] q_\lambda(\boldsymbol{x}) d\boldsymbol{x}}_{A} + \underbrace{\int_{\Omega_x} \nabla_{\boldsymbol{\lambda}} q_\lambda(\boldsymbol{x}) [\log p(\boldsymbol{x},\tilde{\boldsymbol{y}}) - \log q_\lambda(\boldsymbol{x})] d\boldsymbol{x}}_{B}. \qquad (25)
\end{aligned}$$

Next, let's look at each term separately.

$$\begin{aligned}
A &= \int_{\Omega_x} \nabla_{\boldsymbol{\lambda}} [\log p(\boldsymbol{x},\tilde{\boldsymbol{y}}) - \log q_\lambda(\boldsymbol{x})] q_\lambda(\boldsymbol{x}) d\boldsymbol{x} \\
&= -\int_{\Omega_x} \nabla_{\boldsymbol{\lambda}} \log q_\lambda(\boldsymbol{x}) q_\lambda(\boldsymbol{x}) d\boldsymbol{x} \\
&= -\int_{\Omega_x} \nabla_{\boldsymbol{\lambda}} q_\lambda(\boldsymbol{x}) d\boldsymbol{x} \\
&= -\nabla_{\boldsymbol{\lambda}} \int_{\Omega_x} q_\lambda(\boldsymbol{x}) \boldsymbol{x} \\
&= -\nabla_{\boldsymbol{\lambda}} 1 \\
&= 0. \qquad (26)
\end{aligned}$$

$$\begin{aligned}
B &= \int_{\Omega_x} \nabla_{\boldsymbol{\lambda}} q_\lambda(\boldsymbol{x}) [\log p(\boldsymbol{x},\tilde{\boldsymbol{y}}) - \log q_\lambda(\boldsymbol{x})] d\boldsymbol{x} \\
&= \int_{\Omega_x} \frac{\nabla_{\boldsymbol{\lambda}} q_\lambda(\boldsymbol{x})}{q_\lambda(\boldsymbol{x})} q_\lambda(\boldsymbol{x}) [\log p(\boldsymbol{x},\tilde{\boldsymbol{y}}) - \log q_\lambda(\boldsymbol{x})] d\boldsymbol{x} \\
&= \int_{\Omega_x} \nabla_{\boldsymbol{\lambda}} \log q_\lambda(\boldsymbol{x}) q_\lambda(\boldsymbol{x}) \log p(\boldsymbol{x},\tilde{\boldsymbol{y}}) d\boldsymbol{x} - \int_{\Omega_x} \nabla_{\boldsymbol{\lambda}} \log q_\lambda(\boldsymbol{x}) q_\lambda(\boldsymbol{x}) \log q_\lambda(\boldsymbol{x}) d\boldsymbol{x} \\
&= \int_{\Omega_x} \nabla_{\boldsymbol{\lambda}} \log q_\lambda(\boldsymbol{x}) q_\lambda(\boldsymbol{x}) [\log p_\eta(\tilde{\boldsymbol{y}}|\boldsymbol{x}) + \log p(\boldsymbol{x})] d\boldsymbol{x} - \int_{\Omega_x} \nabla_{\boldsymbol{\lambda}} \log q_\lambda(\boldsymbol{x}) q_\lambda(\boldsymbol{x}) \log q_\lambda(\boldsymbol{x}) d\boldsymbol{x} \\
&= \int_{\Omega_x} \nabla_{\boldsymbol{\lambda}} \log q_\lambda(\boldsymbol{x}) q_\lambda(\boldsymbol{x}) [\log p_\eta(\tilde{\boldsymbol{y}}|\boldsymbol{x}) + \log p(\boldsymbol{x}) - \log q_\lambda(\boldsymbol{x})] d\boldsymbol{x} \qquad (27)
\end{aligned}$$

By substituting Eqs. (26) and (27) in Eq. (25)

$$\begin{aligned}
\nabla_{\boldsymbol{\lambda}}^{\text{BBVI}}\mathcal{L} &= \int_{\Omega_x} \nabla_{\boldsymbol{\lambda}} \log q_\lambda(\boldsymbol{x}) q_\lambda(\boldsymbol{x}) [\log p_\eta(\tilde{\boldsymbol{y}}|\boldsymbol{x}) + \log p(\boldsymbol{x}) - \log q_\lambda(\boldsymbol{x})] d\boldsymbol{x} \\
&= \mathop{\mathbb{E}}_{q_\lambda(\boldsymbol{x})} \Big[\nabla_{\boldsymbol{\lambda}} \log q_\lambda(\boldsymbol{x}) [\log p_{\mathcal{Y}}^{\text{like}}(\tilde{\boldsymbol{y}} - \boldsymbol{f}(\boldsymbol{x})) + \log p_{\mathcal{X}}^{\text{prior}}(\boldsymbol{x}) - \log q_\lambda(\boldsymbol{x})]\Big] \qquad (28)
\end{aligned}$$

which is the same equation as Eq. (7)

B Derivation of Eq. (10)

We have $q_\lambda(\boldsymbol{x}) = \mathcal{N}(\boldsymbol{x}; \boldsymbol{\mu}, \Sigma)$ with Σ being a diagonal covariance matrix with the diagonal vector being $\boldsymbol{\sigma^2} = [\sigma_i^2]_{i=1}^{N_{\mathcal{X}}}$ and $\boldsymbol{\gamma} = [\gamma_i = \log(\sigma_i)]_{i=1}^{N_{\mathcal{X}}}$. Now,

$$\begin{aligned}
\log q_\lambda(\boldsymbol{x}) &= -\frac{N_{\mathcal{X}}}{2}\log(2\pi) - \sum_{i=1}^{N_{\mathcal{X}}} \log \sigma_i - \frac{1}{2}\sum_{i=1}^{N_{\mathcal{X}}} \left(\frac{x_i - \mu_i}{\sigma_i}\right)^2 \\
&= -\frac{N_{\mathcal{X}}}{2}\log(2\pi) - \sum_{i=1}^{N_{\mathcal{X}}} \gamma_i - \frac{1}{2}\sum_{i=1}^{N_{\mathcal{X}}} \left(\frac{x_i - \mu_i}{e^{\gamma_i}}\right)^2 \qquad (29)
\end{aligned}$$

$$\implies \nabla_{\mu_k} \log q_\lambda(\boldsymbol{x}) = \frac{x_k - \mu_k}{\sigma_k^2} \qquad (30)$$

In order to derive Eq. (10) from Eq. (9), we need to show that the second term of Eq. (9) goes to $\boldsymbol{l'_{FD}(\mu)}$. i.e., we need to show that $\boldsymbol{g} := \mathop{\mathbb{E}}_{q_{\boldsymbol{\lambda}}(\boldsymbol{x})}\left[\nabla_{\boldsymbol{\mu}} \log q_{\boldsymbol{\lambda}}(\boldsymbol{x}) \boldsymbol{l'_{FD}(\mu)}^T (\boldsymbol{x} - \boldsymbol{\mu})\right] = \boldsymbol{l'_{FD}(\mu)}$ with

$$g_k := \mathop{\mathbb{E}}_{q_{\boldsymbol{\mu}}(\boldsymbol{x})}\left[\nabla_{\boldsymbol{\mu}} \log q_{\boldsymbol{\mu}}(\boldsymbol{x}) \boldsymbol{l'_{FD}(\mu)}^T (x_k - \mu_k)\right]. \tag{31}$$

Let $r = \{1, \ldots, N_{\mathcal{X}}\} \setminus \{k\}$ and $l_k^{'}(\boldsymbol{\mu})$ denote the k^{th} element of the vector $\boldsymbol{l'_{FD}(\mu)}$. By substituting Eq. (30) in Eq. (31) and using the definition of $q_\lambda(\boldsymbol{x})$ we get,

$$\begin{aligned}
g_k = \frac{1}{(2\pi)^{N_{\mathcal{X}}/2}} \int_{\Omega_{\mathcal{X}_r}} \Bigg(\int_{\mathcal{X}_k} l_k^{'}(\boldsymbol{\mu})(x_k - \mu_k) \left(\frac{x_k - \mu_k}{\sigma_k^2}\right) \frac{1}{\sigma_k} \exp\left(-\frac{1}{2}\left(\frac{x_k - \mu_k}{\sigma_k}\right)^2\right) dx_k \Bigg) \\
\frac{1}{\prod_r \sigma_r} \exp\left(-\frac{1}{2} \sum_r \left(\frac{x_r - \mu_r}{\sigma_r}\right)^2\right) \boldsymbol{dx_r} \\
+ \frac{1}{(2\pi)^{N_{\mathcal{X}}/2}} \int_{\Omega_{\mathcal{X}_r}} l_r^{'}(\boldsymbol{\mu})(x_r - \mu_r) \Bigg(\int_{\Omega_{\mathcal{X}_k}} \left(\frac{x_k - \mu_k}{\sigma_k^2}\right) \frac{1}{\sigma_k} \exp\left(-\frac{1}{2}\left(\frac{x_k - \mu_k}{\sigma_k}\right)^2\right) dx_k \Bigg) \\
\frac{1}{\prod_r \sigma_r} \exp\left(-\frac{1}{2} \sum_r \left(\frac{x_r - \mu_r}{\sigma_r}\right)^2\right) \boldsymbol{dx_r}.
\end{aligned} \tag{32}$$

Here, $g_k = g_{k_1} + g_{k_2}$, where g_{k_1} and g_{k_2} refers to the first and the second term in Eq. (32).

Now, let us simplify g_{k_1} by setting $\frac{x-\mu}{\sigma} = t$

$$\begin{aligned}
g_{k_1} &= \frac{1}{(2\pi)^{N_{\mathcal{X}}/2}} \int_{\Omega_{\mathcal{X}_r}} \left(\int_{-\infty}^{\infty} l_k'(\boldsymbol{\mu}) \sigma_k t_k \frac{t_k}{\sigma_k} \frac{1}{\sigma_k} \exp\left(-\frac{t_k^2}{2}\right) \sigma_k dt_k \right) \frac{1}{\prod_r \sigma_r} \exp\left(-\frac{1}{2} \sum_r t_r^2\right) \prod_r \sigma_r \boldsymbol{dt_r} \\
&= l_k'(\boldsymbol{\mu}) \underbrace{\int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty}}_{\text{r times}} \frac{1}{(2\pi)^{\frac{N_{\mathcal{X}}-1}{2}}} \exp\left(-\frac{1}{2} \sum_r t_r^2\right) \boldsymbol{dt_r} \\
&= l_k'(\boldsymbol{\mu})
\end{aligned} \tag{33}$$

Similarly, let us simplify g_{k_2} next by setting $\frac{x-\mu}{\sigma} = t$

$$\begin{aligned}
g_{k_2} &= \frac{1}{(2\pi)^{N_{\mathcal{X}}/2}} \int_{\Omega_{\mathcal{X}_r}} l_r'(\boldsymbol{\mu}) \prod_r (\sigma_r t_r) \left(\int_{-\infty}^{\infty} \frac{t_k}{\sigma_k} \frac{1}{\sigma_k} \exp\left(-\frac{t_k^2}{2}\right) \sigma_k dt_k \right) \frac{1}{\prod_r \sigma_r} \exp\left(-\frac{1}{2} \sum_r t_r^2\right) \prod_r \sigma_r \boldsymbol{dt_r} \\
&= \frac{l_r'(\boldsymbol{\mu}) \prod_r \sigma_r}{(2\pi)^{\frac{N_{\mathcal{X}}-1}{2}} \sigma_k} \int_{\Omega_{\mathcal{X}}} \left(\int_{-\infty}^{\infty} t_k \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{t_k^2}{2}\right) dt_k \right) \prod_r t_r \exp\left(-\frac{1}{2} \sum_r t_r^2\right) \boldsymbol{dt_r} \quad \left(\int_{-\infty}^{\infty} t_k \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{t_k^2}{2}\right) dt_k \to 0\right) \\
&= 0
\end{aligned} \tag{34}$$

So, from Eq. (33) and Eq. (34), $g_k = l_k'(\boldsymbol{\mu})$, i.e., $\boldsymbol{g} = \boldsymbol{l'_{FD}(\mu)}$.

C Derivation of Eq. (12)

From Eq. (29),

$$\nabla_{\gamma_k} \log q_\lambda(\boldsymbol{x}) = -1 + \left(\frac{x_k - \mu_k}{e^{\gamma_k}}\right)^2 \tag{35}$$

Now again, in order to derive Eq. (12) from Eq. (9), we need to show that the second term of Eq. (9) goes to $\mathbf{0}$. i.e., we need to show that $\boldsymbol{h} := \mathop{\mathbb{E}}_{q_{\boldsymbol{\lambda}}(\boldsymbol{x})}\left[\nabla_{\boldsymbol{\gamma}} \log q_{\boldsymbol{\lambda}}(\boldsymbol{x}) \boldsymbol{l'_{FD}(\gamma)}^T (\boldsymbol{x} - \boldsymbol{\gamma})\right] = \mathbf{0}$ with

$$h_k := \mathop{\mathbb{E}}_{q_{\boldsymbol{\lambda}}(\boldsymbol{x})}\left[\nabla_{\boldsymbol{\gamma}} \log q_{\boldsymbol{\lambda}}(\boldsymbol{x}) \boldsymbol{l'_{FD}(\gamma)}^T (x_k - \gamma_k)\right]. \tag{36}$$

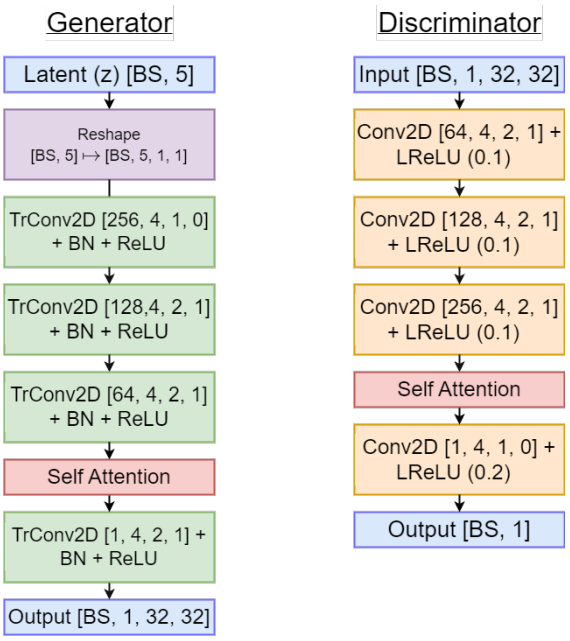


Figure 6: Architectures for the initial condition inversion problem (Section 4.3)

D Details of the model architectures

In this section we describe the architectures of different deep learning models (used as a prior or surrogate forward models) for various inverse problems. Some of the nomenclature we use are as follows:

1. FC(n) — Fully connected layer of width n.
2. LReLU(α), ReLU TanH — Leaky rectified linear unit (with negative slope parameter α), rectified linear unit, and hyperbolic tangent activation functions, respectively.
3. BN — batch normalization.
4. Conv2D (c_{out}, k, s, p) — 2D convolution layer c_{out} output channels, kernel size (k, k), stride s and padding p.
5. Conv2D (c_{out}, (k_v, k_h), s, p) — 2D convolution layer c_{out} output channels, kernel size (k_v, k_h), stride s and padding p.
6. Tr. Conv2D (c_{out}, k, s, p) — 2D transpose convolution layer with c_{out} output channels, kernel size (k, k), stride s, padding p.
7. Tr. Conv2D (c_{out}, (k_v, k_h), s, p) — 2D transpose convolution layer with c_{out} output channels, kernel size (k_v, k_h), stride s, padding p.

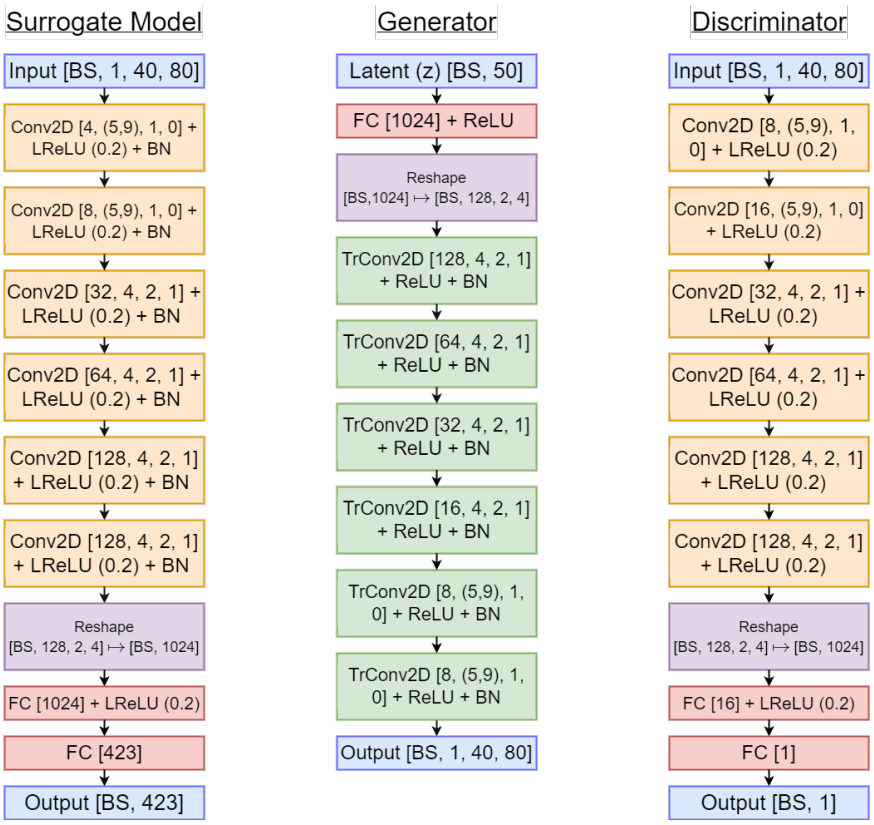


Figure 7: Architectures for the hydraulic tomography problem (Section 4.4)