

Highlights

Multi-fidelity Hamiltonian Monte Carlo

Dhruv V. Patel, Jonghyun Lee, Matthew W. Farthing, Tyler Hesser, Peter K. Kitanidis, Eric F. Darve

- “Gradient-free” Hamiltonian Monte Carlo for high-dimensional Bayesian inference.
- Combines efficient LF ML models with accurate HF numerical solvers.
- Opens up the applicability of HMC to black-box solvers.
- Improved computational efficiency and accuracy over HMC by several orders of magnitude (in both white-box and black-box setting).
- Effective for a range of inverse problems involving in-silico and experimental data.

Multi-fidelity Hamiltonian Monte Carlo

Dhruv V. Patel^a, Jonghyun Lee^b, Matthew W. Farthing^c, Tyler Hesser^c, Peter K. Kitanidis^d, Eric F. Darve^a

^a*Department of Mechanical Engineering, Stanford University, Stanford, CA, USA*

^b*Department of Civil and Environmental Engineering, University of Hawai‘i at Manoa, Honolulu, HI, USA*

^c*U.S. Army Engineering Research and Development Center, Vicksburg, MS, USA*

^d*Department of Civil and Environmental Engineering, Stanford University, Stanford, CA, USA*

Abstract

Numerous applications in biology, statistics, science, and engineering require generating samples from high-dimensional probability distributions. In recent years, the Hamiltonian Monte Carlo (HMC) method has emerged as a state-of-the-art Markov chain Monte Carlo (MCMC) technique that exploits the geometry of the target distribution to generate samples in such high dimensional spaces in an efficient manner. Despite its impressive empirical success and increasing popularity, its wide-scale adoption is still limited due to the high computational cost associated with gradient calculation. Moreover, the application of this method is simply not possible in scenarios where the gradient of the posterior cannot be computed (for example, with black-box simulators and/or with non-differentiable priors). To overcome these challenges, we propose a novel two-stage Hamiltonian Monte Carlo algorithm with a differentiable surrogate model. In this two-stage algorithm, the acceptance probability is computed in the first stage via a standard HMC proposal using an inexpensive differentiable surrogate model (which can be based on deep learning (DL)-based surrogate), and, if the proposal is accepted, the full posterior is evaluated in the second stage using the high fidelity (HF) numerical solver. Splitting the standard HMC algorithm into these two stages allows for the efficient and computationally inexpensive evaluation of the gradient of the posterior using automatic differentiation capabilities of DL-based surrogates (thus retaining advantages of HMC, such as scalability to high dimensions and faster convergence) while producing accurate posterior samples by using HF numerical solvers in the second stage. The proposed method is flexible and effective with surrogate models of different fidelities. We demonstrate the effectiveness of this algorithm for a range of problems including linear and nonlinear Bayesian inverse problems with *in-silico* data and a nonlinear hydraulic tomography problem using experimental data. In different problems, our method outperforms the traditional HMC algorithm in computational and statistical efficiency by several orders of magnitude while retaining or improving the accuracy in computed posterior statistics.

Keywords: Inverse problems, Bayesian inference, Hamiltonian Monte Carlo, Multi-fidelity modeling, Uncertainty quantification

PACS: 0000, 1111

2000 MSC: 0000, 1111

1. Introduction

The ability to generate samples from a probability distribution (which might be known only up to a normalizing constant) has wide applications. This capability is of paramount importance for generating equilibrium configurations of bio-molecules (Boomsma et al., 2013; Habeck et al., 2005), computing high-dimensional integrals in statistics (Gelfand and Smith, 1990; Brooks et al., 2011) and machine learning (Andrieu et al., 2003), for deep generative modeling tasks (Dinh et al., 2017; Nijkamp et al., 2019), and for computing desired quantities of interest for Bayesian inverse problems (Dashti and Stuart, 2017; Martin et al., 2012; Kaipio and Somersalo, 2006). Markov Chain Monte Carlo (MCMC) is a popular method for generating such samples (Metropolis et al., 1953; Gelman et al., 2014). The strength of this method lies in the fact that it can produce unbiased samples with convergence guarantees for the Quantities of Interest (QoIs) with minimal requirements on the target distribution. This robustness however comes at a price of slow convergence—exploring all the relevant parts of the parameter space that has non-negligible probability mass under the given distribution may take a long time, as vanilla MCMC methods (such as random walk Metropolis-Hastings) typically proceed by taking random jumps around the current position without taking into account the shape of the target distribution.

Over the years, a variety of methods have been developed to overcome the challenges associated with this slow convergence. Many of these methods achieve faster convergence by exploiting the geometry of the target probability density. In this manuscript, we limit ourselves to one such method—Hamiltonian (or Hybrid) Monte Carlo (HMC) method (Duane et al., 1987; Gelman et al., 2014), which is a state-of-the-art MCMC method that utilizes the gradient information of the target probability density to improve the overall convergence rate. HMC has shown tremendous promise in a variety of application domains due to its ability to characterize *high-dimensional* probability densities at *a faster convergence* rate.

Despite such promising features, there are still a few hindrances that restrict the wide-scale adoption of the HMC method. The main barrier is computing the gradient of the target probability density. For most of the practical problems of interest, the analytical formula of this gradient is not available and it has to be computed numerically. Computing such numerical gradient is simply not possible in many practical settings, for example, with black-box forward model simulators and/or with non-differentiable priors. For such scenarios, one has to revert back to other inefficient MCMC methods that do not require gradient computations. Even for the application settings where the numerical gradient can be obtained, computing it with reasonable accuracy is an expensive process. This is especially true for the problems governed by partial differential equations (PDEs) (like the ones considered in this manuscript). For such problems, evaluating one gradient typically entails solving two PDE problems (a forward and an adjoint problem). This makes the algorithm computationally demanding, as typically HMC is run for many steps ($\mathcal{O}(10^3)$ – $\mathcal{O}(10^5)$) and at each step two PDE solves are required. To overcome these challenges in this manuscript, we propose a novel Multi-Fidelity Hamiltonian Monte Carlo (MFHMC) algorithm, which leverages the

easy-to-compute-gradient of the surrogate forward model along with the high fidelity (HF) numerical solver for efficient and accurate sampling.

While the algorithm proposed in this manuscript is valid for any application of generating samples from an un-normalized probability distribution, here we focus on the problem of generating samples from the posterior density of a Bayesian inverse problem, where the forward model is typically defined by a PDE. Within this context, the method proposed here proceeds in two steps:

1. In the first step, a surrogate model is developed for the forward map (mapping parameter to solution field). One can use any surrogate model, such as a proper orthogonal decomposition (POD)-based surrogate or DNN-based surrogate model for this *offline* step. The only requirement for this surrogate model is its gradient should be computed cheaply. However, it does not necessarily have to be very accurate.
2. In the second step, this surrogate model is used in the proposed MFHMC algorithm to generate samples from the target posterior density. This algorithm, which is the novel contribution of this manuscript, is described briefly below.

The key idea of the proposed multi-fidelity algorithm (outlined in Figure 1) is to split the HMC algorithm into two stages:

- Stage 1. In the first stage, the surrogate forward model is used (in the likelihood term) to propose samples following the standard HMC proposal. Since this surrogate model is selected in such a way that its gradient can be computed cheaply, it allows the use of the HMC algorithm (which in turn enables scalability to high-dimensional parameter space and a faster convergence rate). DNN-based models are a particularly appealing choice for such surrogate models, as (by leveraging automatic differentiation capabilities of modern machine learning libraries) it allows for computing gradients accurately at *almost no additional cost*.
- Stage 2. The samples accepted by the first stage are passed to the second stage, where the HF numerical simulator is used in the Metropolis-Hastings (MH) MCMC step. The acceptance ratio for this stage is modified to take into account the first stage. Since the MH step does not require any gradient, it opens up the opportunity of using HF *black-box simulators* in this stage to correct errors introduced by the LF surrogate model in the first stage and produce accurate posterior samples as output.

We note that two-stage MCMC algorithms have been proposed in the past ([Andrés Christen and Fox, 2012](#); [Efendiev et al., 2006](#)). However, these algorithms were developed for random walk Metropolis-Hastings MCMC algorithms, which fail to scale to high-dimensional parameter space, thus limiting their practical utility. Unlike those works, this manuscript proposes a two-stage algorithm for HMC, which is the current state-of-the-art MCMC algorithm and is a workhorse for numerous high-dimensional practical inference problems (as mentioned in the references in ([Betancourt et al., 2017](#))).

The main contributions/advantages of the proposed method are summarized below:

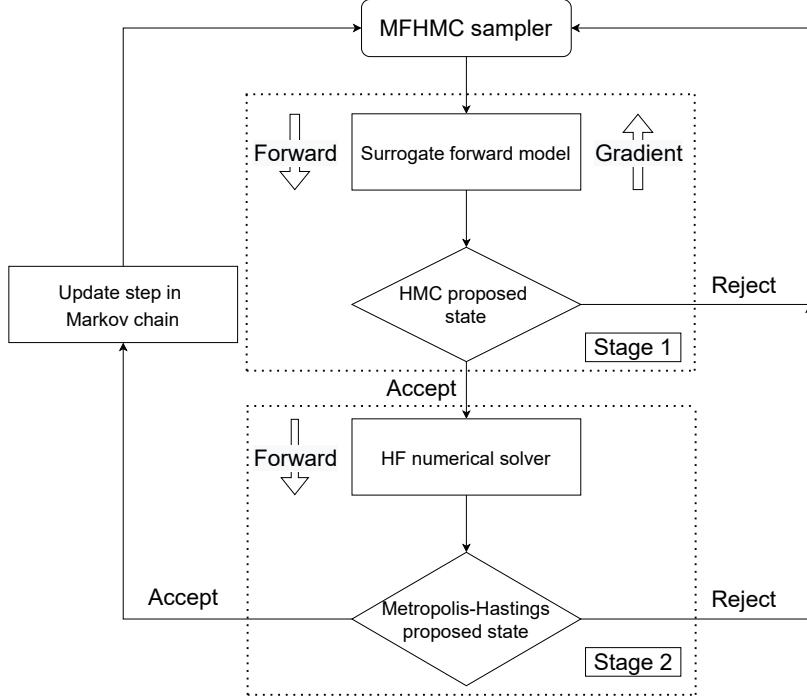


Figure 1: Outline of the proposed two-stage MFHMC algorithm: In the first stage of the algorithm a surrogate forward model (with an easy-to-compute gradient) is used in the standard HMC algorithm. If a given sample is accepted by this first stage, then it is passed to the second stage, where a high-fidelity numerical solver is used in the Metropolis-Hastings step (for which only a forward model evaluation is required with no gradient requirement) to produce accurate posterior samples.

- 1. Black-box simulators:** For many applications in science and engineering, the forward model is defined via a “black-box” simulation code. This code is either old and written in a legacy language (such as Fortran 77) or has inherent limitations such that it can only be used to solve forward problems and computing the gradient (of the forward problem solution field to its parameters) is simply infeasible and/or requires significant modification to the code-base and hence is very time-consuming and impractical. Also, many such legacy codes are not equipped with an efficient adjoint library for gradient calculation and have to resort to finite-difference-based gradient computation, which could lead to inaccurate and inefficient gradients. This bottleneck prevents the use of the standard HMC algorithm for such applications, as it requires reasonably accurate gradient information for robust performance. The method proposed in this manuscript on the other hand is perfectly suitable for such scenarios, as it can use DNN-based surrogate models (which can be trained using inputs and outputs of the black-box forward model simulator) in the first stage and by leveraging automatic differentiation capabilities can produce accurate gradients.
- 2. Computation cost:** Unlike the standard HMC algorithm, which requires solving two PDE problems at each step for gradient evaluation (i.e., one forward and one adjoint problem), we leverage the automatic differentiation capabilities of DNN-based

surrogate to efficiently compute gradients. This approach is not only computationally cheaper but also much faster. Furthermore, the acceptance rate of the standard HMC algorithm is around 60–70%, (i.e., 30 to 40% of samples are discarded). This amounts to $2 \times 0.3 \times N$ to $2 \times 0.4 \times N$ *wasted* PDE solves (where N = no. of MCMC steps). As demonstrated in the results section, our proposed algorithm significantly improves the acceptance rate, thus reducing the number of samples that are rejected.

3. **Statistical efficiency:** The proposed two-stage algorithm facilitates (is amenable to) longer Hamiltonian trajectory at each step due to computationally inexpensive surrogate model in the first stage. This results into bigger jumps and well-mixed Markov chain. This is something that is not practically feasible in the traditional HMC algorithm, as taking longer jumps results into significantly lower acceptance rate, which in turn increases the number of high-fidelity simulations required significantly making it computationally prohibitive. Moreover, due to longer jumps the effective sample size (ESS) of the resulting Markov chain is also larger for two-stage algorithm. This means the samples are more un-correlated and hence it produces low-variance statistical estimates.
4. **Non-differentiable priors:** In many applications (especially in the computer vision domain), use of non-differentiable priors is widespread. This include image denoising (Rudin et al., 1992; Chambolle et al., 2010), deblurring (Beck and Teboulle, 2009), multi-frame superresolution (Farsiu et al., 1996). Other than this, Laplace priors used in Bayesian Lasso regression (Park and Casella, 2012), sample-based priors used in Bayesian inversion (Vauhkonen et al., 1997; Patel et al., 2021), Total-Variation and Bernoulli-Laplace priors used in sparse regularization (Chaari et al., 2013; JJ, 2004; Lee and Kitanidis, 2013b) are other popular examples of such priors. Lack of differentiability of resulting posterior distribution prevents the use of the HMC method for posterior exploration with such priors. This leads to two sub-optimal design choices: (i) selecting a non-gradient-based MCMC method for posterior exploration and/or (ii) selecting a second-choice prior which is differentiable. In contrast, the method proposed in this manuscript does not face any such difficulties and can easily be deployed with non-differentiable priors.

The remainder of this paper is organized as follows: first, we provide relevant background on HMC algorithm and highlight some of the key features and challenges of this traditional HMC algorithm in Section 2. Then we propose our novel MFHMC algorithm in Section 2.4. In Section 3 we provide numerical results on a range of test problems involving synthetic and experimental data to demonstrate the applicability and the effectiveness of the proposed algorithm. Finally, we conclude the paper in Section 4 with a brief discussion and summary of the paper with potential future directions.

2. Methods

The method proposed in this manuscript is applicable for any sampling (from an unnormalized probability) task. However, to make ideas more concrete, here we limit our focus

on the sampling problem arising in the context of Bayesian inverse problems. To this end, we first introduce the classical Bayesian inverse problem and relevant notations below and then explain how such problems can be solved using traditional single stage HMC algorithm followed by our proposed two-stage MFHMC algorithm.

2.1. Bayesian Inverse Problem and Markov Chain Monte Carlo

Consider the following direct/forward problem

$$\tilde{\mathbf{y}} = \mathbf{f}(\mathbf{x}) + \boldsymbol{\eta}, \quad \mathbf{x} \in \Omega_{\mathcal{X}} \subset \mathbb{R}^N, \quad \tilde{\mathbf{y}} \in \Omega_{\mathcal{Y}} \subset \mathbb{R}^M, \quad (1)$$

where $\tilde{\mathbf{y}}$ is the measured response to some input parameter \mathbf{x} , \mathbf{f} is the direct/forward model (often described via PDE), and $\boldsymbol{\eta}$ represents measurement and/or modeling error. The goal of the inverse problem is to recover the unknown parameter \mathbf{x} from the noisy and possibly partial measurement of $\tilde{\mathbf{y}}$. Bayesian inference provides a principled probabilistic framework for solving such ill-posed problems with quantified uncertainty estimates. Within this approach, both the inferred field and the observation are modeled as realization of random variables, \mathcal{X} and \mathcal{Y} , respectively. First, a prior probability density $p_{\mathcal{X}}^{\text{prior}}(\mathbf{x})$, which captures all the constraints and the domain knowledge about the parameter to be inferred (\mathcal{X}) *prior* to observing measurement ($\tilde{\mathbf{y}}$) is assumed. Then this prior probability density is used in conjunction with the forward model (\mathbf{f}) to define the likelihood distribution of observing measurement $\mathcal{Y} = \tilde{\mathbf{y}}$ given $\mathcal{X} = \mathbf{x}$, that is $p_{\mathcal{Y}}^{\text{like}}(\tilde{\mathbf{y}}|\mathbf{x})$. For additive measurement noise model, as described in eq. (1), this likelihood distribution could be written as $p_{\mathcal{Y}}^{\text{like}}(\tilde{\mathbf{y}}|\mathbf{x}) = p_{\boldsymbol{\eta}}(\tilde{\mathbf{y}} - \mathbf{f}(\mathbf{x}))$. Using Bayes' rule this yields the following expression for the posterior distribution of \mathcal{X}

$$p_{\mathcal{X}}^{\text{post}}(\mathbf{x}|\tilde{\mathbf{y}}) = \frac{p_{\mathcal{Y}}^{\text{like}}(\tilde{\mathbf{y}}|\mathbf{x})p_{\mathcal{X}}^{\text{prior}}(\mathbf{x})}{p_{\mathcal{Y}}(\tilde{\mathbf{y}})} \propto p_{\boldsymbol{\eta}}(\tilde{\mathbf{y}} - \mathbf{f}(\mathbf{x}))p_{\mathcal{X}}^{\text{prior}}(\mathbf{x}), \quad (2)$$

where $p_{\mathcal{Y}}(\tilde{\mathbf{y}})$ is called the evidence term, which normalizes the posterior distribution $p_{\mathcal{X}}^{\text{post}}(\mathbf{x}|\tilde{\mathbf{y}})$ so that $\int_{\mathcal{X}} p_{\mathcal{X}}^{\text{post}}(\mathbf{x}|\tilde{\mathbf{y}}) d\mathbf{x} = 1$. Ideally, we would like to have access to the full posterior distribution as output to the Bayesian inference. However, since we lack the appropriate tools to visualize and understand this high-dimensional probability density, in practice, we mostly focus on computing lower-dimensional quantities of the posterior, such as the mean $\bar{\mathbf{x}} = \mathbb{E}[\mathbf{x}] = \int_{\mathcal{X}} \mathbf{x} p_{\mathcal{X}}^{\text{post}}(\mathbf{x}|\tilde{\mathbf{y}}) d\mathbf{x}$ or higher order moments, marginals, and confidence intervals.

For physics-based inverse problems, however, this is problematic as for most of such problems the dimension of \mathcal{X} is typically equal to the number of nodes (number of degrees of freedom to be more precise) in the numerical discretization scheme (such as finite element or finite volume). For complex real-world problems, this could be as high as $\mathcal{O}(10^3)\text{--}\mathcal{O}(10^9)$ due to fine spatio-temporal discretization. Computing integral over such a high-dimensional space is simply infeasible using numerical integration (such as quadrature-based) techniques and the analytical treatment of this integral is not possible (unless for a simple and not-so-practical conjugate prior cases). This motivates the use of MCMC methods that perform random walk in the parameters space and can generate samples according to the posterior probability distribution. Once these samples are generated, then a simple Monte Carlo sum can be taken of these samples to approximate the integral.

2.2. MCMC methods and Metropolis-Hastings

Markov Chain Monte Carlo (MCMC) methods can generate samples from any unnormalized probability distribution, such as $p_{\mathcal{X}}^{\text{U(post)}}(\mathbf{x}|\tilde{\mathbf{y}}) := p_{\eta}(\tilde{\mathbf{y}} - \mathbf{f}(\mathbf{x}))p_{\mathcal{X}}^{\text{prior}}(\mathbf{x})$. For this, MCMC methods construct a Markov Chain whose stationary distribution is the target unnormalized posterior density. This is achieved by given an initial distribution π_0 and a transition kernel K , constructing the following sequence of random variables:

$$X_0 \sim \pi_0, \quad X_{t+1} \sim K(\cdot|X_t). \quad (3)$$

In order for the $p_{\mathcal{X}}^{\text{U(post)}}$ to be the stationary distribution of the Markov Chain, three conditions must be satisfied: the kernel K must be irreducible and aperiodic (these are usually mild conditions and are satisfied) and $p_{\mathcal{X}}^{\text{U(post)}}$ has to be a fixed point of K . This last condition can be expressed as: $p(x') = \int K(x'|x)p_{\mathcal{X}}^{\text{U(post)}}(x)dx$. This condition is often satisfied by satisfying the detailed balance equation described as: $p_{\mathcal{X}}^{\text{U(post)}}(x')K(x|x') = p(x)K(x'|x)$.

Given any proposal distribution q , we can easily construct a transition kernel that respects detailed balance using Metropolis-Hastings accept/reject rules. More formally, starting from $x_0 \sim \pi_0$, at each step t , we sample $x' \sim q(\cdot|X_t)$, and with probability

$$\alpha(x'|x_t) = \min\left(1, \frac{p(x')q(x_t|x')}{p(x_t)q(x'|x_t)}\right),$$

accept x' as the next sample x_{t+1} in the chain and reject the sample x' with the probability $1 - \alpha(x'|x_t)$ and retain the previous state, i.e. $x_{t+1} = x_t$. For typical proposals, this algorithm converges to the target density in an asymptotic sense. However, this comes at the cost of very slow mixing (very slow traversal along the parameter space) as these algorithms are simply taking random jumps without taking into account the geometry of the target distribution. Hamiltonian Monte Carlo (HMC) method tackles this problem of slow mixing by exploiting the geometry of the target density. Specifically, it uses the gradient information of the target density to make bigger proposal jumps in the parameter space along the regions of high probability.

2.3. Hamiltonian Monte Carlo

In Hamiltonian Monte Carlo algorithm (first introduced as “Hybrid Monte Carlo” in (Duane et al., 1987)), the state space to be inferred (\mathbf{x}) is augmented with a fictitious momentum variable $\boldsymbol{\xi} \in \mathbb{R}^N$. Then the canonical joint posterior density is defined as

$$p^{\text{post}}(\mathbf{x}, \boldsymbol{\xi}|\tilde{\mathbf{y}}) \propto \exp\{-H(\mathbf{x}, \boldsymbol{\xi})\},$$

where $H(\mathbf{x}, \boldsymbol{\xi}) := -\log p^{\text{post}}(\mathbf{x}|\tilde{\mathbf{y}}) + \frac{\|\boldsymbol{\xi}\|^2}{2}$ is the Hamiltonian of the system. Here, the Hamiltonian $H(\mathbf{x}, \boldsymbol{\xi}) := U(\mathbf{x}) + K(\boldsymbol{\xi})$, where the potential energy $U(\mathbf{x}) := -\log p^{\text{post}}(\mathbf{x}|\tilde{\mathbf{y}})$ is a function of \mathbf{x} only and the kinetic energy $K(\boldsymbol{\xi}) := \frac{\|\boldsymbol{\xi}\|^2}{2}$ is a function of $\boldsymbol{\xi}$ only. This separation of variables help in defining the update rule for each variable. Interested readers are referred to (Neal, 2012) for more detailed discussion on HMC.

Sampling in the single-stage HMC proceeds by the following iteration from point $(\mathbf{x}^j, \boldsymbol{\xi}^j)$.

1. Draw $\tilde{\xi}^j \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_N)$.
2. Let $(\mathbf{x}(t), \xi(t))$ be the time t solution to Hamilton's equations of motion: $\dot{\mathbf{x}} = \xi, \dot{\xi} = \nabla \log p^{\text{post}}(\mathbf{x}|\hat{\mathbf{y}})$, with initial condition $(\mathbf{x}^j, \tilde{\xi}^j)$.
3. Set $(\mathbf{x}^{j+1}, \xi^{j+1}) = (\mathbf{x}(t), -\xi(t))$, for integration time t .
4. Compute acceptance probability
$$\alpha((\mathbf{x}^j, \xi^j) \rightarrow (\mathbf{x}^{j+1}, \xi^{j+1})) = \min(1, \exp\{H(\mathbf{x}^j, \xi^j) - H(\mathbf{x}^{j+1}, \xi^{j+1})\}).$$
5. Accept $(\mathbf{x}^{j+1}, \xi^{j+1})$ with probability $\alpha((\mathbf{x}^j, \xi^j) \rightarrow (\mathbf{x}^{j+1}, \xi^{j+1}))$. Set $(\mathbf{x}^{j+1}, \xi^{j+1}) = (\mathbf{x}^j, \xi^j)$ otherwise.

Algorithm 1: Single Stage Hamiltonian Monte Carlo Algorithm

Input: $U(\mathbf{x}) = -\log p^{\text{post}}(\mathbf{x}|\hat{\mathbf{y}})$, $K(\xi) = \xi^T \xi / 2$, step size (ϵ), number of leapfrog steps (L), number of HMC steps (m)

Result: $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m$

Choose a starting point \mathbf{z}_0 ;

for $i = 1, \dots, m$ **do**

 Draw $\xi_0 \sim \mathcal{N}(\mathbf{0}, 1)$;

 Make half-step update of ξ ;

$$\xi = \xi_0 - \epsilon \nabla_{\mathbf{z}} U / 2;$$

for $i = 1, \dots, L$ **do**

 Make full step update of \mathbf{x} ;

$$\mathbf{x} = \mathbf{x} + \epsilon \xi;$$

 Make full step update of ξ ;

$$\xi = \xi - \epsilon \nabla_{\mathbf{z}} U;$$

end

 Make half-step update of ξ ;

$$\xi = \xi - \epsilon \nabla_{\mathbf{z}} U / 2;$$

$$\xi = -\xi;$$

 Compute acceptance probability;

$$\alpha(\mathbf{x}_{i-1}, \mathbf{x}^*) = \min\{1, \exp(-U(\mathbf{x}) - K(\xi) + U(\mathbf{x}_0) + K(\xi_0))\};$$

 Set \mathbf{x}_i to

$$\mathbf{x}_i = \begin{cases} \mathbf{x}^*, & \text{with probability } \alpha(\mathbf{x}^*, \mathbf{x}_{i-1}) \\ \mathbf{x}_{i-1}, & \text{with probability } 1 - \alpha(\mathbf{x}^*, \mathbf{x}_{i-1}) \end{cases}$$

end

return $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m$;

As described above, the single-stage HMC algorithm proceeds by taking steps by solving the coupled ODE systems describing the Hamiltonian dynamics. This requires having access to the gradient of the posterior distribution. This in turn requires expensive computation of the gradient of the forward model. In contrast, in the proposed two-stage HMC algorithm proposed below, no gradient of the high-fidelity forward model is required, which leads to significant computation savings as described in the [3](#) section

2.4. Multi-fidelity HMC (MFHMC)

Let \mathbf{f}^* denote the surrogate forward model for \mathbf{f} and let $p^{\text{post}^*}(\mathbf{x}|\hat{\mathbf{y}})$ denote the posterior density induced by this surrogate model. The basic idea of the proposed two-stage HMC algorithm is to use \mathbf{f}^* (and corresponding $p^{\text{post}^*}(\mathbf{x}|\hat{\mathbf{y}})$) in a standard HMC iteration (described above) as the first stage. Then if the sample is rejected by this first stage, we stop there and proceed with the next MCMC iteration. However, if the sample is accepted, then we proceed to the second stage, where the following acceptance probability is computed

$$\alpha^{HF}((\mathbf{x}^j, \boldsymbol{\xi}^j) \rightarrow (\mathbf{x}^{j+1}, \boldsymbol{\xi}^{j+1})) = \min\{1, \exp(r)\},$$

where,

$$r = \frac{p^{\text{post}}(\mathbf{x}^{j+1}|\hat{\mathbf{y}}) p^{\text{post}^*}(\mathbf{x}|\hat{\mathbf{y}})}{p^{\text{post}}(\mathbf{x}|\hat{\mathbf{y}}) p^{\text{post}^*}(\mathbf{x}^{j+1}|\hat{\mathbf{y}})}.$$

Then, the sample $(\mathbf{x}^{j+1}, \boldsymbol{\xi}^{j+1})$ is accepted with probability $\alpha^{HF}((\mathbf{x}^j, \boldsymbol{\xi}^j) \rightarrow (\mathbf{x}^{j+1}, \boldsymbol{\xi}^{j+1}))$ or we retain the previous sample $(\mathbf{x}^j, \boldsymbol{\xi}^j)$ with probability $1 - \alpha^{HF}((\mathbf{x}^j, \boldsymbol{\xi}^j) \rightarrow (\mathbf{x}^{j+1}, \boldsymbol{\xi}^{j+1}))$. Here, r is selected such that the resulting Markov chain satisfies the detailed balance equation.

Note that in this two-stage algorithm, the gradient computation is only required in the first stage for the HMC proposal, where the surrogate model \mathbf{f}^* is used (which is selected in such a way that the gradient computation using \mathbf{f}^* is inexpensive and fast), whereas the HF forward model \mathbf{f} is used in the second stage, which corrects the error introduced by the surrogate forward model in the first stage and for this *no gradient calculation is required*. Next, we provide the full two-stage algorithm and the derivation of the acceptance probability formula for the second stage.

Algorithm 2: Multi-fidelity Hamiltonian Monte Carlo (MFHMC) Algorithm

Input: $p^{\text{post}(HF)}$, $p^{\text{post}(LF)}$, step size (ϵ), number of leapfrog steps (L),
 $U(\mathbf{x}) := -\log p^{\text{post}(LF)}$, $K(\boldsymbol{\xi}) := \boldsymbol{\xi}^T \boldsymbol{\xi} / 2$

Result: $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m$

Choose a starting point \mathbf{x}_0 ;

for $i = 1, \dots, m$ **do**

$\mathbf{x}^{LF} = \mathbf{x}_{i-1}$;

Draw $\boldsymbol{\xi}_0 \sim \mathcal{N}(\mathbf{0}, \mathbf{1})$;

Make half-step update of $\boldsymbol{\xi}$;
 $\boldsymbol{\xi} = \boldsymbol{\xi}_0 - \epsilon \nabla_{\mathbf{z}} U / 2$;

for $l = 1, \dots, L$ **do**

Make full step update of \mathbf{z} ;
 $\mathbf{x}^{LF} = \mathbf{x}^{LF} + \epsilon \boldsymbol{\xi}$;

Make full step update of $\boldsymbol{\xi}$;
 $\boldsymbol{\xi} = \boldsymbol{\xi} - \epsilon \nabla_{\mathbf{z}} U$;

end

Make half-step update of $\boldsymbol{\xi}$;
 $\boldsymbol{\xi} = \boldsymbol{\xi} - \epsilon \nabla_{\mathbf{z}} U / 2$;

$\boldsymbol{\xi} = -\boldsymbol{\xi}$;

Compute acceptance probability;
 $\alpha^{LF}(\mathbf{x}_{i-1}, \mathbf{x}^{LF}) = \min \left\{ 1, \exp(-U(\mathbf{x}^{LF}) - K(\boldsymbol{\xi}^{LF}) + U(\mathbf{x}_{i-1}) + K(\boldsymbol{\xi}_{i-1})) \right\}$;

Set \mathbf{x}^{HF} to

$$\mathbf{x}^{HF} = \begin{cases} \mathbf{x}^{LF}, & \text{with probability } \alpha^{LF}(\mathbf{x}_{i-1}, \mathbf{x}^{LF}) \\ \mathbf{x}_{i-1}, & \text{with probability } 1 - \alpha^{LF}(\mathbf{x}_{i-1}, \mathbf{x}^{LF}) \end{cases}$$

Compute acceptance probability for HF model;
 $\alpha^{HF}(\mathbf{x}_{i-1}, \mathbf{x}^{HF}) = \min \left\{ 1, \frac{p^{\text{post}(HF)}(\mathbf{x}^{HF} | \hat{\mathbf{y}}) h(\boldsymbol{\xi}_0)}{p^{\text{post}(HF)}(\mathbf{x}_{i-1} | \hat{\mathbf{y}}) h(\boldsymbol{\xi})} \right\} \text{ where } h(\boldsymbol{\xi}) := e^{\boldsymbol{\xi}^T \boldsymbol{\xi} / 2}$;

Set sample \mathbf{x}_i to;

$$\mathbf{x}_i = \begin{cases} \mathbf{x}^{HF}, & \text{with probability } \alpha^{HF}(\mathbf{x}_{i-1}, \mathbf{x}^{HF}) \\ \mathbf{x}_{i-1}, & \text{with probability } 1 - \alpha^{HF}(\mathbf{x}_{i-1}, \mathbf{x}^{HF}) \end{cases}$$

end

return $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_m$;

2.5. Analysis of MFHMC

Next, we will analyze the MFHMC algorithm in more detail. Denote

$$\begin{aligned} \mathcal{E} &= \{(\mathbf{x}, \boldsymbol{\xi}); p^{\text{post}(HF)}(\mathbf{x}, \boldsymbol{\xi}) > 0\} \\ \mathcal{E}^* &= \{(\mathbf{x}, \boldsymbol{\xi}); p^{\text{post}(LF)}(\mathbf{x}, \boldsymbol{\xi}) > 0\} \\ \mathcal{D} &= \{(\mathbf{x}, \boldsymbol{\xi}); q(\mathbf{x}, \boldsymbol{\xi} | \mathbf{x}_{i-1}, \boldsymbol{\xi}_{i-1}) > 0 \text{ for some } (\mathbf{x}_{i-1}, \boldsymbol{\xi}_{i-1}) \in \mathcal{E}\}. \end{aligned}$$

The set \mathcal{E} is the support of the target posterior distribution $p^{\text{post(HF)}}(\mathbf{x}, \boldsymbol{\xi})$. It contains all the possible tuples of the inferred variable and momentum variable $(\mathbf{x}, \boldsymbol{\xi})$ which has a positive probability of being accepted as a sample. Similarly, \mathcal{E}^* is the support of the low fidelity posterior distribution $p^{\text{post(LF)}}(\mathbf{x}, \boldsymbol{\xi})$, which contains all tuples $(\mathbf{x}, \boldsymbol{\xi})$ which can be accepted by the first stage of our algorithm. \mathcal{D} contains proposals which can be generated by the proposal distribution $q(\mathbf{x}, \boldsymbol{\xi} | \mathbf{x}_{i-1}, \boldsymbol{\xi}_{i-1})$. For the MFHMC algorithm to work properly, the following two conditions must hold all the time: $\mathcal{E} \subseteq \mathcal{D}$ and $\mathcal{E} \subseteq \mathcal{E}^*$. If one of these conditions is not true, for example, say, $\mathcal{E} \not\subseteq \mathcal{E}^*$, then there will exist a subset $K \subset (\mathcal{E} \setminus \mathcal{E}^*)$ such that

$$p^{\text{post(HF)}}(K) = \int_K p^{\text{post(HF)}}(\mathbf{x}, \boldsymbol{\xi}) d\mathbf{x} d\boldsymbol{\xi} > 0 \quad \text{and} \quad p^{\text{post(LF)}}(K) = \int_K p^{\text{post(LF)}}(\mathbf{x}, \boldsymbol{\xi}) d\mathbf{x} d\boldsymbol{\xi} = 0$$

which means no elements of K can pass the first stage of MFHMC and K will never be visited by the final chain despite its samples having a positive probability of being accepted by $p^{\text{post(HF)}}$ and hence the resulting Markov chain will not be sampled properly.

For most practical purposes, the conditions $\mathcal{E}, \mathcal{E}^* \subset \mathcal{D}$ can be naturally satisfied by selection of appropriate proposal distribution $q(\mathbf{x}, \boldsymbol{\xi} | \mathbf{x}_{i-1}, \boldsymbol{\xi}_{i-1})$. By choosing the appropriate value of the likelihood variance in $p^{\text{post(LF)}}$ the condition $\mathcal{E} \subset \mathcal{E}^*$ can also be satisfied. Thus $\mathcal{E} \subset \mathcal{E}^* \subset \mathcal{D}$. In this case, \mathcal{E}^* is identical to the support of the transition probability distribution $Q(\mathbf{x}, \boldsymbol{\xi} | \mathbf{x}_{i-1}, \boldsymbol{\xi}_{i-1}) = \alpha^{\text{(LF)}}((\mathbf{x}_{i-1}, \boldsymbol{\xi}_{i-1}), (\mathbf{x}, \boldsymbol{\xi})) q(\mathbf{x}, \boldsymbol{\xi} | \mathbf{x}_{i-1}, \boldsymbol{\xi}_{i-1})$:

$$\mathcal{E}^* = \{(\mathbf{x}, \boldsymbol{\xi}); Q(\mathbf{x}, \boldsymbol{\xi} | \mathbf{x}_{i-1}, \boldsymbol{\xi}_{i-1}) > 0 \text{ for some } (\mathbf{x}_{i-1}, \boldsymbol{\xi}_{i-1}) \in \mathcal{E}\}.$$

Due to the very high dimension of the joint field $(\mathbf{x}, \boldsymbol{\xi})$, the support \mathcal{E} of the target distribution $p^{\text{post(HF)}}$ is much smaller than the support \mathcal{D} of the proposal distribution $q(\mathbf{x}, \boldsymbol{\xi} | \mathbf{x}_{i-1}, \boldsymbol{\xi}_{i-1})$. For all the proposed samples $(\mathbf{x}, \boldsymbol{\xi}) \in (\mathcal{D} \setminus \mathcal{E})$, they will never be accepted as valid samples in the final Markov chain in the traditional single-stage HMC algorithm resulting in huge computation waste since $p^{\text{post(HF)}}(\mathbf{x}, \boldsymbol{\xi}) = 0$. In the proposed MFHMC algorithm, however, the transition probability distribution $Q(\mathbf{x}, \boldsymbol{\xi} | \mathbf{x}_{i-1}, \boldsymbol{\xi}_{i-1})$ (which acts as an effective proposal distribution for the second stage) samples from a much smaller support \mathcal{E}^* , hence avoids solving expensive HF problem for all $(\mathbf{x}, \boldsymbol{\xi}) \in \mathcal{D} \setminus \mathcal{E}^*$. For each iteration, the MFHMC algorithm only requires the HF simulation r times in average, where

$$r = \int_{\mathcal{E}^*} \alpha^{\text{(LF)}}((\mathbf{x}_{i-1}, \boldsymbol{\xi}_{i-1}), (\mathbf{x}, \boldsymbol{\xi})) q(\mathbf{x}, \boldsymbol{\xi} | \mathbf{x}_{i-1}, \boldsymbol{\xi}_{i-1}) < 1.$$

Note that $\int_{\mathcal{D}} q(\mathbf{x}, \boldsymbol{\xi} | \mathbf{x}_{i-1}, \boldsymbol{\xi}_{i-1}) = 1$ and $\alpha^{\text{(LF)}}((\mathbf{x}_{i-1}, \boldsymbol{\xi}_{i-1}), (\mathbf{x}, \boldsymbol{\xi})) \leq 1$. If \mathcal{E}^* is close to \mathcal{E} and hence much smaller than \mathcal{D} , then $r \ll 1$. Therefore, the proposed MFHMC method requires much fewer HF simulations while approximately still accepting the same amount of proposals. In other words, the MFHMC algorithm can achieve a much higher acceptance rate for each HF simulation.

Stability properties of the MFHMC

Next we will discuss the stability properties of the MFHMC algorithm and show that it shares the same convergence property as the traditional HMC algorithm. For this we will

show that the resulting Markov chain is ergodic, irreducible, and is aperiodic by proving the satisfaction of the detailed balance equation. Denote by \mathcal{K} the transition kernel of the markov chain \mathbf{x}_i generated by the MFHMC algorithm and by $\mathbf{s} = (\mathbf{x}, \boldsymbol{\xi})$. Since its effective proposal distribution is given by

$$Q(\mathbf{s}|\mathbf{s}_{i-1}) = \alpha^{(\text{LF})}(\mathbf{s}_{i-1}, \mathbf{s})q(\mathbf{s}|\mathbf{s}_{i-1}) + \left(1 - \int \alpha^{(\text{LF})}(\mathbf{s}_{i-1}, \mathbf{s})q(\mathbf{s}|\mathbf{s}_{i-1})\right) \delta_{\mathbf{s}_{i-1}}(\mathbf{s}) \quad (4)$$

The transition kernel of the overall Markov chain is given by

$$\begin{aligned} \mathcal{K}(\mathbf{s}_{i-1}, \mathbf{s}) &= \alpha^{(\text{HF})}(\mathbf{s}_{i-1}, \mathbf{s})Q(\mathbf{s}|\mathbf{s}_{i-1}) \text{ for } \mathbf{s} \neq \mathbf{s}_{i-1} \\ \mathcal{K}(\mathbf{s}_{i-1}, \mathbf{s}) &= 1 - \int_{\mathbf{s} \neq \mathbf{s}_{i-1}} \alpha^{(\text{HF})}(\mathbf{s}_{i-1}, \mathbf{s})Q(\mathbf{s}|\mathbf{s}_{i-1}) \text{ for } \mathbf{s} = \mathbf{s}_{i-1} \end{aligned}$$

It is easy to show that this transition kernel satisfies the detailed balance equation.

$$p^{\text{post(HF)}}(\mathbf{s}_{i-1})\mathcal{K}(\mathbf{s}, \mathbf{s}_{i-1}) = p^{\text{post(HF)}}(\mathbf{s})\mathcal{K}(\mathbf{s}_{i-1}, \mathbf{s}) \quad (5)$$

for any $\mathbf{s}, \mathbf{s}_{i-1} \in \mathcal{E}$. The above equality is obviously true when $\mathbf{s} = \mathbf{s}_{i-1}$. When $\mathbf{s} \neq \mathbf{s}_{i-1}$ then from the definition of the transition kernel of the overall Markov chain we have

$$\begin{aligned} p^{\text{post(HF)}}(\mathbf{s}_{i-1})\mathcal{K}(\mathbf{s}_{i-1}, \mathbf{s}) &= p^{\text{post(HF)}}(\mathbf{s}_{i-1})\alpha^{(\text{HF})}(\mathbf{s}_{i-1}, \mathbf{s})Q(\mathbf{s}|\mathbf{s}_{i-1}) \\ &= p^{\text{post(HF)}}(\mathbf{s}_{i-1})Q(\mathbf{s}|\mathbf{s}_{i-1})\min\left\{1, \frac{p^{\text{post(HF)}}(\mathbf{s})p^{\text{post(LF)}}(\mathbf{s}_{i-1})}{p^{\text{post(HF)}}(\mathbf{s}_{i-1})p^{\text{post(LF)}}(\mathbf{s})}\right\} \\ &= \min\{Q(\mathbf{s}|\mathbf{s}_{i-1})p^{\text{post(HF)}}(\mathbf{s}_{i-1}), Q(\mathbf{s}_{i-1}|\mathbf{s})p^{\text{post(HF)}}(\mathbf{s})\} \\ &= \min\left\{\frac{Q(\mathbf{s}|\mathbf{s}_{i-1})p^{\text{post(HF)}}(\mathbf{s}_{i-1})}{Q(\mathbf{s}_{i-1}|\mathbf{s})p^{\text{post(HF)}}(\mathbf{s})}, 1\right\}Q(\mathbf{s}_{i-1}|\mathbf{s})p^{\text{post(HF)}}(\mathbf{s}) \\ &= \alpha^{(\text{HF})}(\mathbf{s}, \mathbf{s}_{i-1})Q(\mathbf{s}_{i-1}|\mathbf{s})p^{\text{post(HF)}}(\mathbf{s}) \\ &= p^{\text{post(HF)}}(\mathbf{s})\mathcal{K}(\mathbf{s}, \mathbf{s}_{i-1}). \end{aligned}$$

so the detailed balance is always satisfied. Using the above relation we can easily show that $p^{\text{post(HF)}}(A) = \int \mathcal{K}(\mathbf{s}, A)d\mathbf{s}$ for any $A \in \mathcal{B}(A)$, where \mathcal{B} denotes all the Borel measurable subset of \mathcal{E} and hence $p^{\text{post(HF)}}$ is indeed the stationary distribution of \mathcal{K} .

3. Numerical Validation and Results

In this section, we provide numerical results for the proposed MFHMC algorithm. We start the section by performing a systematic study on a high-dimensional target distribution by selecting different low-fidelity models and comparing the performance of MFHMC with these different lower-fidelity models with HMC at different computational budgets (as defined by the number of target density evaluations). Then we consider both linear as well as non-linear Bayesian inverse problems and verify our results using both *in-silico* as well as experimental data.

Dataset and Inverse problems: We consider the initial condition inversion problem in the transient diffusion equation and the coefficient inversion problem for the steady state diffusion equation as our test problems. For the initial condition inversion problem, we use a parametric (rectangular) dataset. This is a synthetic dataset that allows us to compute and compare the obtained posterior statistics with the reference statistics. For the conductivity (coefficient) inversion problem, we consider the parametric (rectangular) as well as channelized flow dataset. In addition to that, we also consider a hydraulic tomography dataset for which we use both *in-silico* as well as experimental data to demonstrate the effectiveness of the proposed algorithm on a real-world problem.

Baseline: For baseline we consider a single-stage HMC algorithm, which is currently one of the state-of-the-art MCMC algorithms.

Performance evaluation metrics: In order to compare the relative performance of the proposed MFHMC algorithm to the standard single-stage HMC algorithm, we consider two different practical scenarios commonly encountered in the real world, and for each scenario, we use different performance metrics for robust comparison of the two methods. These scenarios and the related metrics are as follows:

1. Fixed computation budget, i.e., a fixed number of HF evaluations:
 - computation time
 - error in posterior statistics
2. Fixed error in posterior stats, i.e., a fixed number of MCMC steps
 - effective sample size (ESS)
 - mean squared jump distance (MSJD)
 - number of HF evaluations

3.1. 250-dimensional multivariate normal (MVN)

The purpose of this study is to do a systematic quantitative evaluation of MFHMC versus HMC at different fixed computation budgets. To perform this, we consider a problem where the target distribution (and its statistics such as mean and covariance) are known so that the accuracy of these statistics can be computed.

We consider a 250-dimensional multivariate normal distribution with zero mean and a known precision matrix A^{HF} , i.e.,

$$p^{\text{post(HF)}}(\mathbf{x}) := \mathcal{N}(\mathbf{0}, \Sigma^{HF} = A^{HF^{-1}}) \propto \exp\left(-\frac{1}{2}\mathbf{x}^T A^{HF} \mathbf{x}\right). \quad (6)$$

The matrix A^{HF} was generated from Wishart distribution with identity scale matrix and 250 degrees of freedom. This yields a target distribution with strongly correlated covariance matrix ($\Sigma^{HF} = A^{HF^{-1}}$). The similar target distribution is used in previous studies ([Hoffman et al., 2014](#)).

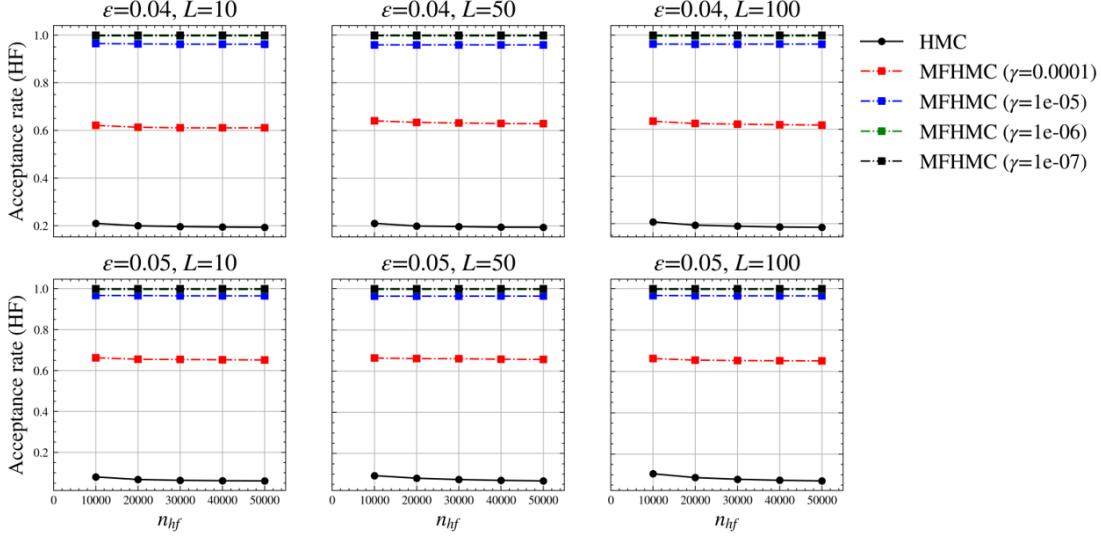


Figure 2: Comparison of acceptance rate of HMC and MFHMC at different computation budgets as defined by number of HF target distribution evaluations

We consider multiple LF target distributions to assess the effect of the fidelity of the LF model on the overall performance of MFHMC. These different LF distributions are parameterized by a scalar parameter γ as follows:

$$\Sigma^{LF} := \Sigma^{HF} + \left(\frac{\gamma}{d}\right) \times \text{trace}(\Sigma^{HF}) \times I(d).$$

where $d = 250$ is the dimension of \mathbf{x} and $I(d)$ denotes identity matrix of dimension 250. Thus, by changing the value of γ , we get different Σ^{LF} and as a consequence different $p^{\text{post(LF)}}(\mathbf{x}) := \mathcal{N}(\mathbf{0}, \Sigma^{LF} = A^{LF^{-1}})$.

Figure 2 and 3 show the relative comparison of HMC and MFHMC algorithm (with different LF models) for both computational and statistical efficiency while 4 and 5 show the comparison of both for accuracy.

3.2. Initial condition inversion

We first consider the problem of inferring the initial condition for the transient heat conduction problem given the noisy measurement of temperature at some later time. Here, by changing the value of γ we get different Σ^{LF} and as a result different $p^{\text{post(LF)}}(\mathbf{x}) :=$

$$\nabla \cdot (\boldsymbol{\alpha} \nabla \mathbf{u}) = \partial \mathbf{u} / \partial t \quad \text{in } \Omega \times (0, T] \quad (7)$$

$$\mathbf{u} = \mathbf{m} \quad \text{on } \Omega \times T_0 \quad (8)$$

$$\mathbf{u} = \mathbf{g} \quad \text{on } \partial\Omega \times T \quad (9)$$

where $\Omega \subset \mathbb{R}^2$ is a square domain with length $= 2\pi$ units and α is thermal diffusivity. \mathbf{u} is temperature at time t , and \mathbf{m} is initial condition for temperature. Here, the parameter to

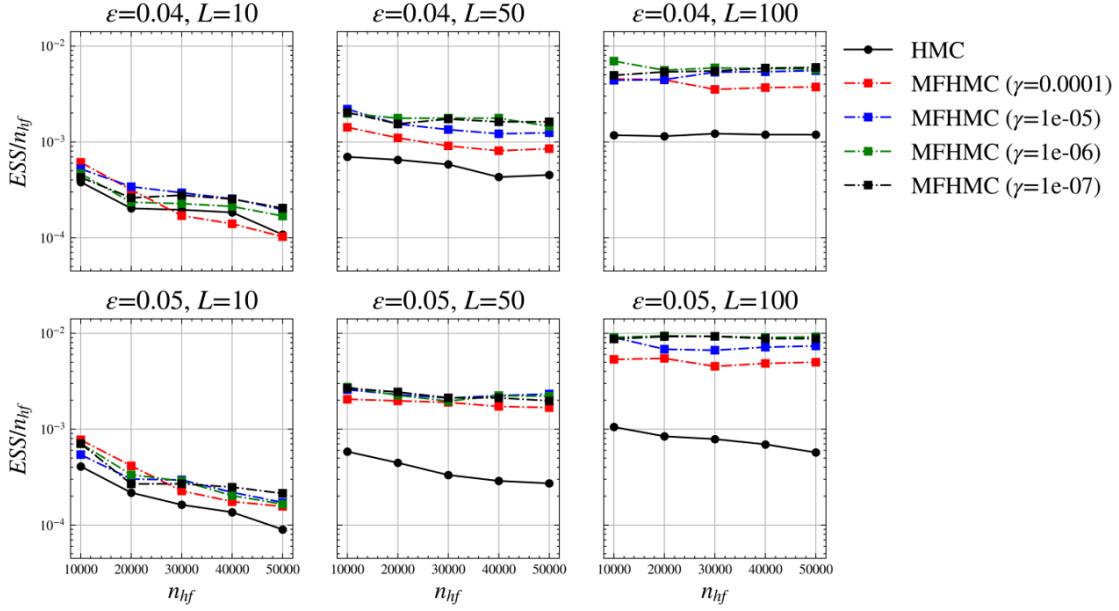


Figure 3: Comparison of effective sample size (ESS) per HF target evaluation of HMC and MFHMC at different computation budgets as defined by the number of HF target distribution evaluations

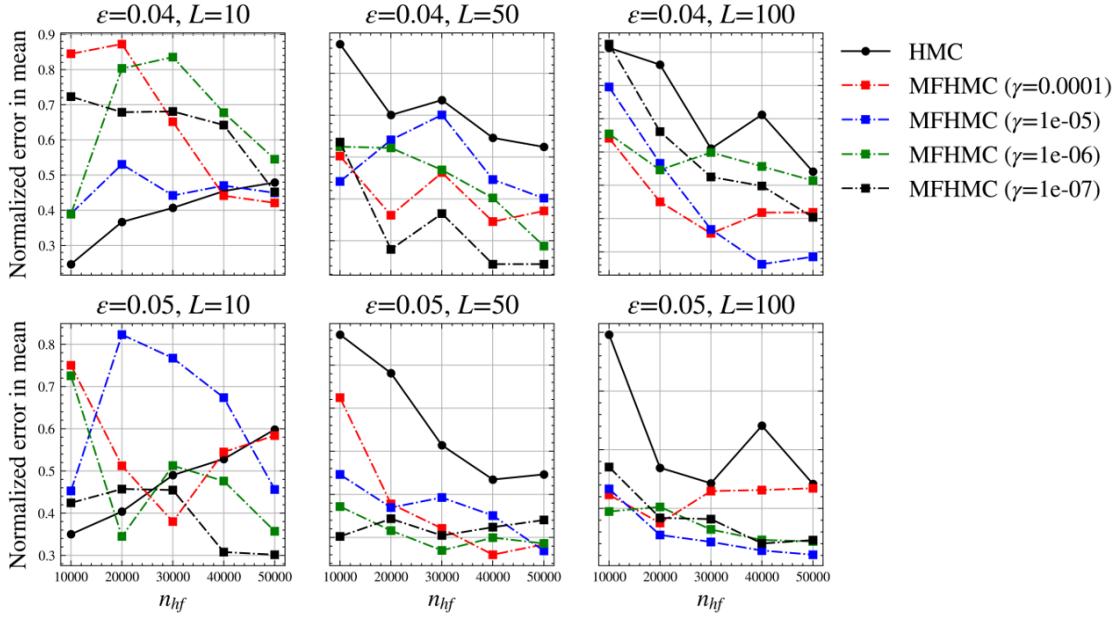


Figure 4: Comparison of normalized error in the mean of HMC and MFHMC at different computation budgets as defined by the number of HF target distribution evaluations

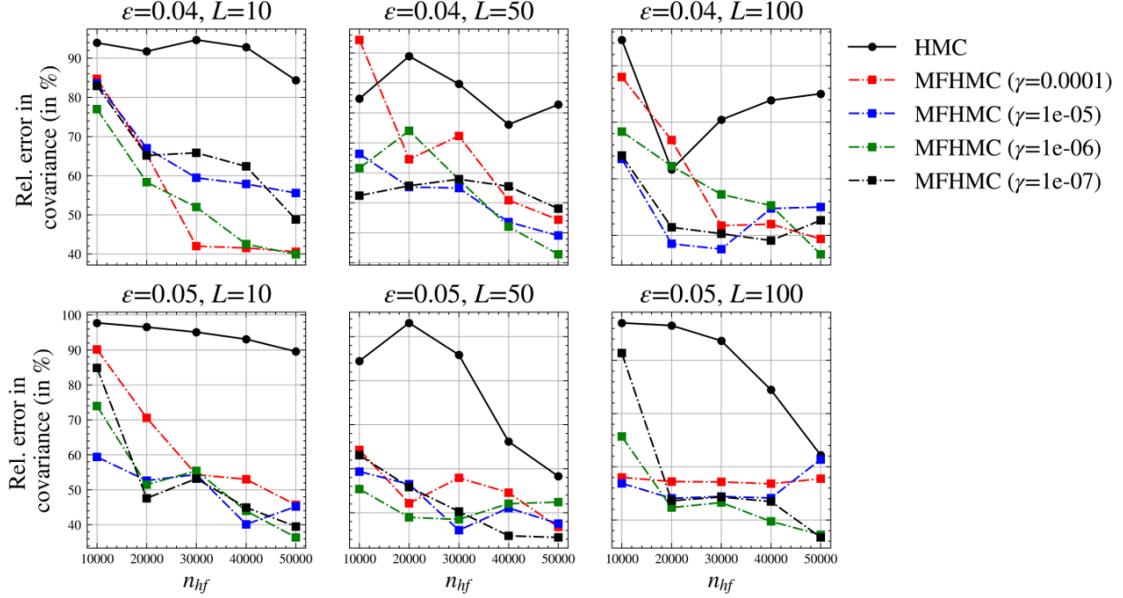


Figure 5: Comparison of relative error in covariance (in %) of HMC and MFHMC at different computation budgets as defined by the number of HF target distribution evaluations

infer \mathbf{x} is the nodal values of initial condition \mathbf{m} and the measurement \mathbf{y} is the nodal values of temperature \mathbf{u} .

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

We choose the second-order finite difference scheme in the spatial domain and backward Euler scheme in the temporal domain to solve (7) numerically. This leads to the following linear forward problem relating the inferred field (initial condition, $m(\mathbf{s})$) to the measured field (final temperature, $u(\mathbf{s}, T)$)

$$\mathbf{y} = \mathbf{F}\mathbf{x}, \quad (10)$$

where \mathbf{x} is a vector of nodal values of initial condition $m(\mathbf{s})$ and \mathbf{y} is a vector of nodal values of final temperature field $u(\mathbf{s})$ at time $t = T$. (For our numerical experiment, we discretize the spatial domain in 28 nodal points in each direction and the temporal domain in 100 time steps.)

In Bayesian inversion, we are interested in inferring the posterior density

$$p^{\text{post}}(\mathbf{x}|\hat{\mathbf{y}}) \propto p(\hat{\mathbf{y}}|\mathbf{x})p(\mathbf{x}).$$

We consider two different prior distributions: (i) Gaussian priors, and (ii) Generative Adversarial Network (GAN)-based priors.

3.2.1. Gaussian priors

In this section, we select a Gaussian distribution for prior and likelihood distribution with zero mean and constant variance. Specifically, we select $p(\mathbf{x}) = p(\hat{\mathbf{y}}|\mathbf{x}) = \mathcal{N}(\mathbf{0}, \sigma^2 \mathbf{I}_N)$ with $\sigma = 0.1$. While selecting Gaussian distribution for likelihood is relatively common, it is not common (and advisable) to select a Gaussian distribution for the prior. However, here we select a Gaussian for a conjugate prior case allowing an analytical solution for the resulting posterior distribution. This in turn will allow us to compare the posterior QoIs obtained by our proposed two-stage HMC algorithm with the “true” QoIs. We further use the single-stage HMC algorithm (which is the current gold standard in Bayesian inversion) as a benchmark to compare the computational efficiency and accuracy of our method. For this Gaussian prior case, we use a truncated singular value decomposition (TSVD) model as a surrogate (\mathbf{f}^*) forward model. We consider five different surrogate forward models based on five different numbers of retained modes in TSVD (latent space dimensionality): 25, 50, 75, 100, 200.

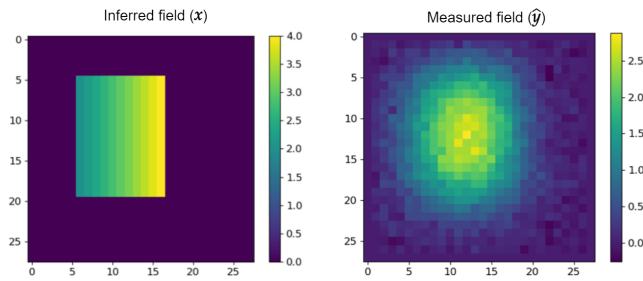


Figure 6: Measured and true inferred field.

Figure 6 (left panel) shows the “true” (discretized) initial condition field. Using this initial condition in (10), the final temperature at time $t = T$ is obtained and then uncorrelated Gaussian noise (with zero mean and 0.01 variance) is added to it to obtain the synthetic measured temperature field (right panel).

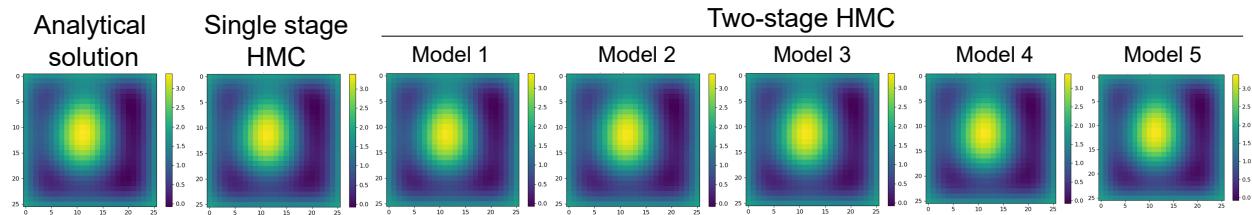


Figure 7: Qualitative comparison of the posterior mean for a Gaussian prior case. *First column:* analytical solution. *Second column:* posterior mean computed using the single-stage HMC. *3–7 column:* posterior mean computed using the proposed MFHMC algorithm with different surrogate models. Each of these models is based on truncated SVD with a different number of retained modes. Model 1: 25 modes, Model 2: 50 modes, Model 3: 75 modes, Model 4: 100 modes, Model 5: 200 modes.

Next, we use the measured temperature field $\hat{\mathbf{y}}$ (shown in the right panel of Figure 6)

to obtain the posterior mean. We first compute the analytical solution for the mean. This is shown in the first column of Figure 7. Next, we compute the posterior mean using the samples produced by the single-stage HMC algorithm (described in Section 2). The results for this case are shown in the second column of Figure 7. Finally, we compute the posterior mean using the multi-fidelity HMC algorithm. We report results for all five surrogate models (model 1: 25 modes, model 2: 50 modes, model 3: 75 modes, model 4: 100 modes, model 5: 200 modes). As can be observed from the last five columns of Figure 7, the computed mean with the proposed method is in good qualitative agreement with the analytical solution as well as the solution obtained from the benchmark. Both single-stage and two-stage HMC algorithms were run for 20,000 MCMC steps and the step size and the number of leapfrog steps were tuned to obtain the target acceptance ratio in the range of 0.6–0.7 following the recommendations of (Beskos et al., 2013).

Next, we show a quantitative comparison of the proposed algorithm with the single-stage HMC algorithm in Table 1. We can observe the following points from Table 1:

- Significant improvement in the acceptance ratio can be achieved with the two-stage algorithm. This translates into a smaller number of *wasted* HF simulations. This is possible because most of the “bad” samples are filtered out by the first stage (without any need for expensive an HF simulation and thus achieving considerable computational saving). And most of the samples passed to the second stage by the surrogate model are already of high quality and so they are retained by the second stage with high probability. This is not the case with the single-stage HMC algorithm, where there is no such filtering mechanism and hence all samples are evaluated by the HF model.
- The number of HF evaluations required by the single-stage algorithm is very high. This is due to the fact that each MCMC step requires solution of the forward and the adjoint problem (i.e., two HF evaluations). Whereas, in our proposed algorithm, only a single HF evaluation (only forward, with no adjoint) is required for samples accepted in the first stage and no HF evaluation is required for samples rejected in the first stage.
- As the fidelity of the low fidelity (LF) model improves, the acceptance rate of the HF model improves dramatically. This is not surprising since as the fidelity of the LF

Table 1: Quantitative comparison of computed QoIs (for Gaussian prior with TSVD-based surrogate model)

Model	Modes	Acceptance rate (LF)	Acceptance rate (HF)	No. of HF evaluations	No. of rejected HF evaluations	Error in mean (in %)	Error in Cov. (in %)
Model 1	25	0.59	0.76	11845	2885	4.03	262.39
Model 2	50	0.58	0.98	11664	213	3.47	228.34
Model 3	75	0.59	0.99	11779	33	3.17	225.36
Model 4	100	0.59	0.99	11720	5	3.13	223.73
Model 5	200	0.59	1.0	11775	0	3.31	217.45
Model 2a	50	0.02	0.98	445	5	3.65	377.09
Model 5a	200	0.02	1.0	447	0	3.33	343.92
1 stage HMC	676 (all)	—	0.59	40000	16350	3.21	223.85

model improves, it becomes more and more similar to the HF model and hence the majority of the samples accepted by the LF model are eventually accepted by the HF model boosting its acceptance rate and reducing *wasted* HF evaluations.

- The error of our proposed method is similar (or better) to the single-stage HMC algorithm. Furthermore, the accuracy can further be increased by using a high-quality LF model.
- In order to test the accuracy-efficiency trade-off and its potential gains, we did two additional MCMC runs (with Model 2 and Model 5), where we tuned the HMC hyperparameters to achieve a very low acceptance rate for the LF model. This translates to orders of magnitude fewer number of HF evaluations. Even with such fewer HF evaluations, our proposed two-stage algorithm is able to achieve accuracy that is comparable to a single-stage HMC, which requires orders of magnitude more HF evaluations.

3.2.2. GAN-based priors

While the Gaussian prior is useful for verification purposes, it is not necessarily the best prior for good inference (as can be observed by comparing the true inferred field in the left panel of Figure 6 with the inferred means of Figure 7). In recent times, various deep generative models have shown tremendous promise as prior for accurate inference in Bayesian inversion. A particularly promising model among them is the Generative Adversarial Network (GAN) (Goodfellow et al., 2014). The most appealing features of this model are (i) its ability to map high-dimensional data into a low-dimensional latent space, and (ii) the ability to learn and model complex probability distributions. By leveraging these two features, in recent years GANs have shown impressive results as priors in various domains such as geoscience, physics, and computer vision (Mosser et al., 2019; Patel and Oberai, 2021).

In this section, we consider using GAN-based priors for the initial condition inversion problem. We consider two different surrogate forward models: (i) TSVD with 100 modes and (ii) a deep neural network that maps the initial condition $m(\mathbf{s})$ to the final temperature field $u(\mathbf{s}, T)$.

TSVD-based surrogate. Figure 8 shows results for Bayesian inversion with GAN-based prior and TSVD-based surrogate for $5 \cdot 10^4$ MCMC steps. As a benchmark comparison, we again consider a single-stage HMC algorithm with HF forward model as described in (10) simulated for $5 \cdot 10^4$ steps. As can be observed from Figure 8, the obtained posterior mean and variance are similar for both algorithms, highlighting the ability of a two-stage algorithm to reliably estimate posterior statistics. However, the number of HF evaluations required for single-stage HMC (10^5) are significantly higher than the proposed two-stage algorithm ($3.85 \cdot 10^4$). Thus, with the two-stage strategy, the computational saving is more than 60% while retaining the same accuracy. We note that the variance estimation obtained using GAN-based priors is consistent with the variance estimation obtained using different methods used in our previous research for sharp boundary object inference (Lee and Kitanidis, 2013a; Qian et al., 2020)

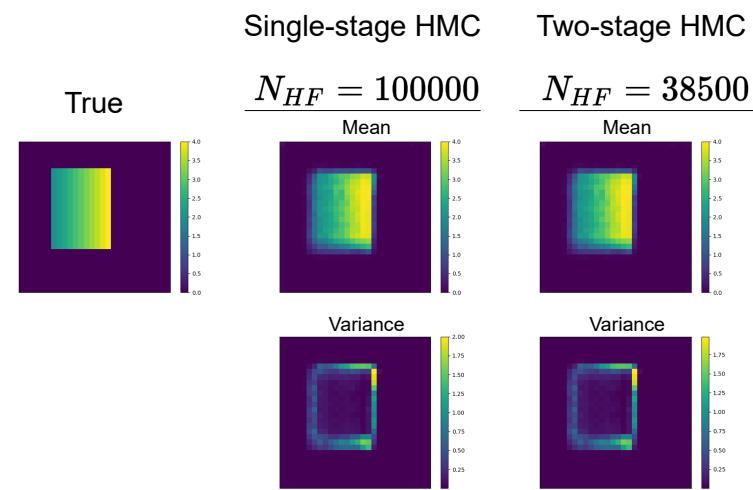


Figure 8: Comparison of posterior QoIs for a GAN-based prior case. *First column:* true parameter field (initial condition). *Second column:* posterior mean and variance using single-stage HMC. *Third column:* posterior mean and variance using proposed MFHMC algorithm with TSVD-based LF model.

DNN-based surrogate. Till now we have considered the TSVD-based surrogate forward models for which the gradient (in the first stage of the MFHMC algorithm) can be computed analytically. While this is an optimal surrogate model showing excellent results for the linear initial condition inversion problem considered here, for more complex and non-linear inverse problems, a TSVD-based surrogate model might not be sufficient due to its linear nature. In recent years, DNN has emerged as a powerful surrogate for highly non-linear and complex forward models (Zhu et al., 2019). Moreover, the automatic differentiation capabilities of modern machine learning libraries enable gradient calculations with a high degree of accuracy at *almost no additional cost*. This feature is particularly useful in the first stage of the MFHMC algorithm. We test the effectiveness of such a DNN-based surrogate forward model in our proposed algorithm for the initial condition inversion problem described before.

For this, we first train a 4-layer deep fully connected neural network surrogate (\mathbf{f}_{DNN}) mapping the vector of the initial condition field (\mathbf{x}) to the vector of the final temperature field (\mathbf{y}) in an *offline stage*. This DNN was trained by minimizing the mean square error loss using the Adam optimizer (Kingma and Ba, 2014) over 10^4 pair-wise training data (of the initial condition and final temperature generated using HF numerical solver). Figure 9 shows the final temperature prediction of the trained DNN for four initial conditions sampled from the validation set. The relative error (measured in L_2 sense) of the trained DNN on the validation set is 0.7%.

Once the DNN is trained in an offline stage, it is then used as a surrogate forward model in the likelihood term of the first stage of the MFHMC method and the gradient of the resulting posterior is computed using automatic differentiation. In the second stage of the MFHMC algorithm, we use the HF numerical solver as before. The resulting posterior QoIs for this case are shown in Figure 10. As before the computed mean and variance from the

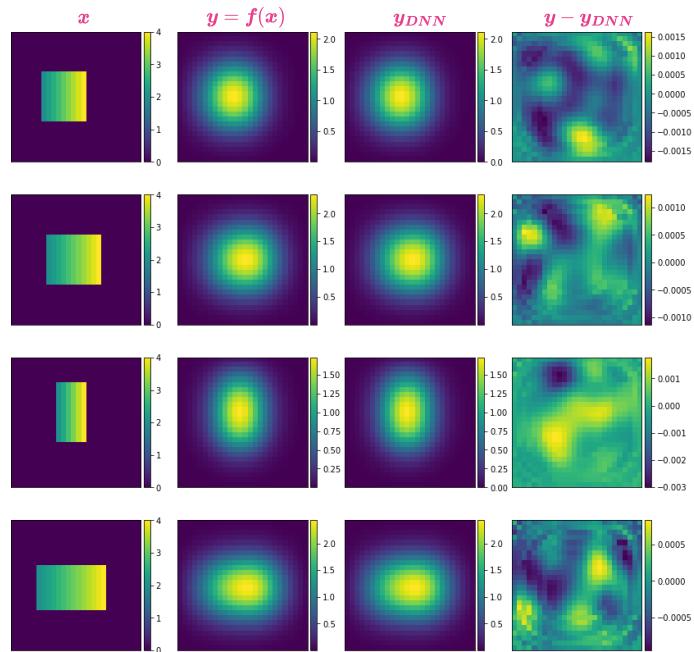


Figure 9: DNN-based surrogate model prediction for four different validation set samples. *First column:* initial condition samples from validation set. *Second column:* true final temperature field obtained using HF numerical solver. *Third column:* final temperature field prediction by trained DNN-based surrogate. *Fourth column:* difference between true and predicted temperature field.

posterior samples of single-stage and two-stage HMC are similar. However, the number of HF evaluations required for single-stage HMC (10^5) are significantly higher than the two-stage HMC (49.7×10^4) method resulting in huge computational saving. We note that our proposed two-stage algorithm requires training of the surrogate model using training data obtained from the HF numerical simulations and that adds up additional 10^4 HF evaluations to our computational cost. However, this is a one-time cost we have to pay in an offline stage, which could be amortized over multiple online inferences.

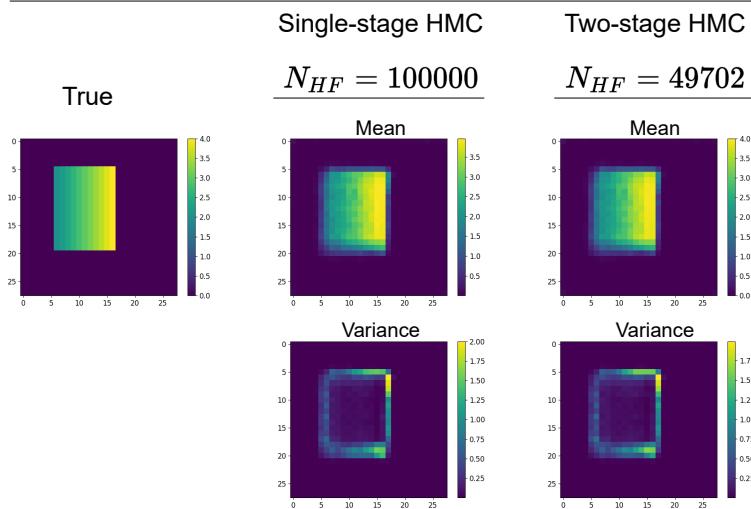


Figure 10: Comparison of the posterior QoIs with a GAN-based prior. *First column:* true parameter field (initial condition). *Second column:* posterior mean and variance using a single-stage HMC. *Third column:* posterior mean and variance using the proposed MFHMC algorithm with a DNN-based LF model.

3.3. Darcy's flow: coefficient inversion

This is a non-linear coefficient inversion problem for an elliptic PDE which arises in many fields such as subsurface flow modeling (Iglesias et al., 2013), electrical impedance tomography (Kaipio and Somersalo, 2006), and inverse heat conduction (Kaipio and Fox, 2011). Here, the goal is to infer the coefficient of the PDE (permeability/thermal conductivity field) given noisy measurement of pressure/temperature field. For this problem, the forward model is described as:

$$\begin{aligned} -\nabla \cdot (\kappa(\mathbf{s}) \nabla u(\mathbf{s})) &= b(\mathbf{s}), & \mathbf{s} = (s_1, s_2) \in \Omega \\ u(\mathbf{s}) &= 0, & \mathbf{s} = (s_1, s_2) \in \partial\Omega \end{aligned} \quad (11)$$

where $\Omega \subset \mathbb{R}^2$ is a square domain with length = 1 unit, and $b(\mathbf{s}) = 10^3$ denotes the heat source. The goal is to infer the posterior QoIs of permeability/conductivity field κ , given a noisy, and potentially partial, measurement of pressure/temperature field u . The nodal values of the pressure/temperature field are stored in the vector \mathbf{y} and those of the permeability/conductivity field are stored in the vector \mathbf{x} .

For this experiment, we consider two different datasets: parametric (rectangular) dataset and the channelized flow dataset. Here, the first dataset corresponds to the inverse heat conduction problem, whereas the second dataset corresponds to the permeability inversion problem commonly encountered in geophysics. We further consider two separate scenarios for each dataset to evaluate their performance. For the rectangular dataset (inverse heat conduction problem), we consider the scenario of fixed statistical error in posterior statistics and compare the performance of MFHMC with HMC, whereas for the channelized flow (permeability inversion problem), we consider the scenario of fixed computational budget of HF evaluations. We consider GAN-based priors for both the datasets.

3.3.1. Inverse heat conduction

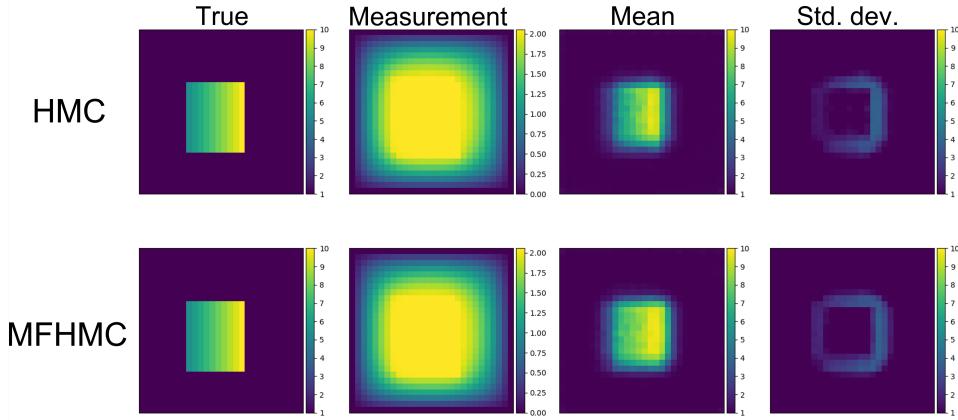


Figure 11: Comparison of the posterior QoIs for the proposed MFHMC algorithm (*second row*) with the single stage HMC algorithm (*first row*) for the fixed statistical error scenario.

For this experiment, we use the parametric rectangular dataset considered in the previous section. We use the GAN-based priors and DNN-based forward model surrogate. We consider the scenario of fixed statistical error in the posterior statistics to compare the relative performance of the HMC and MFHMC algorithms. Figure 11 shows the posterior statistics for both these algorithms. As can be observed from these figures, for both algorithms the posterior statistics are more or less the same with standard deviation plot captures the regions of uncertainty.

As discussed in section 1, one of the advantages of the MFHMC algorithm over the traditional HMC algorithm is the ability to tune hyperparameters (such as step size and the number of leapfrog steps) to achieve high statistical efficiency. We demonstrate this feature in this experiment. Specifically, we choose the trajectory length (which is simply the product of the step size and the number of leapfrog steps) to be three times that of the single-stage HMC algorithm. This enables taking longer jumps in the parameter space and as a result,

Table 2: Comparison of HMC and MFHMC algorithm for the rectangular dataset

Quantity of Interest	HMC	MFHMC
No. of HF evaluations ↓	13450	4797
ESS (per sample) ↑	$\mathcal{O}(1e-5)$	$\mathcal{O}(1e-4)$
MSJD (per sample) ↑	$\mathcal{O}(10)$	$\mathcal{O}(100)$

produces more uncorrelated samples. This can be observed from table 2, where MFHMC has an order of magnitude higher effective sample size and the mean squared jump distance.

3.3.2. Permeability inversion

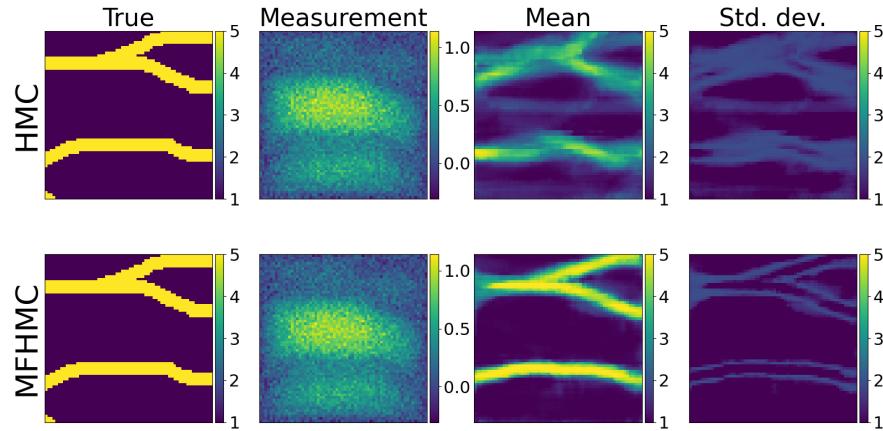


Figure 12: Comparison of the posterior QoIs for the proposed MFHMC algorithm (*second row*) with the single stage HMC algorithm (*first row*) for the fixed number of HF simulations.

For this experiment, we use a channelized flow dataset and consider GAN-based priors. We further consider the scenario where we have a fixed computational budget, where we can only use a fixed number of HF evaluations. This is set to 10000 HF simulations. For this scenario, we ran both single-stage HMC and MFHMC algorithms and computed posterior statistics such as mean and standard deviation. Figure 12 shows these posterior statistics. As can be observed from these figures, as compared to HMC, the mean of the proposed algorithm is much closer to the ground truth and the standard deviation plot captures the regions of uncertainty.

In Table 3, we provide a quantitative comparison of the two algorithms. We consider the total computation time and the error in the posterior mean as our evaluation metric. For error computation, we use the true value of the permeability field as our reference value.

Table 3: Comparison of HMC and MFHMC algorithm for the channelized flow dataset

Quantity of Interest	HMC	MFHMC
Error in mean (in %) ↓	51.4	31.4
Time (in Sec.) ↓	11777.5	3401.5

3.4. Hydraulic Tomography

In this section, we demonstrate the effectiveness of our proposed algorithm to a real-world experimental dataset. Specifically, we consider the laboratory-scale hydraulic tomography experiment. The experiments were conducted at the University of Iowa by Walter Illman and colleagues, and the same set of data have been used previously in various studies ([Liu and Kitanidis, 2011](#)).

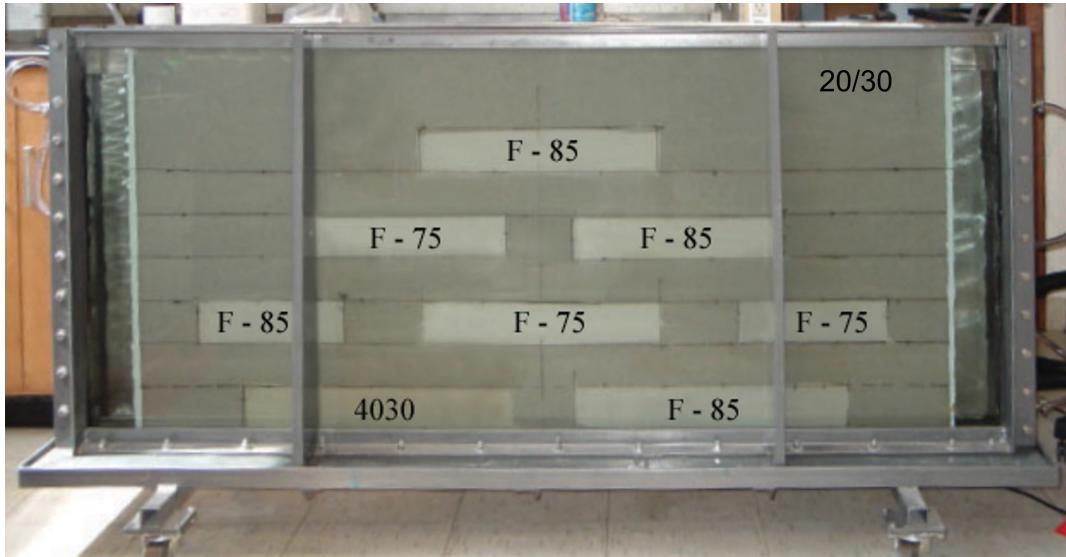


Figure 13: Front view of the Sandbox. Figure reused from ([Liu and Kitanidis, 2011](#))

Figure 13 shows the front view of the sandbox used in this experiment. Figure 14 shows the schematic of the sandbox. As shown in this figure the sandbox is 161 cm long and 81 cm high. Four different kinds of commercially available sands (denoted by F-85, F-75, 4030, and 20/30 in fig. 14) were used to construct the sandbox. The box is composed of finer sand and the background is made of the coarser sand. On the back of the sandbox 48 pressure sensors were installed at locations indicated by solid dots with numbers in fig. 14 to measure the hydraulic head. Nine different experiments were conducted for hydraulic survey at the red circled and squared ports. Data from all eight experiments were used for inference.

The governing problem for this hydraulic survey is given by

$$\begin{aligned}
 -\nabla \cdot (\kappa(\mathbf{s}) \nabla \mathbf{u}(\mathbf{s})) &= \mathbf{f}_i \delta(\mathbf{s} - \mathbf{a}), & \mathbf{s} = (s_1, s_2) \in \Omega \\
 \mathbf{u}(\mathbf{s}) &= \mathbf{g}_i, & \mathbf{s} = (s_1, s_2) \in \partial\Omega_g \\
 \kappa \nabla \mathbf{u}(\mathbf{s}) &= \mathbf{h}_i, & \mathbf{s} = (s_1, s_2) \in \partial\Omega_h
 \end{aligned} \tag{12}$$

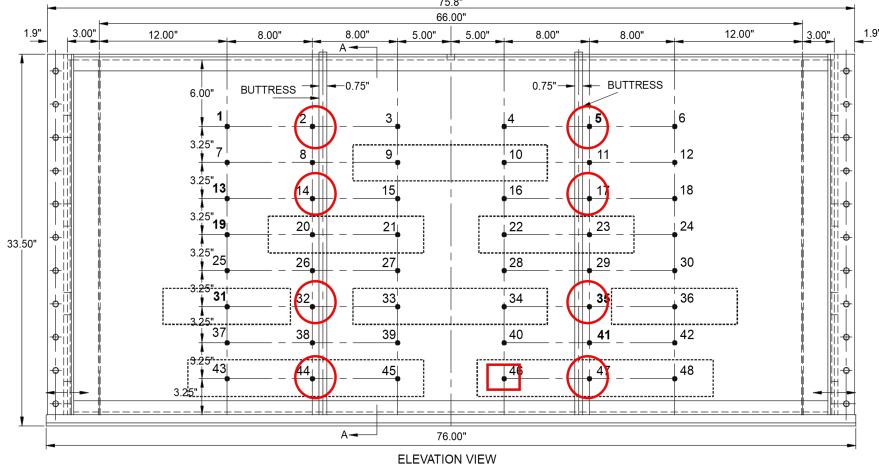


Figure 14: Schematic of the Sandbox with the dimension of various sand blocks and the sensor locations. The numbered nodes (1-48) indicates the location of pressure sensor. The red circle indicates the location of source sensor. Figure reused from ([Liu and Kitanidis, 2011](#))

where κ , \mathbf{u} , and \mathbf{a} indicate hydraulic conductivity, hydraulic head, and the location of circled ports respectively. \mathbf{f}_i , \mathbf{g}_i , and \mathbf{h}_i indicate source, Dirichlet boundary condition, and the flux boundary condition for i^{th} experiment with $i = \{1, 2, \dots, 9\}$.

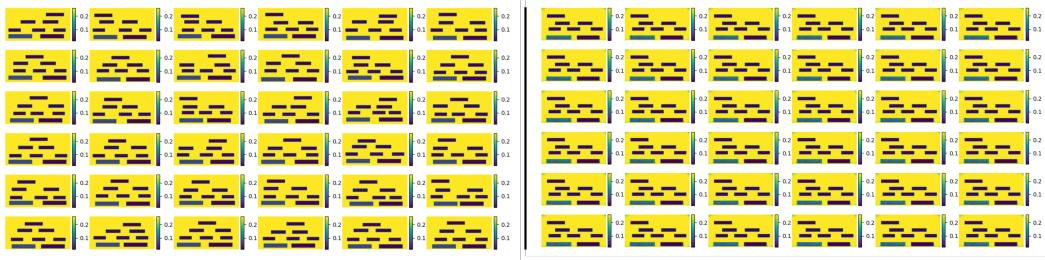


Figure 15: Thirty six realizations of the hydraulic conductivity from the training set used to train the GAN-prior (left panel) and corresponding realizations generated by the trained GAN-prior (right panel).

In order to do the inference, we use the GAN-based priors. This prior was trained by varying the horizontal and vertical location of the eight sand blocks. Figure 15 shows the realizations of hydraulic tomography from the training set used to train the GAN-prior and corresponding realizations from the trained GAN. For LF model, a CNN-based surrogate model was used to map the hydraulic conductivity image to 432 hydraulic head measurements corresponding the 48 head measurements for 9 experiments. As an HF model FEM model was used in FEniCS.

Figure 16 shows the mean and the standard deviation of the posterior distribution inferred using the proposed two-stage MFHMC algorithm. As can be observed the mean is able to capture the location of all the box quite well except the top sand box. It is also able to accurately infer the value of hydraulic conductivity for all the sandboxes except the top one. We hypothesize the difficulty in capturing the exact value of hydraulic conductivity for

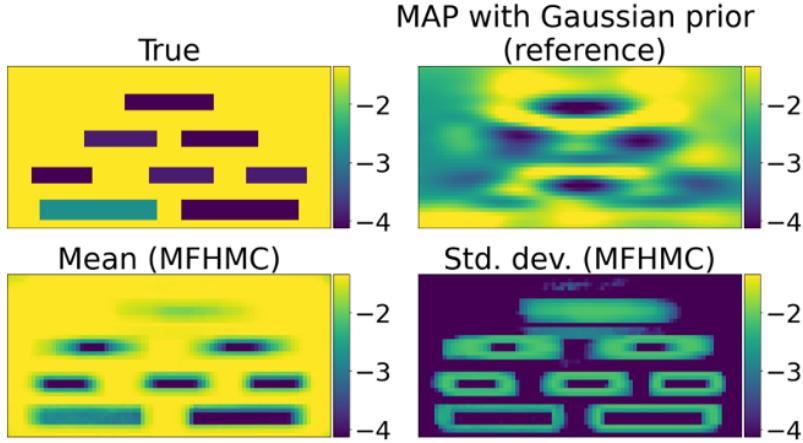


Figure 16: True hydraulic conductivity field from the experiment (top left panel). Mean of the posterior distribution inferred using MFHMC algorithm (bottom left panel). Standard deviation of the posterior distribution inferred using MFHMC algorithm (bottom right panel). Benchmark mean field of the posterior distribution inferred using Gaussian-based prior (top right panel).

this box is probably due to the noisy measurement from the top sensors. The bottom right panel in fig. 16 shows the standard deviation of the posterior distribution obtained using MFHMC algorithm. This standard deviation field, which is our measure of uncertainty, is elevated along the edges of the sandboxes. This is expected as perturbing the location of the sandbox by slightly will not have any significant effect on measurements and hence high uncertainty. As a reference, we also perform inference using traditional GAN-based priors with geostatistical inversion method. Sharper reconstruction results of our proposed method highlights its relative effectiveness in dealing with noisy experimental data.

4. Conclusion

HMC is a powerful method for generating samples from an unnormalized probability distribution. By exploiting the geometry of the target density, it can achieve a faster convergence rate than traditional MCMC methods and scale to high-dimensional parameter spaces. However, running an HMC algorithm is a computationally demanding task, especially for the problems governed by PDEs, as running each step of HMC requires solving two PDEs. Furthermore, it is not possible to use the HMC algorithm for applications where computing the gradient is impossible. Such scenarios arise when dealing with black-box simulators of physical processes or applications with non-differentiable priors. In this manuscript, we have proposed a novel multi-fidelity Hamiltonian Monte Carlo algorithm, which is a “gradient-free” HMC algorithm with respect to the HF numerical solver. We have demonstrated the effectiveness and applicability of the proposed algorithm on a range of linear and non-linear inverse problems. In all numerical experiments, it has outperformed traditional HMC algorithm in terms of statistical accuracy and computational as well as statistical efficiency. Moreover, we have also evaluated its performance using experimental data in the context of hydraulic tomography.

References

- J. Andrés Christen and Colin Fox. Markov chain Monte Carlo Using an Approximation. <http://dx.doi.org/10.1198/106186005X76983>, 14(4):795–810, dec 2012. ISSN 10618600. doi: 10.1198/106186005X76983. URL <https://www.tandfonline.com/doi/abs/10.1198/106186005X76983>.
- Christophe Andrieu, Nando de Freitas, Arnaud Doucet, and Michael I. Jordan. An Introduction to MCMC for Machine Learning. *Machine Learning* 2003 50:1, 50(1):5–43, jan 2003. ISSN 1573-0565. doi: 10.1023/A:1020281327116. URL <https://link.springer.com/article/10.1023/A:1020281327116>.
- A. Beck and M. Teboulle. Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems. *IEEE Transactions on Image Processing*, 18: 2419–2434, 2009.
- Alexandros Beskos, Natesh Pillai, Gareth Roberts, Jesus Maria Sanz-Serna, and Andrew Stuart. Optimal tuning of the hybrid Monte Carlo algorithm. *Bernoulli*, 19(5 A):1501–1534, 2013. ISSN 13507265. doi: 10.3150/12-BEJ414. URL <https://projecteuclid.org/euclid.bj/1383661192>.
- Michael Betancourt, Simon Byrne, Sam Livingstone, and Mark Girolami. The geometric foundations of Hamiltonian Monte Carlo. <https://doi.org/10.3150/16-BEJ810>, 23(4A):2257–2298, nov 2017. ISSN 1350-7265. doi: 10.3150/16-BEJ810. URL <https://projecteuclid.org/journals/bernonulli/volume-23/issue-4A/The-geometric-foundations-of-Hamiltonian-Monte-Carlo/10.3150/16-BEJ810.full>.
- W. Boomsma, J. Frellsen, T. Harder, Sandro Bottaro, K. E. Johansson, Pengfei Tian, K. Stovgaard, Christian Andreetta, S. Olsson, Jan B. Valentin, L. D. Antonov, Anders S. Christensen, M. Borg, Jan H. Jensen, K. Lindorff-Larsen, J. Ferkinghoff-Borg, and T. Hamelryck. Phaistos: A framework for markov chain monte carlo simulation and inference of protein structure. *Journal of Computational Chemistry*, 34:1697 – 1705, 2013.
- Steve Brooks, Andrew Gelman, Galin L. Jones, and Xiao Li Meng. *Handbook of Markov Chain Monte Carlo*. CRC Press, may 2011. ISBN 9781420079425. doi: 10.1201/b10905.
- Lotfi Chaari, J. Tourneret, and H. Batatia. Sparse bayesian regularization using bernoulli-laplacian priors. *21st European Signal Processing Conference (EUSIPCO 2013)*, pages 1–5, 2013.
- Antonin Chambolle, Vicent Caselles, Daniel Cremers, Matteo Novaga, and Thomas Pock. *An Introduction to Total Variation for Image Analysis*:, pages 263–340. De Gruyter, 2010. doi: doi:10.1515/9783110226157.263. URL <https://doi.org/10.1515/9783110226157.263>.

Masoumeh Dashti and Andrew M. Stuart. The Bayesian Approach to Inverse Problems. In *Handbook of Uncertainty Quantification*, pages 311–428. Springer, Cham, jun 2017. doi: 10.1007/978-3-319-12385-1_7. URL https://link.springer.com/referenceworkentry/10.1007/978-3-319-12385-1_7.

Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real nvp. *ArXiv*, abs/1605.08803, 2017.

Simon Duane, A.D. Kennedy, Brian J. Pendleton, and Duncan Roweth. Hybrid monte carlo. *Physics Letters B*, 195(2):216–222, 1987. ISSN 0370-2693. doi: [https://doi.org/10.1016/0370-2693\(87\)91197-X](https://doi.org/10.1016/0370-2693(87)91197-X).

Y. Efendiev, T. Hou, and W. Luo. Preconditioning Markov Chain Monte Carlo Simulations Using Coarse-Scale Models. <http://dx.doi.org/10.1137/050628568>, 28(2):776–803, jul 2006. ISSN 10648275. doi: 10.1137/050628568.

Sina Farsiu, D. Robinson, Michael Elad, and P. Milanfar. Fast and robust multi-frame super-resolution. *IEEE Transactions on Image Processing*, 1996.

Alan E. Gelfand and Adrian F.M. Smith. Sampling-based approaches to calculating marginal densities. *Journal of the American Statistical Association*, 85(410):398–409, 1990. ISSN 1537274X. doi: 10.1080/01621459.1990.10476213.

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.

Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Networks. jun 2014. URL <http://arxiv.org/abs/1406.2661>.

Michael Habeck, Michael Nilges, and Wolfgang Rieping. Replica-exchange monte carlo scheme for bayesian data analysis. *Physical review letters*, 94(1):018105, 2005.

Matthew D Hoffman, Andrew Gelman, et al. The no-u-turn sampler: adaptively setting path lengths in hamiltonian monte carlo. *J. Mach. Learn. Res.*, 15(1):1593–1623, 2014.

M Iglesias, K Law, and A Stuart. Evaluation of Gaussian approximations for data assimilation in reservoir models. *Computational Geosciences*, 17:851–885, 2013.

Fernández-Durán JJ. Circular distributions based on nonnegative trigonometric sums. *Biometrics*, 60(2):499–503, jun 2004. ISSN 0006-341X. doi: 10.1111/J.0006-341X.2004.00195.X. URL <https://pubmed.ncbi.nlm.nih.gov/15180676/>.

Jari Kaipio and Erkki Somersalo. *Statistical and computational inverse problems*, volume 160. Springer Science & Business Media, 2006.

- Jari P Kaipio and Colin Fox. The Bayesian Framework for Inverse Problems in Heat Transfer. *Heat Transfer Engineering*, 32(9):718–753, 2011. doi: 10.1080/01457632.2011.525137. URL <https://doi.org/10.1080/01457632.2011.525137>.
- Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. dec 2014. URL <http://arxiv.org/abs/1412.6980>.
- J. Lee and P. K. Kitanidis. Bayesian inversion with total variation prior for discrete geologic structure identification. *Water Resources Research*, 49: 7658–7669, 11 2013a. ISSN 1944-7973. doi: 10.1002/2012WR013431. URL <https://onlinelibrary.wiley.com/doi/full/10.1002/2012WR013431><https://onlinelibrary.wiley.com/doi/abs/10.1002/2012WR013431><https://agupubs.onlinelibrary.wiley.com/doi/10.1002/2012WR013431>.
- J Lee and PK Kitanidis. Bayesian inversion with total variation prior for discrete geologic structure identification. *Water Resources Research*, 49(11):7658–7669, 2013b.
- X. Liu and P. K. Kitanidis. Large-scale inverse modeling with an application in hydraulic tomography. *Water Resources Research*, 47(2):2501, feb 2011. ISSN 1944-7973. doi: 10.1029/2010WR009144. URL <https://onlinelibrary-wiley-com.stanford.idm.oclc.org/doi/full/10.1029/2010WR009144><https://onlinelibrary-wiley-com.stanford.idm.oclc.org/doi/abs/10.1029/2010WR009144><https://agupubs-onlinelibrary-wiley-com.stanford.idm.oclc.org/doi/10.1029/2010WR009144>.
- James Martin, Lucas C. Wilcox, Carsten Burstedde, and Omar Ghattas. A Stochastic Newton MCMC Method for Large-Scale Statistical Inverse Problems with Application to Seismic Inversion. *SIAM Journal on Scientific Computing*, 34(3):A1460–A1487, jan 2012. ISSN 1064-8275. doi: 10.1137/110845598. URL <http://pubs.siam.org/doi/10.1137/110845598>.
- Nicholas Metropolis, Arianna W. Rosenbluth, Marshall N. Rosenbluth, Augusta H. Teller, and Edward Teller. Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, 21:1087, 12 1953. ISSN 0021-9606. doi: 10.1063/1.1699114. URL <https://aip.scitation.org/doi/abs/10.1063/1.1699114>.
- Lukas Mosser, Olivier Dubrule, and Martin J. Blunt. Stochastic Seismic Waveform Inversion Using Generative Adversarial Networks as a Geological Prior. *Mathematical Geosciences 2019* 52:1, 52(1):53–79, nov 2019. ISSN 1874-8953. doi: 10.1007/S11004-019-09832-6. URL <https://link.springer.com/article/10.1007/s11004-019-09832-6>.
- Radford M. Neal. MCMC using Hamiltonian dynamics. *Handbook of Markov Chain Monte Carlo*, pages 1–592, jun 2012. doi: 10.1201/b10905. URL <http://arxiv.org/abs/1206.1901><http://dx.doi.org/10.1201/b10905>.

Erik Nijkamp, Mitch Hill, Tian Han, Song Chun Zhu, and Ying Nian Wu. On the Anatomy of MCMC-Based Maximum Likelihood Learning of Energy-Based Models. *AAAI 2020 - 34th AAAI Conference on Artificial Intelligence*, pages 5272–5280, mar 2019. ISSN 2159-5399. doi: 10.1609/aaai.v34i04.5973. URL <https://arxiv.org/abs/1903.12370v4>.

Trevor Park and George Casella. The Bayesian Lasso. <https://doi.org/10.1198/016214508000000337>, 103(482):681–686, jun 2012. doi: 10.1198/016214508000000337. URL <https://www.tandfonline.com/doi/abs/10.1198/016214508000000337>.

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, jan 2021. doi: 10.1137/20M1354210.

Dhruv V. Patel, Deep Ray, and Assad A. Oberai. Solution of physics-based bayesian inverse problems with deep generative priors. *ArXiv*, abs/2107.02926, 2021.

Yizhou Qian, Mojtaba Forghani, Jonghyun Harry Lee, Matthew Farthing, Tyler Hesser, Peter Kitanidis, and Eric Darve. Application of deep learning-based interpolation methods to nearshore bathymetry, 2020.

Leonid I. Rudin, Stanley Osher, and Emad Fatemi. Nonlinear total variation based noise removal algorithms. *Physica D: Nonlinear Phenomena*, 60(1-4):259–268, nov 1992. ISSN 0167-2789. doi: 10.1016/0167-2789(92)90242-F.

M Vauhkonen, J P Kaipio, E Somersalo, and P A Karjalainen. Electrical impedance tomography with basis constraints. *Inverse Problems*, 13(2):523–530, apr 1997. ISSN 0266-5611. doi: 10.1088/0266-5611/13/2/020. URL <http://stacks.iop.org/0266-5611/13/i=2/a=020?key=crossref.46559bf45aab26a8302acc14e8db4c89>.

Yinhao Zhu, Nicholas Zabaras, Phaedon Stelios Koutsourelakis, and Paris Perdikaris. Physics-constrained deep learning for high-dimensional surrogate modeling and uncertainty quantification without labeled data. *Journal of Computational Physics*, 394:56–81, oct 2019. ISSN 0021-9991. doi: 10.1016/J.JCP.2019.05.024.