University of Exeter
College of Engineering, Mathematics, and Physical Sciences

Multilevel Delayed Acceptance MCMC with Applications to Hydrogeological Inverse Problems

Mikkel Bue Lykkegaard

Submitted by Mikkel Bue Lykkegaard, to the University of Exeter as a thesis for the degree of Doctor of Philosophy in Water Informatics Engineering, March, 2022.

I certify that all material in this thesis which is not my own work has been identified and that any material that has previously been submitted and approved for the award of a degree by this or any other University has been acknowledged.

Signed: Mikkel Bue

Abstract

Quantifying the uncertainty of model predictions is a critical task for engineering decision support systems. This is a particularly challenging effort in the context of statistical inverse problems, where the model parameters are unknown or poorly constrained, and where the data is often scarce. Many such problems emerge in the fields of hydrology and hydro–environmental engineering in general, and in hydrogeology in particular. While methods for rigorously quantifying the uncertainty of such problems exist, they are often prohibitively computationally expensive, particularly when the forward model is high–dimensional and expensive to evaluate. In this thesis, I present a Metropolis–Hastings algorithm, namely the Multilevel Delayed Acceptance (MLDA) algorithm, which exploits a hierarchy of forward models of increasing computational cost to significantly reduce the total cost of quantifying the uncertainty of high–dimensional, expensive forward models. The algorithm is shown to be in detailed balance with the posterior distribution of parameters, and the computational gains of the algorithm is demonstrated on multiple examples. Additionally, I present an approach for exploiting a deep neural network as an ultra–fast model approximation in an MLDA model hierarchy. This method is demonstrated in the context of both 2D and 3D groundwater flow modelling. Finally, I present a novel approach to adaptive optimal design of groundwater surveying, in which MLDA is employed to construct the posterior Monte Carlo estimates. This method utilises the posterior uncertainty of the primary problem in conjunction with the expected solution to an adjoint problem to sequentially determine the optimal location of the next datapoint.

Acknowledgements

I don't believe anything, but I have many suspicions.

— Robert Anton Wilson

A confession: Completing a PhD was never an ambition of mine. I was always curious about the Universe and have conducted my own (sometimes accidental) experiments for as long as I can remember, but somehow I never really considered the option of doing research professionally. I would like to thank Dr Hedda Weitz and Professor Graeme Paton of the University of Aberdeen for encouraging me to take this path. I also want to thank my supervisor, Professor Tim Dodwell for the guidance and advice, for giving me the freedom to pursue my own ideas, and for trusting me to find my own way.

I want to thank my parents for always supporting me, even sometimes when I have made dubious life decisions. I am not sure whether to count the decision to complete a PhD as one of those yet. I would like to thank my Dungeons & Dragons group (you know who you are) for keeping me sane throughout the COVID–19 lockdowns. Vaggu–Bam will live forever. Last, but not least, I want to thank my partner, Mirjam Nanko, for being an indispensable intellectual sparring partner, the Devil's Advocate, a shoulder to cry on, and my best friend.

This work was funded as part of the Water Informatics Science and Engineering Centre for Doctoral Training (WISE CDT) under a grant from the Engineering and Physical Sciences Research Council (EPSRC), grant number EP/L016214/1.

If I forgot anyone, don't be sad. I only had one page.

Mikkel Bue Lykkegaard, Exeter, 25th August 2022

Contents

1. Introduction

Inverse problems are ubiquitous in science and engineering and emerge in such diverse fields as materials science, medical imagining, geophysics, acoustics, astronomy and many others. Inherently uncertain and often ill-posed, inverse problems constitute a colossal challenge in terms of both application, methodology and the underlying theory. The Bayesian perspective offers a biaxial antidote. First, by preserving the inherently probabilistic nature of these problems, it allows for rigorously quantifying the uncertainty. Second, by incorporating prior information and physics-informed regularisation, it offers a straightforward and transparent way to constrain ill-posed inverse problems. The computational workhorse of Bayesian inverse problems is Markov Chain Monte Carlo (MCMC), a large family of Monte Carlo algorithms capable of drawing samples from the Bayesian posterior distribution. While these methods have achieved widespread use in statistics and data-science, their application to inverse problems remains fairly limited and is mostly restricted to academic benchmarks and (rarely) case studies (see Chapter 2).

One apparent reason for this lack of uptake may be the computational burden of running MCMC for the relatively expensive models often encountered in the realm of inverse problems. Standard MCMC methods are innately wasteful, and often the forward problem must be solved many times to yield a single independent MCMC sample. At the heart of MCMC lives a fundamental paradox, pertaining to the proposal distribution used to generate MCMC samples. For an MCMC algorithm to be efficient, the proposal distribution must be well-aligned with the posterior distribution, the very object of the Bayesian enquiry. Moreover, there exists a trade-off between algorithmic efficiency (the acceptance rate) and statistical efficiency (the effective sample size), for which the proposal distribution must be carefully balanced.

For these reasons, immense scientific efforts are made to develop efficient MCMC algorithms, and the work presented here can be framed in this context.

Another reason for the perceived lack of uptake may simply be an insensitivity amongst practitioners to the both technical, economic and environmental advantages of knowing the exact uncertainty of a given inverse problem. In the context of e.g. environmental risk assessments, Monte Carlo analysis is often employed for forward uncertainty propagation. However, for similar endeavours involving inverse problems, the engineering solutions often rely on brute-force techniques and (historical, but essentially arbitrary) safety margins, which are wasteful and expensive. There are legitimate historical reasons for this discrepancy since Bayesian methods for inverse problems have only very recently become computationally feasible for practitioners. It is my opinion that it can also be attributed to an unhealthy attachment to deterministic problems and unique solutions. However, this is a question of epistemology and *Weltanschauung* rather than technical ability and beyond the scope of this work. Thomas Kuhn said that "the competition between paradigms is not the sort of battle that can be resolved by proofs" (Kuhn, 1996), referring to his view that each scientific paradigm is built upon a particular axiomatic worldview that does not necessarily admit proofs from other paradigms. While engineering sciences are particularly resistant to epistemological paradigm shifts, I expect the probabilistic worldview to prevail - not because it is "truer" than the deterministic one, but because it provides a richer framework to explain the data.

1.1 Aims and Objectives

The aim of the work presented in this thesis has been two-fold, encompassing both methodological and practical developments. First, I present a novel methodological development in the field of MCMC, which addresses critical challenges when dealing with high-dimensional and large-scale Bayesian inverse problems. MCMC is a popular method for sampling from unnormalised distributions, such as the ones emerging as the posterior distribution of Bayesian inverse problems. When this distribution is high-dimensional and the forward model constituting the likelihood (or – broadly –

data misfit) functional is computationally expensive, the standard MCMC methods become infeasible. Gradient-based MCMC alleviates the challenge of sampling from a high-dimensional distribution but requires additional information that is typically not readily available for complex models. Methods that exploit a cheaper approximation of the forward model, such as Delayed Acceptance (DA) MCMC (Christen and Fox, 2005) and Multilevel MCMC (MLMCMC, Dodwell et al. (2015)) handle both high-dimensionality and expensive models in a natural way that is also easy to implement, and can readily be combined with gradient-based MCMC if the forward model allows it. However, the original manifestation of the DA algorithm is a relatively inflexible approach, and MLMCMC is theoretically only ergodic at an infinite computational cost. Hence, the first aspect of this thesis is concerned with developing a novel MCMC algorithm, termed Multilevel Delayed Acceptance (MLDA), which combines the strengths of both methods, and – admittedly – inherits some of their weaknesses. MLDA is, in its non-adaptive variety, strictly ergodic and in detailed balance with the exact posterior, also at a finite computational cost. It can exploit a model hierarchy of arbitrary size, rather than just a two-level hierarchy, and allows for sampling subchains of extended length to decorrelate samples across levels. However, like most MCMC algorithms, it turns out to be difficult to parallelise.

Second, I present two distinct practical developments, particular to application-specific challenges within hydrogeological inverse problems, i.e. inverse groundwater flow modelling. The first of these concerns the design of a fast surrogate for the forward model. As hinted above, a significant computational bottleneck of MCMC is the cost of the forward model, which must be solved many times for the MCMC sampler to converge to the desired distribution. The advantage of using the various above-mentioned multilevel methods (DA, MLMCMC and MLDA) is that they are completely agnostic to the nature of the approximate model, and any reasonably good approximation can be used on the coarse level(s). The flexibility of artificial neural networks in general and *deep* neural networks in particular, make them a natural candidate for such model approximations, and exploiting this in the context of MLDA is the fundamental idea of the first practical development. Hence, in the enclosed paper, we design deep neural networks that can be employed as surrogate models in the context of uncertainty quantification of the groundwater flow problem using MLDA. The second practical development concerns a downstream application of the uncertainty estimates generated by exposing the groundwater flow problem to MLDA. Groundwater surveying is typically extremely costly due to the inherent variability of subsurface physical properties in combination with the practical complexity of establishing a monitoring well. We tackle the problem of optimally choosing the next monitoring well (given data from existing wells) for groundwater surveying, using both the uncertainty of the model parameters and the expectation of an adjoint state equation. The presented approach can be utilised to maximise the information gain of a sequential groundwater survey at a constant cost or to minimize the cost of the survey at some acceptable level of uncertainty.

1.2 Preliminaries

Each enclosed research paper contains theory and methodology sections relevant to the topic covered by the respective paper, with some overlap. However, in this section, I will introduce the overarching theoretical framework and discuss some topics that are not covered in depth by the papers but may contribute to a broader understanding of the topic at large.

1.2.1 The Bayesian Inverse Problem

The objective of the Bayesian inverse problem is to determine the distribution of parameters given noisy measurements. We begin by defining the *data-generating* model as

$$\mathbf{d} = \mathcal{F}(\boldsymbol{\theta}) + \boldsymbol{\epsilon} \tag{1.1}$$

where $\mathbf{d} \in \mathcal{D}$ is a vector of noisy measurements, $\mathcal{F} : \Theta \to \mathcal{D}$ is the (possibly nonlinear) forward map from the parameter space Θ to the measurement space $\mathcal{D}$, $\boldsymbol{\theta} \in \Theta$ is a vector of model parameters and $\boldsymbol{\epsilon}$ is the noise. If the forward operator perfectly describes the underlying true data generating process, $\boldsymbol{\epsilon}$ represents the measurement errors. However, in most practical applications, $\boldsymbol{\epsilon}$ will encapsulate both measurement errors, model misspecification errors, discretisation errors, etc. We assume that the noise is distributed according to some probability distribution $\boldsymbol{\epsilon} \sim \pi_\epsilon$, allowing us to define a misfit or *likelihood* functional $\mathcal{L}(\mathbf{d}|\boldsymbol{\theta}) = \pi_\epsilon(\mathbf{d} - \mathcal{F}(\boldsymbol{\theta}))$.

The distribution of parameters $\boldsymbol{\theta}$ given measurements $\mathbf{d}$ can then be expressed in terms of Bayes' Theorem:

$$\pi(\boldsymbol{\theta}|\mathbf{d}) = \frac{\pi_0(\boldsymbol{\theta})\,\mathcal{L}(\mathbf{d}|\boldsymbol{\theta})}{\pi(\mathbf{d})} \tag{1.2}$$

where $\pi(\boldsymbol{\theta}|\mathbf{d})$ is the *posterior* distribution of parameters given measurements, $\pi_0(\boldsymbol{\theta})$ is the *prior* distribution of parameters, $\mathcal{L}(\mathbf{d}|\boldsymbol{\theta})$ is the likelihood of observing the measurements given the parameters, and $\pi(\mathbf{d})$ is a normalising constant, commonly referred to as the *evidence*. At this point, we are confronted with a challenge. The evidence can be expanded to

$$\pi(\mathbf{d}) = \int_\Theta \pi_0(\boldsymbol{\theta})\,\mathcal{L}(\mathbf{d}|\boldsymbol{\theta})\,d\boldsymbol{\theta} \tag{1.3}$$

which is typically impossible to determine analytically and prohibitively expensive to approximate with sufficient accuracy. The key to unlocking this challenge is to recognise that $\pi(\mathbf{d})$ is constant with respect to the parameters $\boldsymbol{\theta}$. This allows us to write

$$\pi(\boldsymbol{\theta}|\mathbf{d}) \propto \pi_0(\boldsymbol{\theta})\,\mathcal{L}(\mathbf{d}|\boldsymbol{\theta})\,, \tag{1.4}$$

or, equivalently,

$$\frac{\pi(\mathbf{y}|\mathbf{d})}{\pi(\mathbf{x}|\mathbf{d})} = \frac{\pi_0(\mathbf{y})\,\mathcal{L}(\mathbf{d}|\mathbf{y})}{\pi_0(\mathbf{x})\,\mathcal{L}(\mathbf{d}|\mathbf{x})} \tag{1.5}$$

with $\mathbf{x}, \mathbf{y} \in \Theta$. In other words, while we cannot compute the unnormalised posterior density of a single parameter set $\pi(\boldsymbol{\theta}|\mathbf{d})$, we can compute the exact ratio between the posterior densities of two different parameter sets. This is the simple intuition underpinning the logic of Markov Chain Monte Carlo (MCMC) methods for Bayesian inverse problems, as well as related methods such as the Independence Sampler (IS).

More specifically, if we have some means of generating parameter realisations

```
Algorithm 1: Metropolis–Hastings MCMC
Choose $\boldsymbol{\theta}_0$. Then, for $t = 0, \ldots, N-1$:
    1. Given $\boldsymbol{\theta}_t$, generate a proposal $\boldsymbol{\theta}'$ from a proposal distribution $q(\boldsymbol{\theta}'|\boldsymbol{\theta}_t)$,
    2. Accept proposal $\boldsymbol{\theta}'$ as the next sample with acceptance probability
           $\alpha(\boldsymbol{\theta}'|\boldsymbol{\theta}_t) = \min\left\{1, \frac{\pi_0(\boldsymbol{\theta}')\,\mathcal{L}(\mathbf{d}|\boldsymbol{\theta}')\,q(\boldsymbol{\theta}_t|\boldsymbol{\theta}')}{\pi_0(\boldsymbol{\theta}_t)\,\mathcal{L}(\mathbf{d}|\boldsymbol{\theta}_t)\,q(\boldsymbol{\theta}'|\boldsymbol{\theta}_t)}\right\}$ ,
       i.e. set $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}'$ with probability $\alpha$, and $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t$ with probability $1-\alpha$.
```

$\boldsymbol{\theta}'$, or (colloquially) *proposals*, from a proposal distribution $q(\cdot|\cdot)$, we can sample indirectly from the posterior distribution $\pi(\boldsymbol{\theta}|\mathbf{d})$ by sequentially generating and comparing proposals to the (at any given time) current state $\boldsymbol{\theta}_t$ of the Markov chain. Broadly, the proposal is accepted as a sample $\boldsymbol{\theta}_{t+1}$ with a probability equal to the ratio of the proposal's posterior density to the state's posterior density (eq. 1.5). However, if the proposal distribution is asymmetric, care must be taken to ensure *detailed balance* (see Section 1.2.2). This procedure is referred to as the Metropolis–Hastings algorithm (Algorithm 1).

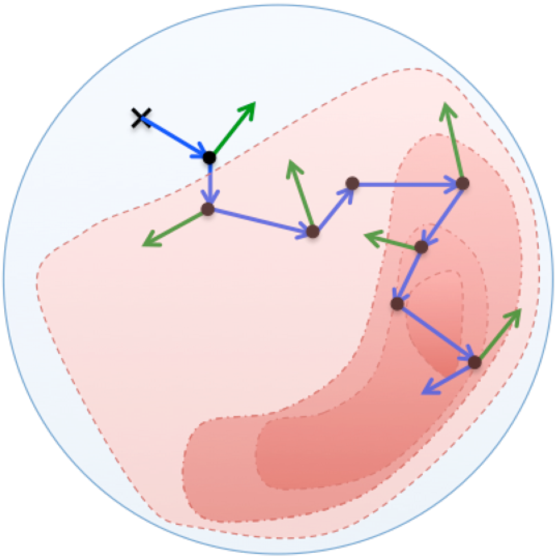


Figure 1.1: MCMC sampler traversing the parameter space Θ and converging towards a high-density region. The contours show isolines of a contrived posterior density, with higher saturation signifying higher density. The arrows show accepted MCMC samples (→●) and rejected proposals (→), respectively.

In this fashion, the Metropolis–Hastings algorithm will traverse the parameter space Θ, and gather samples according to their relative posterior densities (Figure 1.1).

As sampling progresses, it collects more samples from regions of high posterior density and fewer from regions of low posterior density. When converged, the Metropolis–Hastings algorithm will have yielded a sequence of samples $\{\boldsymbol{\theta}_0, \boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{N-1}\}$, which (after discarding an initial *burnin*) are distributed exactly according to the posterior distribution $\pi(\boldsymbol{\theta}|\mathbf{d})$. See e.g. Gelman (2004), Liu (2004) and Brooks (2011) for additional details.

1.2.2 Detailed Balance

A critical condition with respect to the design of MCMC algorithms is that of *detailed balance.* It is referred to numerous times in this work, but the underlying reasoning is not included in the papers since it is well-known and widely accepted. For completeness, I will briefly reproduce the reasoning following the argument presented in Liu (2004).

The detailed balance condition is in fact a proxy for another crucial condition, namely that the Metropolis transition kernel is *invariant* with respect to the target distribution. Let $\pi(\cdot)$ be the target distribution, $T(\cdot|\cdot)$ the transition kernel and let $\mathbf{x}, \mathbf{y} \in \Theta$, where Θ is the sample space. Then the invariance condition states that the following must hold:

$$\int \pi(\mathbf{x}) T(\mathbf{y}|\mathbf{x}) d\mathbf{x} = \pi(\mathbf{y}) \tag{1.6}$$

This condition can be difficult to prove in practice, and that is where the detailed balance condition comes in, which takes the following form:

$$\pi(\mathbf{x}) T(\mathbf{y}|\mathbf{x}) = \pi(\mathbf{y}) T(\mathbf{x}|\mathbf{y}). \tag{1.7}$$

This condition is more restrictive than the invariance condition, but if detailed balance is satisfied, so is invariance:

$$\int \pi(\mathbf{x}) T(\mathbf{y}|\mathbf{x}) d\mathbf{x} = \int \pi(\mathbf{y}) T(\mathbf{x}|\mathbf{y}) d\mathbf{x} = \pi(\mathbf{y}) \int T(\mathbf{x}|\mathbf{y}) d\mathbf{x} = \pi(\mathbf{y}). \tag{1.8}$$

It should be noted that the transition kernel $T(\cdot|\cdot)$ is the *true* transition kernel which includes the acceptance probability, as opposed to the proposal distribution $q(\cdot|\cdot)$:

$$T(\mathbf{y}|\mathbf{x}) = q(\mathbf{y}|\mathbf{x}) \min \left\{ 1, \frac{\pi(\mathbf{y})q(\mathbf{x}|\mathbf{y})}{\pi(\mathbf{x})q(\mathbf{y}|\mathbf{x})} \right\} \tag{1.9}$$

Hence, when designing a new MCMC algorithm with some (potentially exotic) proposal distribution $q(\cdot|\cdot)$, we can satisfy the detailed balance condition (1.7) by ensuring that indeed $\int T(\mathbf{x}|\mathbf{y}) d\mathbf{x} = 1$ and then simply plugging the correct proposal distribution into the acceptance probability $\alpha = \min \left\{ 1, \frac{\pi(\mathbf{y})q(\mathbf{x}|\mathbf{y})}{\pi(\mathbf{x})q(\mathbf{y}|\mathbf{x})} \right\}$. However, this is not always a trivial task when the actual proposal distribution is not obvious, as evident from the detailed balance proof for MLDA described in Chapter 4.

1.2.3 Adaptive MCMC

Another topic central to the material presented here is that of *adaptive* MCMC algorithms. While an exhaustive discussion is beyond the scope of this work, I would like to highlight that the concept itself is subject to debate between researchers, and some do not consider it to be a completely legitimate approach. Some of the potential issues with adaptive MCMC are discussed in Atchadé and Rosenthal (2005) and Andrieu and Moulines (2006). The reason for the apparent controversy is that adaptive MCMC algorithms cannot strictly be in detailed balance with the target distribution, since the proposal distribution (and hence the transition kernel) changes as MCMC sampling progresses. This problem can be completely alleviated by simply stopping adaptation after some burn-in, after which the (now constant) transition kernel is certainly in detailed balance with the posterior. However, the most widely used technique to ensure detailed balance for adaptive MCMC algorithms is that of *diminishing adaptation* (Andrieu and Thoms, 2008; Roberts and Rosenthal, 2009). According to this criterion, instead of abruptly ceasing adaptation, the MCMC algorithm should be designed in such a way that the tuning parameters subject to adaptation converge to some (hopefully optimal) values, so that adaptation naturally diminishes as MCMC sampling progresses. In this work I will assume the position that adaptive MCMC algorithms are indeed legitimate, so long as the diminishing adaptivity condition is satisfied, which is true for all the algorithms presented herein.

Not only has this assumption been studied closely in the aforementioned papers, but adaptivity is often a necessity in the context of high-dimensional Bayesian inverse problems with complex posteriors and no obvious and/or inexpensive way to compute the gradient. I will discuss adaptive MCMC in more detail in Section 2.2.1.

1.2.4 A remark on MCMC for Bayesian inverse problems

The majority of the content in this thesis revolves around the use of Markov Chain Monte Carlo (MCMC) methods for Bayesian inverse problems in general and the groundwater flow problem in particular. I will not go into more detail with respect to MCMC in this section, since the methodology is explained in great detail in the papers enclosed below as Chapters 3, 4 and 5. However, I want to remark that MCMC can be employed in any context, where sampling from some (unnormalised) distribution is required. Hence, MCMC, and in particular the ubiquitous NUTS sampler (Hoffman and Gelman, 2014), is used extensively in the context of statistical inference, such as Bayesian (generalised) linear regression models and mixed-effect models (Brooks, 2011). Conversely, MCMC is not the only method capable of tackling Bayesian inverse problems. Another widely used approach is that of Variational Inference (VI), where in place of sampling from the posterior distribution, it is approximated using some pre-defined probability distribution, or *variational* distribution. This is typically achieved by minimising the Kullback–Leibler divergence between the true posterior and the variational distribution. While the VI approach is usually computationally much cheaper than running MCMC, it is also more laborious to set up, and the resulting posterior is, critically, tainted by a bias induced by the choice of variational distribution. Hence, in this work, I will focus exclusively on MCMC methods. For an extensive overview of VI, see e.g. Wainwright and Jordan (2007), and for a recent application of VI to a Bayesian inverse problem, see e.g. Zhang and Curtis (2020).

2. Literature Review

My research has been balanced between two distinct scientific disciplines, each with a rich history of discoveries. Hence, the following literature review will visit both of these disciplines individually and in depth. First, I will address the existing literature on groundwater flow as a (Bayesian) inverse problem, and second, the most significant developments of Markov Chain Monte Carlo (MCMC) methodology. Finally, I will present an overview of the intersection between the two.

2.1 Groundwater Flow as an Inverse Problem

Groundwater flow modelling is ubiquitous in environmental engineering, and emerges in the context of multiple different technical endeavours, including pollution control and management, environmental risk assessment, the environmental fate of contaminants, nuclear waste disposal, water resource surveying, agricultural water management, integrated earth systems modelling, contaminant source location and seawater intrusion modelling, to name a few. The model complexity varies according to various circumstances, including whether the groundwater flow is in steady-state or transient, whether the aquifer is confined or unconfined, and whether transport equations are solved or not (Anderson, Woessner and Hunt, 2015). While the groundwater flow problem is not always posed (or understood) as an inverse problem, there are a number of inverse problems associated with groundwater flow modelling, because of some inherent characteristics of aquifers. Since aquifers are (sometimes deep) underground, both model input and output parameters are not directly observable and must be measured either indirectly or pointwise. Indirectly measuring any parameters is associated with uncertainty and typically involves solving some inverse problem, while point measurements are expensive, effectively resulting in data scarcity. Additionally, it is significantly simpler to measure the model outputs, such as hydraulic head, flux, and solute concentration, than the model inputs, such as hydraulic conductivity and storativity (Zhou, Gómez-Hernández and Li, 2014). Indirect methods for subsurface and aquifer characterisation include Electrical Resistivity Tomography (ERT, Loke et al. (2013)) and Electromagnetic (EM) methods, such as Airborne Electromagnetic (AEM, Auken, Boesen and Christiansen (2017)) and (towed) Transient Electromagnetic (tTEM, Auken et al. (2019)). Electromagnetic methods allow for direct inversion yielding complete images of the subsurface, however, the actual aquifer characteristics are derived from a (typically ill-conditioned) deconvolution problem. While these methods have applications in e.g. large-scale and preliminary surveying, in this review I will focus on their compliment, namely direct point measurements of subsurface hydraulics, which only allow for indirect inversion. While deconvolution problems are typically ill-conditioned and hence highly sensitive to measurement noise, the point measurement approach is typically strongly underdetermined and hence ill-posed. Both of these challenges require some reformulation of the original problem, for example reducing the number of estimated parameters, exploiting prior information to constrain the inversion, or imposing some form of *regularisation* (Zhou, Gómez-Hernández and Li, 2014).

2.1.1 Basic Methods

Early attempts at solving the inverse groundwater flow problem typically relied on extrapolating point measurements of hydraulic head to the entire domain and then performing direct inversion using the interpolated head function (Stallman, 1956; Nelson, 1960; Neuman, 1973). However, these approaches make strong assumptions about the hydraulic head function, do not handle noisy data well, and can lead to numerical instability. Hence, they will not be discussed further in this review and I will instead turn towards later developments that are more closely related to the techniques employed in this thesis. Please refer to .e.g. Neuman (1973), Carrera and Neuman (1986) and Zhou, Gómez-Hernández and Li (2014) for more detail on the early methods.

One of the first attempts to solve the problem by parameter dimensionality reduction was the Geostatistical Approach of Kitanidis and Vomvoris (1983). Here, it is assumed that some measurements of both hydraulic conductivity and hydraulic head are available. A geostatistical (random field) model is imposed on the hydraulic conductivity, drastically reducing the number of model parameters. Then, a joint probability distribution of the conductivity and the (linearised) hydraulic head is used to find the maximum likelihood estimate of the geostatistical parameters. Finally, the hydraulic conductivity of the entire domain is interpolated using co-kriging. While this approach is relatively inexpensive, the requirement to linearise the governing equations makes it unsuitable for problems with a high level of heterogeneity, and the requirement to include measurements of the hydraulic conductivity makes it unsuitable for many practical applications.

A more flexible approach is the Maximum Likelihood method of Carrera and Neuman (1986) which allows for incorporating any kind of measurements to estimate any model parameters, in a straightforward way. The likelihood function is maximised by minimising the corresponding objective function, which can also be augmented with prior information by way of a “penalty” term. While the authors do not frame their work in the context of a Bayesian inverse problem, this penalty term effectively corresponds to the Bayesian prior. The difference is not practical but theoretical, pertaining to the authors’ perspective on reality. In the Maximum Likelihood approach, the parameter dimensionality is typically reduced by zonation of the domain, so that the model parameters are assumed to be zone-wise constant. While this effectively removes the ill-posedness, it is far from realistic and it is also not obvious exactly how to determine the zones, a choice that may have a strong influence on the result of the inversion.

Another approach, which was developed concurrently with the Geostatistical and Maximum Likelihood methods, was the Pilot Point Method (Marsily, 1984; Certes and Marsily, 1991), see Figure 2.1. A simple idea, the approach proved highly effective and is still developed further, and broadly used (Alcolea, Carrera and

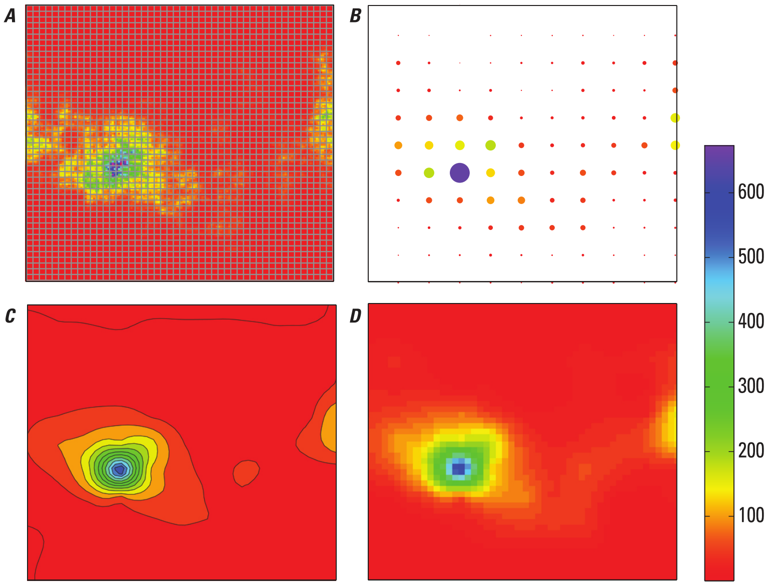


Figure 2.1: Principle of the Pilot Point Method (Doherty and Hunt, 2010). Panel A shows the true coefficient field and the model grid. Panel B shows the pilot points and their respective calibrated values (size and shape). Panel C shows the kriging map rendered by the pilot points and panel D shows the same kriging map projected onto the model grid.

Medina, 2006; Christensen and Doherty, 2008; Klaas and Imteaz, 2017). It is also one of the core features of the ubiquitous PEST software package (Doherty and Hunt, 2010; Doherty, Hunt and Tonkin, 2010), which I will discuss in more detail later. In the Pilot Point Method, a number of so-called "pilot points" are established on the domain and assigned some initial conductivity values, possibly guided by existing conductivity measurements. The conductivity of the entire domain is determined using kriging of both existing measurements and the fictitious pilot point values. The groundwater flow model is then solved to obtain the predicted heads, and a generalised least squares criterion, broadly corresponding to a Gaussian negative log-likelihood function, is computed with respect to the data. The pilot point values are then updated according to the gradient of this criterion with respect to each pilot point value, and the procedure is repeated until an acceptable error has been achieved. This way, the number of calibrated parameters can be controlled, and the problem remains well-posed. Pilot points can be placed either in bulk or sequentially, according to various criteria, including the variance of the kriging map, or simply at areas of particular interest, such as close to wells, boundaries or downstream of contaminant sources. While the Pilot Point Method has been fairly successful, it does entail some significant difficulties. First, it can result in artifacts, since the method seeks the best fit by manipulating only the pilot points. Second, the result of the Pilot Point Method may be biased by the placement of the pilot points. And third, the method yields only a single representation of the conductivity, rather than multiple plausible realisations. These issues were all addressed by Rubin et al. (2010), who also developed the anchored distributions approach, which is broadly a probabilistic (Bayesian) translation of the pilot point idea.

In light of the third difficulty of the Pilot Point Method, the self-calibrating method was developed (Gómez-Hernánez, Sahuquillo and Capilla, 1997; Capilla, Jaime Gómez-Hernández and Sahuquillo, 1997; Capilla, Jaime Gómez-Hernández and Sahuquillo, 1998). Here, instead instead of starting from a single kriging map, multiple conditional realisations are generated and then updated in a similar fashion to the Pilot Point Method. This, in essence, makes the self-calibrating method a form of ensemble method, since each realisation after updating represents an equally plausible representation of the aquifer. In Gómez-Hernánez, Franssen and Sahuquillo (2003), the authors provide a review of the method and compare it to other previous methods, including the Maximum Likelihood method of Carrera and Neuman (1986).

2.1.2 PEST

The PEST (Parameter ESTimation) software package is an environmental model calibration package, which is widely used for inverse groundwater modelling (Doherty and Hunt, 2010; Doherty, Hunt and Tonkin, 2010). It incorporates the Pilot Point Method along with other methods unique to PEST, including the *hydrid regularised inversion* method and the *subspace* or *null-space* Monte Carlo method.

The hybrid regularisation inversion methodology combines Truncated Singular Value Decomposition (TSVD) with Tikhonov regularisation to achieve stable inversion of a potentially under-determined system (Tonkin and Doherty, 2005). The model is first subjected to an initial over-determined calibration, using e.g. zonation of the domain to alleviate ill-posedness. Then, a Jacobian matrix describing the forward action of the fully parameterised, potentially under-determined model is computed using either perturbation or adjoint sensitivity equations. This matrix is decomposed using SVD and the decomposition is truncated at some relative threshold (see e.g. Hansen (2010)). The dominant "superparameters" then become the target of further calibration, and the remaining SVD modes are left untouched. A Tikhonov regularisation scheme is constructed for the base parameters (see e.g. Hansen (2010)), and the model is calibrated using the TSVD superparameters, subject to this Tikhonov regularisation scheme. The result is a model that is calibrated in a measurement-informed subspace and regularised according to whichever constraints the modeller imposes through the Tikhonov regularisation scheme. It should be mentioned that under certain conditions, the Tikhonov solution is equivalent of the Bayesian Maximum a Posteriori (MAP) estimate (Vogel, 2002).

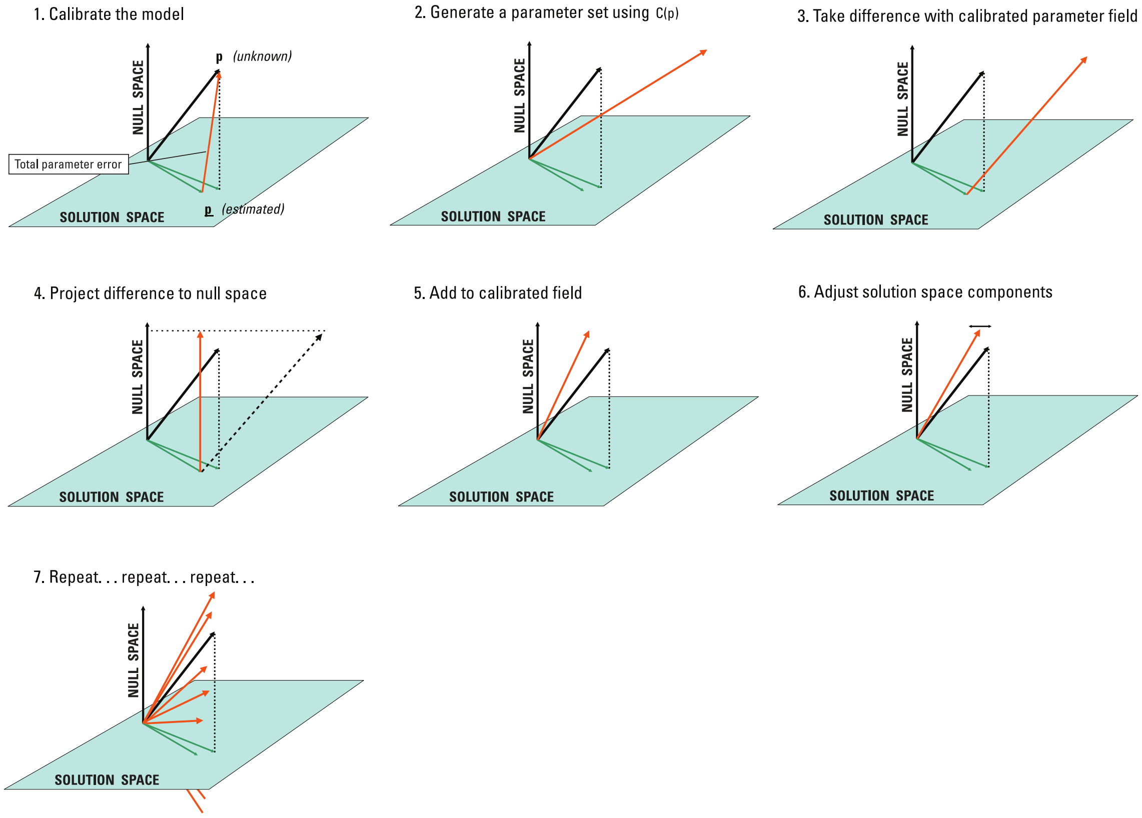


Figure 2.2: Workflow of the subspace (null-space) Monte Carlo method implemented in PEST (Doherty, Hunt and Tonkin, 2010).

The subspace (null-space) Monte Carlo method (Figure 2.2) is essentially an extension to the hybrid regularised inversion methodology, which allows for some quantification of uncertainty within the existing framework (Tonkin and Doherty, 2009). In this context, the dominant SVD components constitute the *calibration* *subspace*, while its complement, the remaining SVD components, constitute the *calibration null-space*. After calibration using the hybrid methodology described above, a Monte Carlo sample is drawn from the distribution of base parameters. The sample is projected into the calibration null-space, added to the calibrated parameters, and the model is recalibrated according to the new configuration. Since the calibration null-space constitutes the subspace of the model parameters that are not informed by the measurements, each Monte Carlo sample drawn this way is by design conditioned on the measurements. This process is repeated as many times as necessary, yielding a Monte Carlo estimate of the uncertainty of the model. While this method allows for determining the uncertainty of the model within the established hybrid framework, it does add significantly to the computational cost of the inversion, particularly if the parameter space is high-dimensional and the original problem is very ill-posed. The result also depends on several user-defined parameters, such as the TSVD truncation threshold and the Tikhonov regularisation constraints, both of which require extensive knowledge, not only of the physical problem but of the intricate details of each method. The Bayesian approach of Markov Chain Monte Carlo, which I will cover now, removes many of the difficulties associated with the traditional approaches to solving under-determined inverse problems, albeit at an even higher computational cost than any of the previously mentioned methods.

2.2 Markov Chain Monte Carlo

While an exhaustive review of Markov Chain Monte Carlo (MCMC) methods is beyond the scope of this thesis, I will here outline some of the most important developments in the field. The terms “MCMC” and the “Metropolis–Hastings algorithm” are often used interchangeably, highlighting the importance of the seminal papers by Metropolis et al. (1953) and Hastings (1970). While some later developments, at first sight, appear relatively far removed from these original formulations, most (if not all) MCMC methods can be formulated as special cases of the Metropolis–Hastings algorithm (Brooks, 2011). Of special interest is the Gibbs-sampler (Geman and Geman, 1984), which enjoyed high popularity for many years owing to the paper by

Gelfand and Smith (1990), since it requires relatively little tuning compared to previous MCMC algorithms. The `GS`-family of samplers, including `WinBUGS`, `OpenBUGS` (Lunn et al., 2009) and `JAGS` (Plummer, 2003), are all implementations of the Gibbs-sampler. While using the Gibbs-sampler requires little tuning, it does require the practitioner to specify conditional distributions for all the random variables subject to Gibbs-sampling. This may have been a contributing factor to the Gibbs-sampler falling out of use in many fields since that requires a deeper statistical understanding of the problem than what the average data science practitioner typically possesses. Another contributing factor was the appearance of the No-U-Turn Sampler (NUTS) (Hoffman and Gelman, 2014), an extension to Hamiltonian (or "Hybrid") Monte Carlo (HMC, Duane et al. (1987)), that automatically determines the essential tuning parameters of HMC. Both NUTS and HMC exploit the gradient of the target distribution, which is typically determined through automatic differentiation, to achieve more efficient MCMC sampling. Moreover, HMC and NUTS do not require the conditional distributions, and the software implementations allow practitioners to specify statistical models with relatively little effort. The development of the NUTS sampler has further popularised the use of Bayesian inference across many scientific disciplines, and it is the default sampler in the popular software frameworks `Stan` (Carpenter et al., 2017) and `PyMC` (Salvatier, Wiecki and Fonnesbeck, 2016).

2.2.1 Adaptive MCMC

Since the NUTS sampler relies on automatic differentiation to compute the gradient of the posterior distribution, its use is generally restricted to problems that can be defined in terms of closed-form probability distributions. For more complicated problems, such as PDE-constrained Bayesian inverse problems, the gradient of the posterior distribution is typically not readily available and can only be approximated through numerical methods, such as the finite difference method. For high-dimensional problems, this approach will result in an unacceptably high computational cost, and this is exactly the challenge that *gradient-free* MCMC methods aim to solve. Broadly, the aim is to construct an inexpensive proposal that is closely aligned to the posterior distribution. Since we typically know very little about the posterior distribution in advance of running MCMC (we run the MCMC exactly to explore the posterior), gradient-free methods typically involve some manner of *adaptivity.* The seminal paper by Haario, Saksman and Tamminen (2001), in which the Adaptive Metropolis (AM) algorithm was developed, laid the foundation for this now highly active field of research. The idea of AM is simple but effective. The proposal distribution is set to a multivariate Gaussian which is iteratively improved during sampling by using all previous samples to sequentially compute the covariance matrix. This is achieved by way of the following recursive formula:

$$\mathbf{C}_{t+1} = \frac{t-1}{t}\mathbf{C}_t + \frac{s_d}{t}\Big(t\,\bar{\mathbf{x}}_t\,\bar{\mathbf{x}}_t^T - (t+1)\,\bar{\mathbf{x}}_{t+1}\,\bar{\mathbf{x}}_{t+1}^T + \mathbf{x}_{t+1}\,\mathbf{x}_{t+1}^T + \varepsilon\mathbf{I}_d\Big) \tag{2.1}$$

where t is the current MCMC iteration, $\mathbf{C}_t$ is the sample covariance matrix at iteration t, $\mathbf{x}_t$ is the MCMC sample at iteration t, with $\bar{\cdot}$ signifying the arithmetic mean, $s_d = 2.4^2/d$ is a scaling parameter, d is the target dimension and ε is a small parameter that prevents $\mathbf{C}_t$ from becoming singular.

Using the AM algorithm, the autocorrelation of successive MCMC samples can be drastically reduced (Figure 2.3), as the proposal distribution adaptively approaches the target.

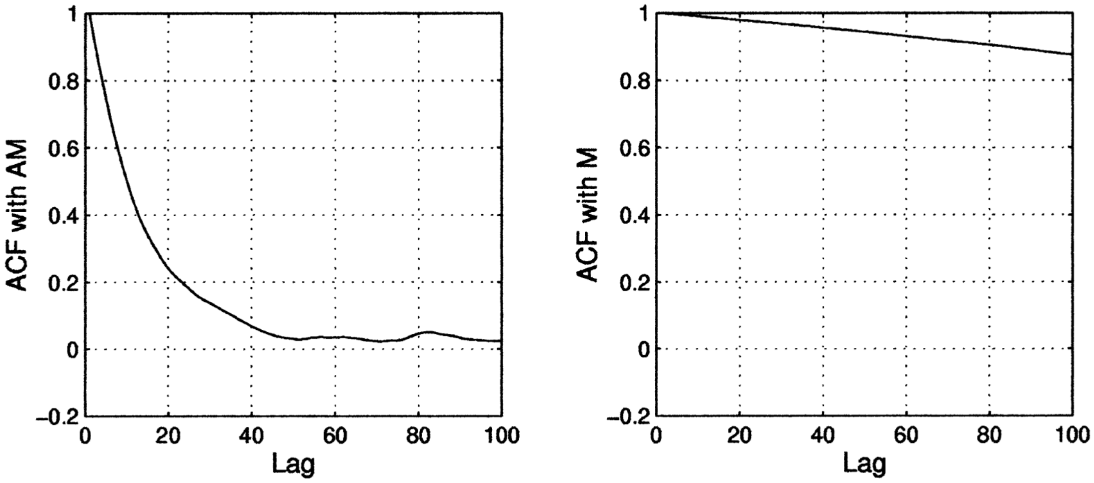


Figure 2.3: Autocorrelation functions for largest component of the target covariance matrix for an 8-dimensional uncorrelated Gaussian target. Adaptive Metropolis sampler (left panel) and Metropolis sampler (right panel) (Haario, Saksman and Tamminen, 2001).

The AM algorithm was later extended to a Gibbs-like setting with the development of the Single Component Adaptive Metropolis (SCAM) algorithm (Haario, Saksman and Tamminen, 2005). Later yet, Andrieu and Thoms (2008) developed a wide range of techniques that could be utilised for adaptive MCMC and explored some of the possible pitfalls. In this light, the most important aspect of designing an adaptive MCMC is to ensure that it exhibits *diminishing* or *vanishing adaptivity* (Roberts and Rosenthal, 2007; Roberts and Rosenthal, 2009). Broadly speaking, this condition dictates that, as the algorithm proceeds with sampling, the "amount" of adaptation approaches zero. The diminishing adaptivity condition is naturally obeyed by the AM algorithm and can be imposed by design on other schemes, such as using a Robbins-Monro recursion to determine a global scaling parameter (Andrieu and Thoms, 2008). In this context, I will also highlight a lesser know adaptive MCMC algorithm, the Adaptive Proposal (AP) distribution (Haario, Saksman and Tamminen, 1999), which lead to the development of the AM algorithm. The AP algorithm adapts the proposal distribution to only the latest samples, and hence does not, in fact, subscribe to the diminishing adaptation condition, and does not sample from the exact posterior (Haario, Saksman and Tamminen, 2001). It may still be useful if the posterior is strongly non-Gaussian and if exactness is not crucial.

While a complete overview of adaptive MCMC algorithms are beyond the scope of this paper, I will highlight two recent and promising avenues, namely those of *surrogate* HMC and *transport-map accelerated* MCMC. First, the work of Strathmann et al. (2015) who developed Kernel HMC, where an exponential family model is fitted to previous samples in a Reproducing Kernel Hilbert Space (RKHS). The gradient of the exponential family model is then used in place of the true gradient. This method was a further development of the Kernel Adaptive Metropolis–Hastings of Sejdinovic et al. (2014), where the covariance of the RKHS informs the proposal so that it locally targets areas of the parameter space with a high expected density. Second, the HMC using surrogate functions with random bases (RNS-HMC) of Zhang, Shahbaba and Zhao (2017), where the authors use shallow neural networks to approximate the target density. Multiple different algorithms are proposed, including a two-stage exploration-exploitation strategy, where (1) the parameter space is explored using standard HMC,

(2) a surrogate is constructed using the initial samples, and (3) the surrogate is exploited for further surrogate HMC sampling. This approach is similar to that of Rasmussen (2003). The authors also present a fully adaptive version (ARNS-HMC) of the same algorithm with diminishing adaptation using the principles of Andrieu and Thoms (2008), and a Riemann Manifold HMC (RMHMC) version, which aligns the method with the RMHMC of Girolami and Calderhead (2011). The third and final adaptive MCMC method I want to highlight is the transport-map accelerated MCMC of Parno and Marzouk (2018), a method that exploits optimal transport maps (El Moselhy and Marzouk, 2012) to generate proposals. Rather than adapting the proposal distribution itself, a reference proposal is drawn from a simple distribution, such as a Gaussian, and is then pushed through an adaptively constructed optimal transport map that connects the reference distribution with the target. While this approach is relatively uncomplicated for forward uncertainty propagation (Marzouk et al., 2016), the MCMC version requires inversion of the transport map, which can be computationally expensive for high-dimensional problems.

2.2.2 Preconditioned Crank-Nicolson

A relatively recent and important development in gradient-free MCMC proposals is the preconditioned Crank-Nicolson (pCN) proposal (Cotter et al., 2013). This proposal is constructed through the Crank-Nicolson discretisation of a Stochastic Partial Differential Equation (SPDE) describing Brownian motion. With some clever manipulation the proposal arrives, in its simplest form, at an expression very closely resembling a Random Walk Metropolis–Hastings (RWMH) proposal:

$$\boldsymbol{\theta}' = \sqrt{1-\beta^2}\,\boldsymbol{\theta}_t + \beta\boldsymbol{\xi} \tag{2.2}$$

where $\boldsymbol{\theta}'$ is the pCN proposal, $\boldsymbol{\theta}_t$ is the current MCMC state, $\beta \in [0,1]$ is a scaling parameter and $\boldsymbol{\xi} \sim \mathcal{N}(0, \mathbf{C}_0)$ is a random draw from the prior distribution of parameters $\mathcal{N}(0, \mathbf{C}_0)$. However, the pCN proposal has one crucial advantage over RWMH. The pCN proposal is robust with respect to the target dimension, whereas the RWMH proposal is not. Plainly speaking, as the dimensionality of the target distribution grows, the RWMH proposal requires smaller step sizes, while for the pCN proposal we can (theoretically) maintain a constant step size (Cotter et al., 2013). However, the pCN proposal requires the prior to be Gaussian and does not necessarily perform well for strongly non-Gaussian posteriors, or posteriors with covariances that were not specified by the prior. This makes the algorithm particularly suitable for problems with a prior specified by e.g. a Karhunen–Loève decomposition of a Gaussian Process, but not as universally applicable as the AM algorithm. In the same paper, Cotter et al. (2013) also developed the preconditioned Crank-Nicolson Langevin (pCNL) proposal and an Independence Sampler (IS) based on the same insights as the "Vanilla" pCN proposal. However, the pCNL algorithm, similarly to the Metropolis-adjusted Langevin (MALA) proposal (Roberts and Tweedie, 1996; Roberts and Rosenthal, 1998), requires the elusive gradient, and the IS requires the prior to be relative close to the posterior, and hence these proposals are of little interest to our applications. The pCN algorithm was developed further by Law (2014), who proposed the *operator-weighted* pCN proposal, which allows for weighting different dimensions differently, according to some linear operator $\mathbf{B}$:

$$\boldsymbol{\theta}' = \sqrt{\mathbf{B}_t}\,\boldsymbol{\theta}_t + \sqrt{\mathbf{I} - \mathbf{B}_t}\,\boldsymbol{\xi}. \tag{2.3}$$

If $\mathbf{B}_t = (1 - \beta^2)\mathbf{I}_d$, this simplifies to the original pCN proposal (Cotter et al., 2013), but Law (2014) suggests constructing $\mathbf{B}_t$ from a relaxed Hessian of the forward operator with respect to the parameters, evaluated near the posterior mode, and exploit this curvature information to improve the sampling efficiency. Cui, Law and Marzouk (2016) expanded further on this idea and introduced a family of *Dimension-independent Likelihood-informed* (DILI) samplers that restrict pCN sampling to a likelihood-informed subspace, and simply sample from the prior in its complement, not unlike the subspace Monte Carlo of Tonkin and Doherty (2009). It also exploits local gradient information to determine the best proposal direction, and hence may not be viable for expensive problems with no way to directly compute this gradient. Of more interest to my applications is the Adaptive pCN (ApCN) proposal, developed by Hu, Yao and Li (2016). This proposal can be considered a hybrid of the operator-weighted pCN proposal and the AM algorithm, in that it exploits past samples to construct a weighting operator that is closely aligned with the posterior. However, this proposal requires taking the square root of a matrix at each adaptive update, and this may be not viable for very high-dimensional problems. In this light, the same authors developed the Hydrid ApCN algorithm (Zhou et al., 2017), which uses the standard AM algorithm to sample from a predefined subspace, and the standard pCN proposal to sample from its compliment, completely avoiding this complication. However, unlike every other pCN variant mentioned above, it does not strictly have a dimension-independent acceptance rate.

2.2.3 Differential Evolution Adaptive Metropolis

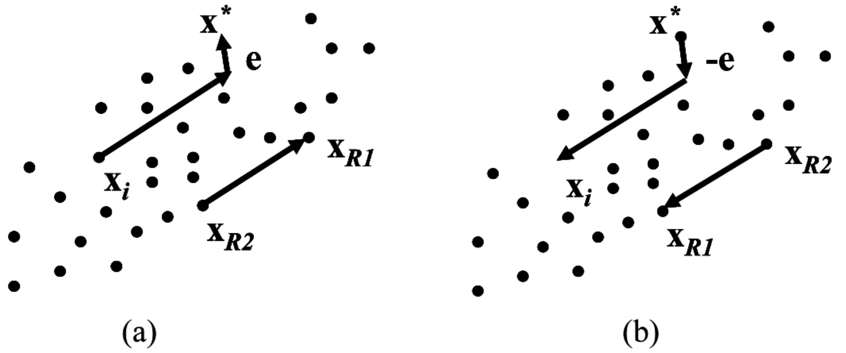


Figure 2.4: Principle of Differential Evolution Markov Chain. Here, $\mathbf{x}_i$ represents the state of one chain, and the proposal $\mathbf{x}^*$ is the sum of the difference between two other states $\mathbf{x}_{R1}$ and $\mathbf{x}_{R2}$ and a random perturbation $\mathbf{e}$. Panel (a) shows the proposal generating mechanism, while panel (b) illustrates that the proposal is reversible. From ter Braak and Vrugt (2008).

Another interesting subset of gradient-free MCMC algorithms is the family of DREAM-samplers (DiffeRential Evolution Adaptive Metropolis, (Vrugt, 2016)), which belong to a larger family of algorithms commonly called population-based MCMC samplers. While merely tangential to the research presented in this thesis, the DREAM samplers have achieved some uptake in more applied studies (Hinnell et al., 2010; Keating et al., 2010; Malama, Kuhlman and James, 2013; Laloy et al., 2013; Shafii, Tolson and Matott, 2014), and are hence included in this literature review for completeness. They were developed from the original Differential Evolution Markov Chain (DE-MC) (ter Braak, 2006), which uses the states of multiple parallel chains to generate proposals (Figure 2.4). This allows for sampling efficiently from complex, multimodal distributions, but the original algorithm requires many parallel chains to function correctly, i.e. $N = 2d$ for a d-dimensional target. This shortcoming motivated the development of the DE-MC$_{\mathrm{Z}}$ algorithm (ter Braak and Vrugt, 2008), where the Z refers to an archive of past samples. The introduction of an archive not only allows for using significantly fewer parallel chains but also effectively makes the DE-MC$_{\mathrm{Z}}$ algorithm an adaptive one. The DE-MC$_{\mathrm{Z}}$ proposal takes the following form:

$$\boldsymbol{\theta}' = \boldsymbol{\theta}_t + (\mathbf{I}_d + \mathbf{e})\gamma(\delta, d)\left[\sum_{i=1}^{\delta} \boldsymbol{\theta}_{R1(i)} - \sum_{j=1}^{\delta} \boldsymbol{\theta}_{R2(j)}\right] + \boldsymbol{\xi} \tag{2.4}$$

where $\mathbf{e} \sim \mathcal{U}_d(-b, b)$ and $\boldsymbol{\xi} \sim \mathcal{N}_d(0, \sigma^\star)$ are random perturbations with b and $\sigma^\star$ small, $\gamma(\delta, d)$ is a scaling function with δ the number of archival pairs used to construct the proposal, and $R1(i)$ and $R2(j)$ are indices of previous samples from the archive. The acronym DREAM was coined by Vrugt et al. (2009), who also introduced another adaptive feature to the algorithm. Seeing that perturbing every parameter at each proposal step often leads to small and trivial moves, they developed a method called Randomised Subspace Sampling, in which each parameter is perturbed according to a crossover probability, which is adaptively determined to facilitate long moves. Finally, Laloy and Vrugt (2012) extended the algorithm with Multiple-Try Metropolis functionality (Liu, Liang and Wong, 2000), allowing for evaluating multiple proposals and picking (probabilistically) the best one. This can be useful in cases where there are idle processors, but it does lead to an appreciably higher waste of computational resources.

Delayed Acceptance

The two final topics I will cover in this section are the distinct but analogous methods Delayed Acceptance (DA) MCMC and Multi-level Markov Chain Monte Carlo (MLMCMC). Both methods exploit one or more *coarse models* or Reduced Order Models (ROMs) to reduce the computational cost of running MCMC, but each method has its own advantages and disadvantages that should become apparent. Delayed Acceptance is a flexible two-stage method first proposed by Christen and

Fox (2005), where a single ROM is used to "filter" proposals before passing them on the full order model. When the ROM is state-independent, the DA method is identical to the Surrogate Transition Method of Liu (2004) with a single surrogate transition step and the preconditioned two-stage MCMC of Efendiev, Hou and Luo (2006). The DA sampler works by running any Metropolis–Hastings sampler on the coarse level, which is set to target the approximate posterior generated by the ROM using any valid proposal distribution. Only if the proposal is accepted on the coarse level, it will be evaluated on the fine level, where it will be subject to a second accept/reject step. While the sampler on the coarse level does not sample from the exact posterior, this second accept/reject step ensures that the samples on the fine level come from the exact posterior. One of the greatest attractions of DA is the flexibility with respect to choosing a ROM. In their original paper, Christen and Fox (2005) used a local linearisation as the coarse model, corresponding to a state-dependent ROM, which would not generate a valid Markov Chain when using the approach of Liu (2004). Other examples of methods that have been successfully employed as coarse models in DA sampling include polynomial chaos expansions (Laloy et al., 2013), inverse-distance weighted averages (Sherlock, Golightly and Henderson, 2015) and deep neural networks (Lykkegaard, Dodwell and Moxey, 2021). However, data-driven models may add significantly to the precomputation cost, and may not always perform well, particularly in the tails of the distribution. Hence, when the posterior is constrained by some model where equations are solved on a grid, the most obvious choice for a coarse model might simply be the same equations solved on a coarser grid. This perspective was explored in detail by Kaipio and Somersalo (2007), who addressed some issues of model discretisation in the context of Bayesian inverse problems, and developed a theory of how to handle the associated model discretisation error in a rigorous, Bayesian manner. A similar idea was explored by Brynjarsdóttir and O'Hagan (2014) who used a Gaussian Process Regression to model the discrepancy between the model output and the data. However, Brynjarsdóttir and O'Hagan (2014) considered to problem of model *misspecification*, rather than model *discretisation*, and the paper is only mentioned here for completeness. In Kaipio and

Somersalo (2007), the authors suggest constructing a discretisation error model from the prior distribution of parameters. While this is a perfectly sensible thing to do, it may not always lead to the most useful error model if the discretisation error varies strongly across the parameter space and the posterior support is markedly different than the prior. To alleviate this problem, posterior or *adaptive* Approximation Error Models (AEMs) were later developed (Cui, Fox and O'Sullivan, 2011; Cui, Fox and O'Sullivan, 2012; Cui, Fox and O'Sullivan, 2019). These AEMs are tailor-made for Delayed Acceptance MCMC and construct a (Gaussian) error model while sampling by extracting the discrepancy between the full order model and the ROM every time both models are evaluated with the same parameter set (Figure 2.5). There are broadly two different approaches, the *state-independent* AEM and the *state-dependent* AEM. The state-dependent AEM is mostly better, however, if the forward operator is strongly ill-conditioned then the state-independent AEM may provide better stability.

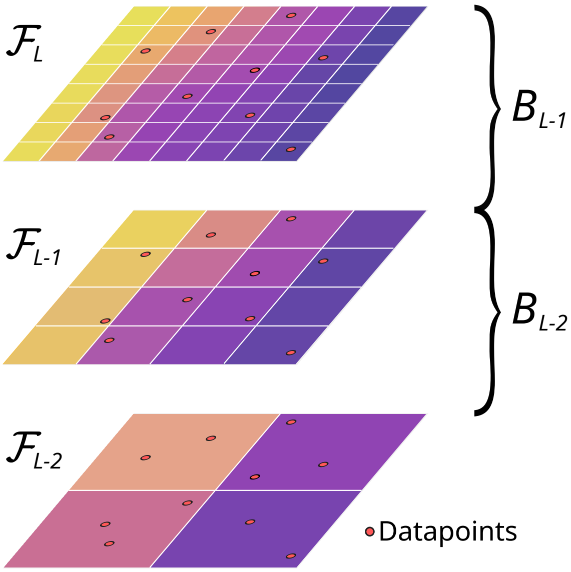


Figure 2.5: Principle of approximation error models generalised to a multi-level context. For each coarse model level $\mathcal{F}_\ell$, $\ell \in \{1, 2, \dots, L-1\}$, the bias B_ℓ describes the difference of the model output and the model output of the next-finer level $\mathcal{F}_{\ell+1}$. Typically, this is modelled using a Gaussian, so that $B \sim \mathcal{N}(\mu_B, \Sigma_B)$.

2.2.4 Multilevel MCMC

The MLMCMC method is a relatively recent development (Dodwell et al., 2015). It was motivated by the now ubiquitous Multilevel Monte Carlo (MLMC) method first proposed in Giles (2008a) and Giles (2008b). A similar idea can also be found in Heinrich (2001). The method was studied in the context of elliptic Partial Differential Equations (PDEs) with random coefficients, a class of problems that the groundwater flow problem belongs to under certain assumptions, by Cliffe et al. (2011) and Charrier, Scheichl and Teckentrup (2013). The central idea in MLMC is to use a particular MLMC estimator taking the shape of a telescoping sum to construct Monte Carlo estimates of a quantity of interest (QoI) using samples from all levels of a model hierarchy with an arbitrary number of levels (Giles, 2008a):

$$\mathbb{E}(\mathcal{Q}_L) = \mathbb{E}(\mathcal{Q}_0) + \sum_{\ell=1}^{L} \mathbb{E}(\mathcal{Q}_\ell - \mathcal{Q}_{\ell-1}) \tag{2.5}$$

where $\mathcal{Q}_\ell$ is the quantity of interest on level $\ell = 0, \ldots, L$. The higher the level, the fewer Monte Carlo samples are evaluated and vice versa. This effectively yields a Monte Carlo estimate with an accuracy reflecting the resolution of the finest model, but with a variance and a Monte Carlo Standard Error (MCSE) reflecting the multitude of samples across levels. This is commonly referred to as Variance Reduction (VR). The MLMCMC of Dodwell et al. (2015) takes this idea from the domain of forward uncertainty propagation and into the realm of uncertainty quantification for Bayesian inverse problems. While MLMCMC appears superficially similar to DA, there are crucial differences. In DA, coarse samples that are rejected by the second Metropolis accept/reject step are simply discarded, which ensures that the samples on the fine level come from the exact posterior. In MLMCMC, the coarser samplers pass proposals to the finer samplers, and continue from where they were, regardless of whether a sample was accepted on the finer levels or not (Figure 2.6). This has two major consequences. First, it allows for massively parallelising the MLMCMC sampler (Seelinger et al., 2021), and second, it requires samples passed from the coarser levels to be independent (Dodwell et al., 2015).

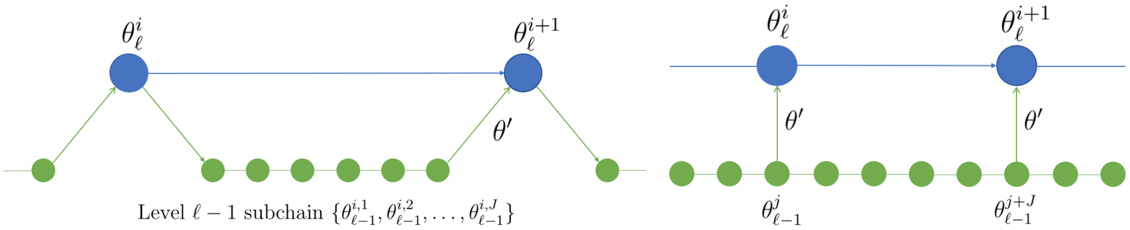


Figure 2.6: The conceptual difference between MLDA (left panel) and MLMCMC (right panel). Since DA can be framed as a two-level single-step version of MLDA, this also applies to DA. While the coarse samples in (ML)DA are consecutively realigned with the fine distribution according to the second Metropolis accept/reject step, the coarse sampler in MLMCMC is allowed to continue sampling, whether the proposal was accepted or not.

Parallelisation is a massive advantage over most other MCMC algorithms (barring Multiple-Try Metropolis (Liu, Liang and Wong, 2000)), where parallelisation is commonly restricted to either running multiple MCMC samplers simultaneously (which should be done in any case to diagnose convergence), or to simply parallelise the likelihood function, which is usually the most computationally expensive element. However, the independence requirement is a potentially significant complication. MCMC samples are necessarily autocorrelated because of the sequential nature of the algorithm, and while independence can theoretically be ensured with infinitely many samples, in practice this autocorrelation may lead to an indeterminable bias in the multilevel estimator (Fox, 2021). For the DA or Surrogate Transition Method, achieving VR is even less straightforward. For the multilevel estimator to be valid, both the (coarse) samples on any level and the proposals to the next-finer level must come from the same (stationary) distribution, which is not generally the case for the DA sampler, since the second accept/reject step consecutively realigns the coarse sampler with the target (fine) distribution. In this light, the coarse samples come from a mixture distribution “between” the two distributions generated by the coarse and fine models, respectively. One possible workaround is the Randomised Surrogate Transition proposal, outlined in Chapter 4. This approach ensures (probabilistically) that coarse samples are indeed from the same mixture distribution as the proposals to the next-finer level.

2.3 MCMC for Groundwater Flow Modelling

The first attempt at using MCMC for inversion and uncertainty quantification of a groundwater flow problem was, to my knowledge, made by Oliver, Cunha and Reynolds (1997). Their methodology consisted of using a Gibbs-like sampler to update the conductivity for only one grid cell at each iteration of the algorithm. This was done to ensure stability and increase the acceptance rate, but it makes the algorithm extremely slow to converge, particularly for high-dimensional problems. To alleviate this problem, Fu and Gómez-Hernández (2009) proposed the Blocking MCMC approach, where at each iteration a block of grid cells are updated at once, using a proposal conditioned on each updated cell and their immediate neighbourhood. The authors also present a two-stage version of their algorithm referring to the work of Efendiev, Hou and Luo (2006) but do not acknowledge explicitly that the presented algorithm constitutes a Delayed Acceptance sampler, possibly because the two-stage method of Efendiev, Hou and Luo (2006) was published in a journal specifically targeting hydrology and hydraulics. Simultaneously, Dostert, Efendiev and Mohanty (2009) considered the two-stage approach from a different angle. Addressing Richard's equation, an infamously difficult nonlinear PDE describing unsaturated flow in the vadose zone, they utilised the Multiscale Finite Element Method (MsFEM) (Efendiev, Ginting and Hou, 2004) to construct coarse-grid basis functions and then employed the MsFEM formulation as a coarse model. The significantly lower cost of the coarse model allowed them to use what they refer to as a preconditioned coarse-gradient Langevin algorithm as the coarse proposal, where the gradient was computed using finite differences. While the MALA algorithm is certainly an improvement compared to RWMH, the finite difference approach to gradient estimation requires $d+1$ model evaluations for a d-dimensional problem, and it is not clear how this method would compare to a well-tempered AM sampler on the coarse level. A similar approach was taken in Mondal et al. (2010), where the authors use upscaling (Durlofsky, 1998) and a mixed MsFEM formulation (Arbogast, 2002) to solve the two-phase flow equations on a coarse grid. Additionally, the authors present two interesting developments. First, the hierarchical modelling of hydraulic conductivity by discrete geological facies populated with Gaussian Processes (GP). The facies boundaries are modelled with piecewise linear functions, each describing an interface, and each facies is then equipped with a GP represented by a truncated Karhunen–Loève (KL) expansion (Figure 2.7). The piecewise linear functions are described by an arbitrary number of points with stochastic location, resulting in an MCMC sampling problem with an arbitrary target dimensionality. Second, the authors circumvent this apparent problem by using Reversible jump MCMC (Green, 1995), an algorithm designed to achieve exactly this, while maintaining detailed balance. They demonstrate that the approach can recover simple channel geometries, but the number of channels must be known in advance, as there is no apparent way to add a complete set of points and an additional GP describing a new channel.

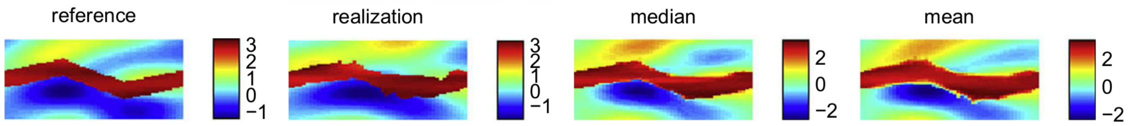


Figure 2.7: Piecewise linear facies boundaries populated with GPs. Left to right: the reference conductivity field, one posterior realisation, the median of the posterior and the mean of the posterior. Adapted from Mondal et al. (2010).

2.3.1 DREAM MCMC Samplers

The DREAM MCMC samplers described in Section 2.2 have achieved some uptake in the applied groundwater modelling community and as a black-box MCMC framework for theoretical studies of e.g. geological models. In Keating et al. (2010), the authors compare the results of PEST (see Section 2.1) and DREAM using real data from a very challenging inverse problem, namely the nuclear test site in Yucca Flat, Nevada, US (Fenelon, 2005). With respect to PEST, the authors calibrate the model using the Covariance Matrix Adaptation Evolutionary Scheme (CMA-ES, (Hansen and Ostermeier, 2001)) and then use null-space Monte Carlo to determine the uncertainty. With respect to DREAM, the authors use the standard approach described in Vrugt et al. (2009), without an archive. This may be more theoretically correct than using an archive but is very computationally expensive. Since the true parameters of the problem are unknown for this real-world example, it is not clear which of the methods is better. The authors claim that "each method is capable of providing consistent estimates of parameter uncertainty and of providing samples from posterior parameter distributions that, in the main, are consistent with each other", but there are certain differences that are not addressed in the paper. The most striking issue is that, while both methods identify broadly the same parameters as constrained by the calibration, they do not agree on their degree of "unconstrainedness" (Figure 2.8).

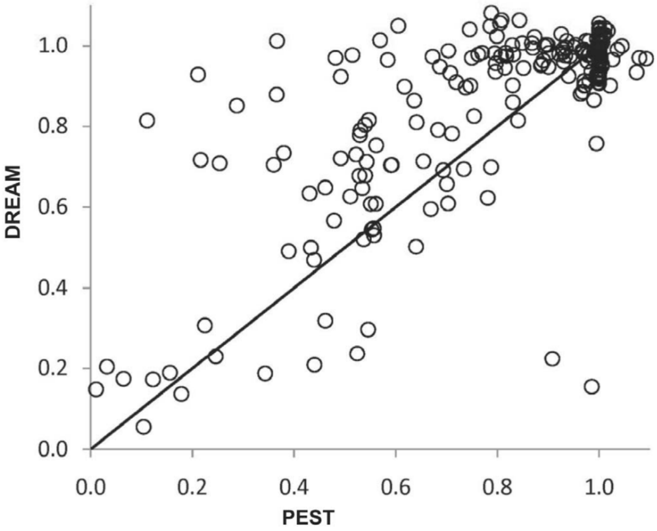


Figure 2.8: Relative "unconstrainedness" of each parameter of the model of Keating et al. (2010), measured by $\sigma^* = \sigma_{\text{posterior}}/\sigma_{\text{prior}}$. Results from the DREAM MCMC sampler vs. results from subspace Monte Carlo (PEST)

The authors explain that it is indeed a highly complex problem and that neither method is capable of capturing the true parameter values, nor the true uncertainty. However, deeper investigations into the root causes of the discrepancies and the potential flaws of both methods are warranted. The uncertainty quantification of PEST and DREAM were also compared in the study by Malama, Kuhlman and James (2013). The primary objective of the study, however, was a comparison of different models to predict the breakthrough time of different tracers through core samples. The authors did not consider the conductivity of the medium as stochastic, and hence the stochastic dimensionality of the problem was significantly smaller than the previously mentioned studies. However, it does provide a much more absorbable appraisal of the differences between PEST and DREAM. The two methods arrive at broadly the same median parameter values, but the uncertainty estimates from PEST exhibit some artefacts that are difficult to account for. According to the authors, they could not identify a null space for the PEST uncertainty analysis and assumed that it was one-dimensional. This highlights some of the potential issues regarding the hybrid regularisation approach of PEST (Tonkin and Doherty, 2005), namely that the truncation threshold for the TSVD is a modelling choice. Singular dimensions in the calibration space are assumed to be *absolutely* determined by the data, and all uncertainty is projected into the null-space. This is not reflective of the true nature of the inverse problem, where all parameters should reflect, at the very least, the uncertainty of the measurements. The uncertainty quantification produced by the DREAM MCMC sampler is sharply contrasted to the one produced by PEST. While the authors do not provide MCMC diagnostics, the presented posterior densities appear, at least at face value, credible.

A study related to the work presented in the enclosed journal paper on using deep neural networks as coarse models for Delayed Acceptance (DA) MCMC (Lykkegaard, Dodwell and Moxey, 2021) was the research presented in Laloy et al. (2013), who instead used a Polynomial Chaos Expansion as a coarse model and DREAM as the coarse MCMC sampler. While Laloy et al. (2013) demonstrate their methodology on a significantly more complicated forward model, there are some issues with respect to the presented results. The authors admit that their sampler did not formally converge according to the $\hat{R}$-criterion suggested by Gelman and Rubin (1992), which is currently not even considered sufficiently conservative to guarantee convergence (Gelman, 2004; Vehtari et al., 2020). Instead, the authors resort to using the root-mean-square error (RMSE) as a rudimentary measure of convergence, which has little meaning in the context of MCMC. Hence, in essence, Laloy et al. (2013) employ their DA sampler as a glorified optimisation algorithm, and the presented uncertainty quantification can most likely not be trusted. The DREAM MCMC sampler was also employed for inversion and uncertainty quantification in Laloy et al. (2015), where the authors present an alternative parameterisation of Gaussian Processes known as Circulant Embedding (Dietrich and Newsam, 1997), in the context of a groundwater flow problem. The authors compare the method to the popular KL expansion and show that the Circulant Embedding dimensionality reduction outperforms the KL expansion, and is capable of reproducing small scale variability, even in a relatively low dimensional parameter space. Another advantage of this approach is that the computational complexity is significantly lower than that of KL decomposition, which allows for making e.g. Gaussian kernel parameters stochastic rather than fixing them in advance of running the MCMC. The drawback of this dimensionality reduction is that the variance of the reduced-order Gaussian Random Field does not exactly correspond to the full-order one. In Laloy et al. (2018), the authors present a novel approach to generating realisations of a (random) hydraulic conductivity field, using a Spatial Generative Adversarial Network (SGAN) and use the DREAM MCMC sampler to demonstrate the method on an inverse groundwater flow problem. As discussed in Heße, Comunian and Attinger (2019), the Gaussian Process approach to modelling conductivity employed in many studies is not representative of some geological structures, including high-permeability channels, geological faults and other discontinuities. The Multiple-Points Statistics (MPS) approach (Guardiano and Srivastava, 1993; Strebelle, 2002) allows for exploiting complex prior information enclosed in a training image to generate more realistic geological structures, but it is fairly computationally demanding, as realisations are constructed iteratively, point by point. In the SGAN-based approach, rather than scanning the training image for matching geological structures, a convolutional neural network is trained to generate images that a structurally similar to the original training image (Figure 2.9). While the precomputation cost is higher than for the MPS approach, realisations can subsequently be generated at a very low cost. The method represents a fast and flexible way to generate random realisations of complex geological structures, but there are some caveats. First, the prior distribution is more or less nonsensical, and consist of uniform noise that is then propagated through the generator network to generate an image. Second, the generator map is not necessarily well-behaved, and similar generator images could be far apart in parameter space, which could complicate exploration of the full posterior significantly.

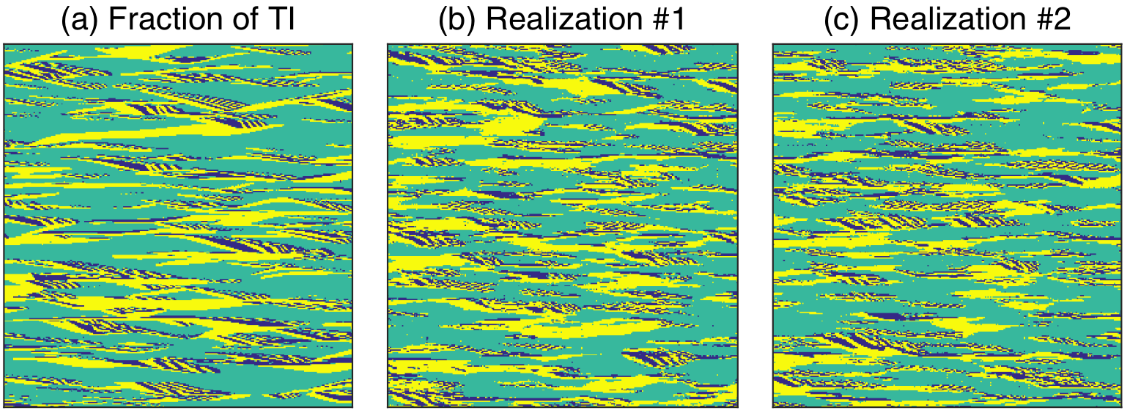


Figure 2.9: A fraction of the original training image intended to represent a braided river aquifer and two random realisations produced by an SGAN trained on the same training image (Laloy et al., 2018).

2.3.2 Benchmarks

The groundwater flow problem is a popular benchmark in theoretical studies of MCMC, particularly of methods aimed at problems constrained by expensive forward models. The steady-state equation for confined aquifers is in fact Poisson's equation, which also governs e.g. heat flow and other diffusion problems. While relatively simple, it represents a canonical example of an elliptic Partial Differential Equation (PDE), a class of problems for which there is usually no analytical solution, and the numerical solution involves solving a typically large system of equations. Here, I present a review of significant MCMC developments that employ the groundwater flow problem as a benchmark. While their primary focus is MCMC methodology, they each provide insight into the governing PDE, inverse solution strategies and uncertainty quantification. I remark that the groundwater flow problems used as benchmarks in the studies below are not always realistic and often very simple. However, the Bayesian perspective and the various geological priors may still serve as a template for handling harder and more realistic hydrogeological inverse problems.

In Higdon, Lee and Holloman (2003), the authors demonstrate the use of Gaussian Markov random fields (MRF) and Gaussian Processes as hydraulic conductivity models on an example of the breakthrough time of conservative tracers (Figure 2.10). The study was otherwise dedicated to an MCMC algorithm that exploits a coarse approximation to improve the mixing of the fine MCMC chain by swapping proposals between the fine and coarse sampler using Metropolis coupled MCMC

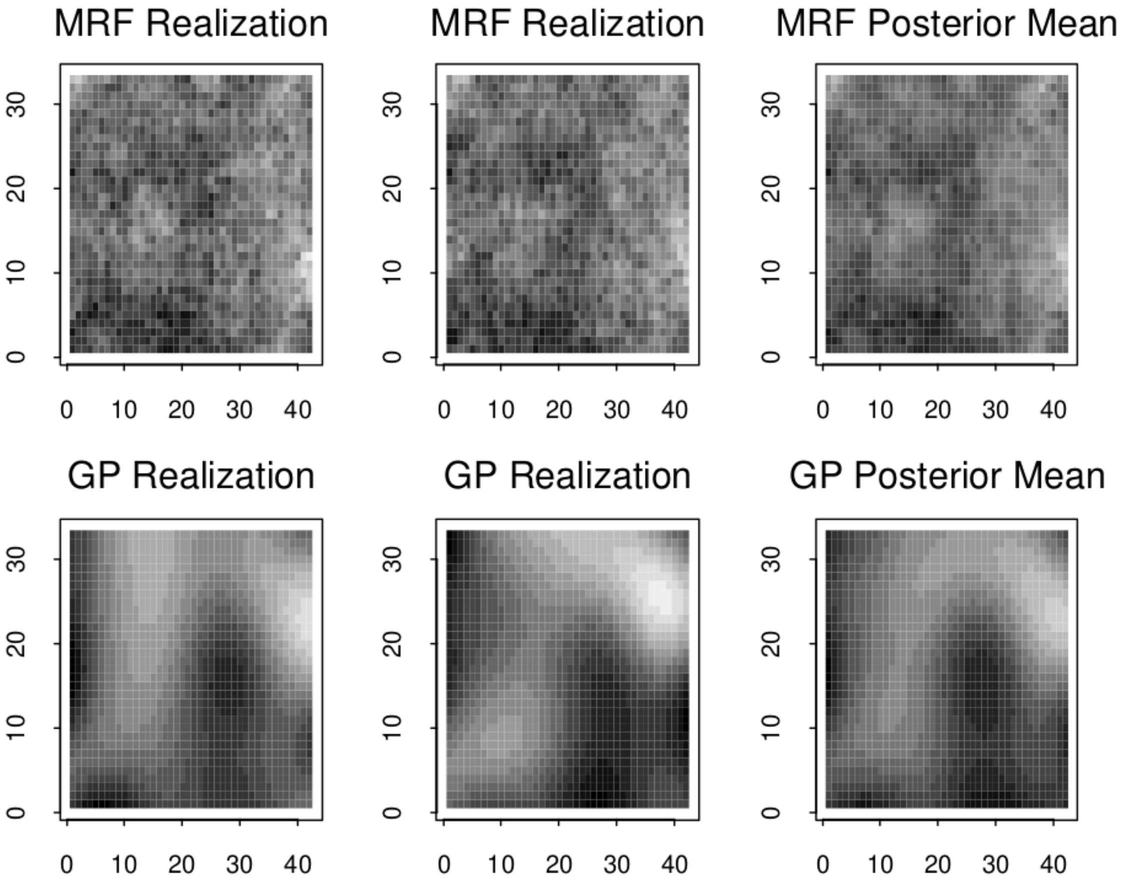
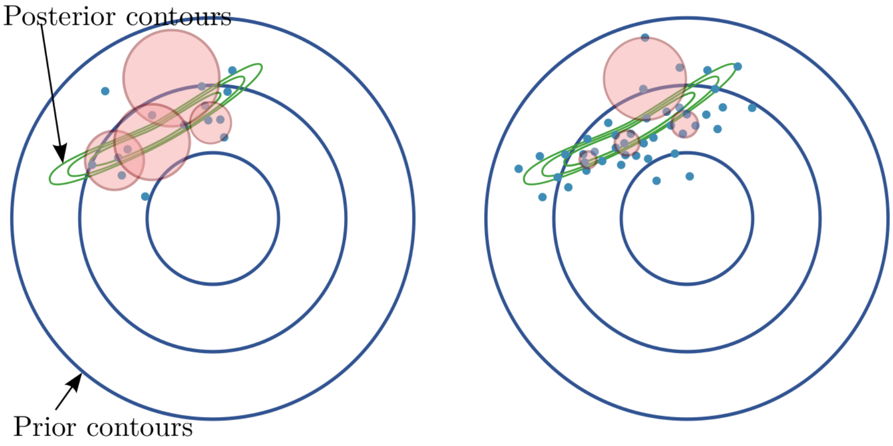


Figure 2.10: Gaussian Markov random field (MRF) realisations and the MRF posterior mean (top row), and GP realisations and the GP posterior mean (bottom row). Adapted from Higdon, Lee and Holloman (2003).

(Geyer, 1991). This approach is not unlike the differential evolution based samplers described in Section 2.2 (ter Braak, 2006), where proposals are also swapped between different MCMC samplers to improve mixing, except the population-based samplers combine the proposals of different chains, rather than just swapping, and do not generally exploit a coarse approximation. The authors also demonstrate how the MRF realisation can be projected to and from the coarse space, to allow for relatively seamless transitions when swapping proposals between levels. In Marzouk, Najm and Rahn (2007) and Marzouk and Najm (2009), the authors develop a methodology for MCMC sampling of inverse problems, in which they construct a computationally inexpensive surrogate density that is used in place of the true posterior. They create Polynomial Chaos Expansion (PCE) for the model parameters, substitute these expansions into the governing equations, and use the solution to construct the surrogate posterior density. The PCE expansion is much cheaper to evaluate than original the forward model, which allows for sampling otherwise intractable inverse problems using standard MCMC. The authors demonstrate the methodology on a simple groundwater flow problem and show that the approximate posterior density is very close to the true posterior. However, the presented examples are very simple and there is no way to validate the approximate posterior other than sampling from the true one, so the method has limited applicability for real-world problems. Another study that demonstrated a novel MCMC sampler on a groundwater flow problem was the *active subspace* MCMC described in Constantine, Kent and Bui-Thanh (2016). Here, the authors describe an algorithm, where MCMC sampling is restricted to a likelihood-informed subspace (Cui et al., 2014) while its complement is sampled using simple Monte Carlo sampling of the prior distribution. This approach is strongly related to the null-space Monte Carlo of Tonkin and Doherty (2009) but transformed into a fully Bayesian setting, where the the likelihood-informed subspace (or “calibration” subspace in PEST-terminology) is also subjected to quantification of uncertainty. While the methods are similar in execution, the goal is different. For null-space Monte Carlo, the intention is to perform uncertainty quantification for the subspace of parameters, where there is uncertainty. For active subspace

MCMC, the intention is to decrease the computational burden of running MCMC by identifying "free" parameters and offsetting the cost by using the (significantly cheaper) Monte Carlo sampling on those. The example used for demonstration was identical to the steady-state groundwater flow equation for confined aquifers, with hydraulic conductivity modelled by a KL decomposition of a GP, an almost classical example problem at this point. In Conrad et al. (2016) and Conrad et al. (2017), the authors develop an adaptive MCMC algorithm, where in place of sampling from the true posterior, a local approximate posterior distribution is iteratively constructed. The model response is modelled with linear or quadratic approximations fitted to samples of the true model response in the immediate neighbourhoods of the MCMC state and proposal, or within a "ball", as formulated by the authors (Figure 2.11).



(a) Early times.

(b) Late times.

Figure 2.11: Principle of local approximation MCMC. The blue dots show exact MCMC samples from the true model and the red balls show the neighborhoods of potential proposals, from which samples are used to approximate the model output at that proposal. As sampling progresses each neighborhood will be denser, since more exact samples are available, allowing the balls to shrink while producing more accurate local approximations. Adapted from Conrad et al. (2017).

The approximation is refined by evaluating the true model whenever the error of cross-validating the approximation exceeds a certain threshold. This is a very flexible approach, which reduces the cost of running MCMC for problems constrained by expensive forward models with very little tuning, but it does require the exact posterior to be somewhat smooth, and the forward operator to be well-conditioned, since the only metric of accuracy is a relative one. Additionally, the algorithm does not, in fact, sample from the exact posterior but from the distribution associated with the approximation of the forward model. In Conrad et al. (2016), they demonstrate the approach on a relatively low dimensional groundwater flow example, where the hydraulic conductivity is modelled using a truncated KL expansion of a GP incorporating the 6 highest energy eigenmodes. In Conrad et al. (2017), they use a different model for the hydraulic conductivity, where it is assumed to be constant within 6 predefined zones, and the objective is to find the posterior distribution within each zone. This is also a fairly low-dimensional and unrealistic problem since it is highly unlikely that zones of constant hydraulic conductivity would be fully known a priori. Finally, the groundwater flow problem was used as a benchmark problem in some recent developments of the pCN proposal, described in Section 2.2 (Cotter et al., 2013). In Beskos et al. (2017), the authors present various *geometric* extensions of the pCN proposal, all of which exploit geometric information about the posterior to sample more efficiently from non-Gaussian posteriors, and problems where the posterior deviates strongly from the prior. The pCN proposal is designed to sample efficiently from a Gaussian prior, but it will under-perform when subject to complex posteriors. This is addressed by the geometric pCN methods (Beskos et al., 2017) by incorporating the posterior gradient and Hessian in the proposals. Recently, Lan (2019) combined the geometric pCN proposals of Beskos et al. (2017) with the *Dimension Independent Likelihood Informed* (DILI) sampler of Cui, Law and Marzouk (2016). He constructs a likelihood-informed subspace, similar to the active subspace technique of Constantine, Kent and Bui-Thanh (2016) as explained above, and employs geometric pCN on that subspace, rather than the original parameter space, to reduce the effective dimensionality of the target distribution. Both Beskos et al. (2017) and Lan (2019) demonstrate their methods on groundwater flow problems with the hydraulic conductivity modelled by a KL expansion of a GP.

2.4 Final Remarks

While there are various important inverse problems associated with the field of hydrogeology, the majority of research efforts are directed towards inferring continuous geophysical properties, e.g. hydraulic conductivity, from discrete point measurements of e.g. hydraulic head and flux. This problem is inherently ill-posed, unless regularisation, dimensionality reduction or some other constraints (such as the Bayesian prior) are enforced. This challenge has triggered the development of a wealth of methods that cleverly constrain the inverse problem, such as zonation combined with Maximum Likelihood estimation, the Pilot Point Method and the hybrid regularisation approach, which combines Truncated SVD with Tikhonov regularisation. The primary disadvantage with these methods is that they introduce an array of opaque tuning parameters that are essentially subjective and can lead to artefacts in the recovered solutions. Nestling the inverse problem in the Bayesian context allows for clearly and transparently enforcing constraints through the prior distribution of parameters.

The theory of such Bayesian inverse problems is well-established, but they entail their own integral challenge, namely the cost of running MCMC for high-dimensional, large-scale inverse problems. Additionally, many Bayesian inverse problems involve a highly complex likelihood function for which the gradient cannot easily be computed. The remedy for these challenges is not trivial. The second challenge can be alleviated by using adaptive MCMC proposal distributions, improving the proposal efficiency without necessarily requiring the gradient. The first challenge is complex and is best approached by exploiting some cheaper surrogate or reduced-order model. The simplicity of defining a reduced-order model for many engineering problems makes this approach a highly attractive one. The principal techniques to rigorously achieve this are MLMCMC and DA. However, Monte Carlo samples generated by MLMCMC are generally tainted by an unidentifiable bias and DA, albeit unbiased, is a relatively inflexible approach. The difficulties associated with these existing techniques present a potential for the development of new MCMC algorithms that similarly exploit reduced-order models.

There is little evidence of the use of MCMC for inversion and uncertainty quantification in more applied sciences and the majority of the above-mentioned studies are simply using the groundwater flow problem as a well-balanced but essentially theoretical benchmark for MCMC algorithms. This discrepancy cannot be attributed only to the computational cost associated with MCMC, since many methods for reducing that already exist. I ascribe it partly to the inaccessibility of such methods, both in terms of the underlying theory, which is often presented in the form of (what appears to the average engineer as) convoluted mathematical jargon and to the lack of an easy-to-use software framework for Bayesian inverse problems. Additionally, it is not always clear from the algorithmic studies what the advantage of quantifying the uncertainty of a given problem is. Apart from the, in my opinion – obvious, potential applications with respect to environmental risk assessment, there are numerous ways that model uncertainties could be used to inform engineering decision support systems, allowing engineers to create better and more parsimonious designs.

3. Multilevel Delayed Acceptance MCMC with an Adaptive Error Model in PyMC3

This conference paper (Lykkegaard et al., 2020), presented at the Machine Learning for Engineering Modeling, Simulation, and Design Workshop (ML4ENG)[1] at the Neural Information Processing Systems (NeurIPS) 2020 conference presents a novel development in adaptive error modelling for multilevel Bayesian inverse problems. The paper also briefly discusses a new algorithm called Multilevel Delayed Acceptance (MLDA), which was explored further in the following journal paper. Please note that the ML4ENG workshop was peer–reviewed but non–archival, and hence there may be some overlap between Chapter 3 and 4.

The idea was conceived by Tim Dodwell and me. I developed the computer code in collaboration with Greg Mingas, conducted the experiments and wrote the paper. Tim Dodwell and Robert Scheichl provided feedback during the research process. All authors contributed to the editing. The paper was peer reviewed by the ML4ENG organisers.

[1] `https://ml4eng.github.io/`

Multilevel Delayed Acceptance MCMC with an Adaptive Error Model in `PyMC3`

Mikkel B. Lykkegaard
Centre for Water Systems and
Institute for Data Science and Artificial Intelligence
University of Exeter
EX4 4QF, United Kingdom
`m.lykkegaard@exeter.ac.uk`

Grigorios Mingas
The Alan Turing Institute
NW1 2DB, United Kingdom
`gmingas@turing.ac.uk`

Robert Scheichl
Institute for Applied Mathematics and
Interdisciplinary Center for Scientific Computing
Ruprecht-Karls-Universität Heidelberg
69120 Heidelberg, Germany
`r.scheichl@uni-heidelberg.de`

Colin Fox
Department of Physics
University of Otago
Dunedin 9016, New Zealand
`colin.fox@otago.ac.nz`

Tim J. Dodwell
Institute for Data Science and Artificial Intelligence
University of Exeter
EX4 4QF, United Kingdom
`t.dodwell@exeter.ac.uk`

Abstract

Uncertainty Quantification through Markov Chain Monte Carlo (MCMC) can be prohibitively expensive for target probability densities with expensive likelihood functions, for instance when the evaluation it involves solving a Partial Differential Equation (PDE), as is the case in a wide range of engineering applications. Multilevel Delayed Acceptance (MLDA) with an Adaptive Error Model (AEM) is a novel approach, which alleviates this problem by exploiting a hierarchy of models, with increasing complexity and cost, and correcting the inexpensive models on-the-fly. The method has been integrated within the open-source probabilistic programming package `PyMC3` and is available in the latest development version. In this paper, the algorithm is presented along with an illustrative example.

1 Introduction

Sampling from an unnormalised posterior distribution $\pi(\cdot)$ using Markov Chain Monte Carlo (MCMC) methods is a central task in computational statistics. This can be a particularly challenging problem when the evaluation of $\pi(\cdot)$ is computationally expensive and the parameter space θ and data $\mathbf{d}$ defining $\pi(\cdot)$ are high-dimensional. The sequential (highly) correlated nature of a Markov chain and the slow converge rates of Monte Carlo sampling, means that many MCMC samples are often required to obtain a sufficient representation of a posterior distribution $\pi(\cdot)$. Examples of such problems frequently occur in Bayesian inverse problems, image reconstruction and probabilistic machine learning, where simulations of the measurements (required to calculate a likelihood) depend on the evaluation of complex mathematical models (e.g. a system of partial differential equations) or the evaluation of prohibitively large data sets.

Workshop on machine learning for engineering modeling, simulation and design @ NeurIPS 2020

In this paper a MCMC approach capable of accelerating existing sampling methods is proposed, where a hierarchy (or sequence) $\pi_0(\cdot), \ldots, \pi_{L-1}(\cdot)$ of computationally cheaper approximations to the 'full' posterior density $\pi(\cdot) \equiv \pi_L(\cdot)$ are available. As with the original delayed acceptance algorithm, proposed by Christen and Fox [1], the idea is to generate MCMC proposals for the next step in the chain from runs of MCMC subchains targeting the computationally cheaper, approximate densities. The original DA method proposed the approach for just two levels. In this paper, the approach is extended to recursively apply delayed acceptance across a complete hierarchy of model approximations, a method termed *multilevel delayed acceptance* (MLDA). There are close connections to and similarities with multilevel variance reduction techniques, first proposed by Giles [2], widely studied for forward uncertainty propagation problems and importantly extended to Multilevel Markov Chain Monte Carlo approach by Hoang et al. [3] and Dodwell *et al.* [4], and further to a Multi-Index setting by Jasra *et al.* [5]. As in other multilevel approaches, the subchains in MLDA can be exploited for variance reduction, but this is beyond the scope of this paper.

The increase in use of Bayesian probabilistic tools has naturally coincided with the development of user-friendly computational packages, allowing users to focus on model development and testing, rather than algorithm development of sampling methods and post-processing diagnostics. Various high quality packages are available. Examples include: `MUQ`, `STAN` and `Pyro`.[1] A guiding principle of our work and of this contribution was to ensure that the MLDA implementation is easily accessible, well supported and gives flexibility to users to define complex models in a friendly language. To achieve this we embed our sampler into the widely used open-source probabilistic programming package `PyMC3` [6]. The method and implementation have been accepted in the development version, and will be made available with the next full release (version 3.9.4).

2 Adaptive Multilevel Delayed Acceptance (MLDA)

2.1 Preliminaries: Metropolis-Hastings MCMC Algorithms

Here, a typical Bayesian inverse problem is considered. Given are (limited) observations $d \in \mathbb{R}^M$ of a system and a mathematical model $\mathcal{F}(\theta) : \mathbb{R}^R \mapsto \mathbb{R}^M$, which maps from a set of model parameters $\theta \in \mathbb{R}^R$ to the space of model predictions of the data. The connection between model and data is then, in the simplest case, described by the additive model

$$d = \mathcal{F}(\theta) + \epsilon \tag{1}$$

(but it can also be more general). Here, ϵ is a random variable, which can depend on θ and captures the uncertainty of the model's reproduction of the data. It might include measurement uncertainty of the recorded data, uncertainty due to model mis-specification and/or uncertainties due to sing in practice a numerical approximation of the mathematical model. The distribution of the random variable ϵ defines the *likelihood*, i.e. the probability distribution $\mathcal{L}(d|\theta)$. For simplicity it is assumed to be Gaussian, i.e. $\epsilon \sim \mathcal{N}(\mu_\epsilon, \Sigma_\epsilon)$ and $\mathcal{L}(d|\theta) \sim \mathcal{N}(d - \mathcal{F}(\theta) - \mu_\epsilon, \Sigma_\epsilon)$, but it does not have to be.

Given *prior* information $\pi(\theta)$ on the distribution of the model parameters θ, the aim is to condition this distribution on the observations, i.e. to obtain samples from the *posterior* distribution $\pi(\theta|\mathbf{d})$. Through Bayes' theorem, it follows that

$$\pi(\theta|d) = \frac{\mathcal{L}(d|\theta)\pi(\theta)}{\pi(d)} \propto \mathcal{L}(d|\theta)\pi(\theta). \tag{2}$$

Since the normalising constant $\pi(d)$ (the *evidence*) is not typically known, the conditional distribution $\pi(\theta|d)$ is generally intractable and exact sampling is not possible. There are various computational strategies for generating samples from $\pi(\theta|d)$. This paper focuses on the Metropolis-Hastings MCMC algorithm, described in Algorithm 1. It creates a Markov chain $\{\theta^j\}_{j\in\mathbb{N}}$ of correlated parameter states θ^j that (in the limit) target the exact posterior distribution $\pi(\theta|d)$ (cf. e.g. [7]). The efficiency of the algorithm is determined by the choice of the proposal distribution $q(\cdot|\cdot)$.

Whilst MCMC methods are the gold-standard for sampling from complex posterior distributions, for many types of models and data they come with significant practical challenges. Firstly, each cycle of Alg. 1 requires the evaluation of the model $\mathcal{F}(\theta')$ which may be computationally very expensive. Secondly, the samples generated in the chain are correlated, and therefore many cycles of Alg. 1 are

[1] `MUQ`: http://muq.mit.edu, `STAN`: https://mc-stan.org, `Pyro`: https://pyro.ai

Algorithm 1 (Metropolis-Hastings MCMC): Choose θ^0. Then, for $j = 0, \dots, J-1$:

1. Given θ^j, generate a proposal θ' from a given proposal distribution $q(\theta'|\theta^j)$,
2. Accept proposal θ' as the next sample with probability

$$\alpha(\theta'|\theta^j) = \min\left\{1, \frac{\mathcal{L}(d|\theta')\,\pi(\theta')\,q(\theta^j|\theta')}{\mathcal{L}(d|\theta^j)\pi(\theta^j)q(\theta'|\theta^j)}\right\},$$

i.e. set $\theta^{j+1} = \theta'$ with probability α, and $\theta^{j+1} = \theta^j$ with probability $1 - \alpha$.

often required to produce a sufficient number of ”independent” (or *effective*) samples from $\pi(\theta|\mathbf{d})$. The ideal proposal distribution generates cheap candidate proposals θ', which have a high probability of being accepted, and are independent of the previous sample θ^j.

In this paper, efficient, Metropolis-style proposal strategies are developed that exploit a hierarchy of approximations $\mathcal{F}_\ell(\theta)$, for $\ell = 0, \dots, L-1$, to the full model $\mathcal{F}_L := \mathcal{F}$, which are assumed to be ordered according to increasing accuracy and computational cost.

2.2 Multilevel Delayed Acceptance

Delayed Acceptance (DA) is an approach first introduced by Christen and Fox [1], exploiting a simple, but highly effective idea. The original DA approach is a two-level method that assumes a computationally cheaper approximation $\mathcal{F}^*$ for the forward map $\mathcal{F}$ is available. The idea is that for any chosen proposal θ', a standard Metropolis accept/reject step (as given in Alg. 1) is performed with the approximate forward map $\mathcal{F}^*(\theta')$ before the expensive forward model $\mathcal{F}(\theta')$ is evaluated. Only if accepted, a second accept/reject step with the original forward map $\mathcal{F}(\theta')$ and with acceptance probability $\alpha = \min\left\{1, \frac{\mathcal{L}(d|\theta')\mathcal{L}^*(d|\theta^j)}{\mathcal{L}(d|\theta^j)\mathcal{L}^*(d|\theta')}\right\}$ is carried out. Here, $\mathcal{L}^*(d|\cdot)$ denotes the posterior distribution with the likelihood defined by $\mathcal{F}^*$. The validity of this approach as a proposal method, yielding a convergent MCMC algorithm, is provided in [1].

The basic DA approach can be extended in two ways. First, instead of doing a single check for the proposal that comes from the fine level, a subchain of length J can be ran on the coarse level [8, 9]. This does not affect the theory, but has the advantage of decorrelating samples passed back as proposals to the fine level. Second, and this is the main, novel algorithmic contribution, DA is extended to a general multilevel setting, exploiting links to the Multilevel Markov Chain Monte Carlo (MLMCMC) Method proposed by Dodwell *et al.* [4].

The subtle differences between the approaches are apparent when comparing the schematics of the two multilevel proposal processes shown in Fig. 1. Algorithmically, Multilevel Delayed Acceptance (MLDA) can be seen as a recursion of Delayed Acceptance over multiple levels $\ell = \{0, 1, \dots, L\}$. Crucially, if θ^i_ℓ is the current state at level ℓ, and a proposal θ' from the coarse subchain on level $\ell - 1$ is rejected at level ℓ, the coarse subchain to generate the subsequent proposal for level ℓ is again initiated from θ^i_ℓ. For MLMCMC, even if the coarse proposal is rejected, the coarse chain continues independently of the fine chain and does not revert to the state θ^i_ℓ (see Fig. 1, right). As a result, coarse and fine chains will detach, and only align once a coarse proposal is accepted at the fine level.

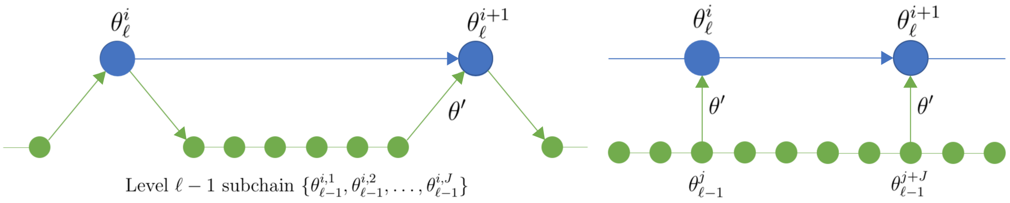


Figure 1: Schematic for generating a proposal θ' on level ℓ in MLDA (left) and in MLMCMC (right).

The new MLDA algorithm with subchain length $J_\ell \in \mathbb{N}$ on level $0 \le \ell < L$ is described in Algorithm 2.

Algorithm 2 (Multilevel Delayed Acceptance MCMC):
Choose θ^0 and set the states of all subchains $\theta_0^0 = \ldots = \theta_{L-1}^0 = \theta^0$. Then, for $j = 0, \ldots, J-1$:

1. Given θ^j and $\theta_\ell^{j_\ell}$ such that $j_\ell < J_\ell$ for all $1 \leq \ell < L$, generate a subchain of length J_0 with Alg. 1 on level 0, starting from $\theta_0^0 = \theta_1^{j_1}$ and using the transition kernel $q(\theta_0'|\theta_0^{j_1})$.
2. Let $\ell = 1$ and $\theta_1' = \theta_0^{J_0}$.
3. If $\ell = L$ go to Step 7. Otherwise compute the delayed acceptance probability on level ℓ, i.e.,
$$\alpha_\ell = \min\left\{1, \frac{\mathcal{L}_\ell(d|\theta_\ell')\,\mathcal{L}_{\ell-1}(d|\theta_\ell^{j_\ell})}{\mathcal{L}_\ell(d|\theta_\ell^{j_\ell})\mathcal{L}_{\ell-1}(d|\theta_\ell')}\right\}.$$
4. Set $\theta_\ell^{j_\ell+1} = \theta_\ell'$ with probability α_ℓ and $\theta_\ell^{j_\ell+1} = \theta_\ell^{j_\ell}$ otherwise. Increment $j_\ell \to j_\ell + 1$.
5. If $j_\ell = J_\ell$ set $\theta_{\ell+1}' = \theta_\ell^{J_\ell}$, increment $\ell \to \ell + 1$ and return to Step 3.
6. Otherwise set $j_k = 0$ and $\theta_k^0 = \theta_\ell^{j_\ell}$, for all $0 \leq k < \ell$, and return to Step 1.
7. Compute the delayed acceptance probability on level L, i.e.,
$$\alpha_L = \min\left\{1, \frac{\mathcal{L}_\ell(d|\theta_L')\mathcal{L}_{\ell-1}(d|\theta^j)}{\mathcal{L}_\ell(d|\theta^j)\,\mathcal{L}_{\ell-1}(d|\theta_L')}\right\}.$$
Set $\theta^{j+1} = \theta_L'$ with probability α_L and $\theta^{j+1} = \theta^j$ otherwise. Increment $j \to j + 1$.
8. Set $j_\ell = 0$ and $\theta_\ell^0 = \theta^j$, for all $0 \leq \ell < L$, and return to Step 1.

2.3 Adaptive correction of the approximate posteriors

While the approach outlined above does guarantee sampling from the exact posterior, there are situations when convergence can be prohibitively slow. When the model approximation is poor, the delayed acceptance probability is low, and many proposals are rejected. This will result in suboptimal acceptance rates and low effective sample sizes. The leftmost panel in Fig. 2 shows a contrived example, where the approximate likelihoods (red/orange isolines) are offset from the likelihood on the finest level (blue contours) and their scales, shapes and orientations are incorrect. Thus, as an additional modification, an Adaptive Error Model (AEM) is introduced to account for discrepancies between model levels.

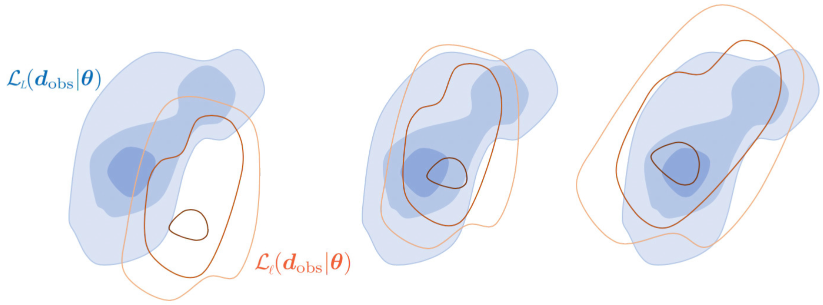

$\mathcal{L}_L(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta})$
$\mathcal{L}_\ell(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta})$

Figure 2: Effect of applying the Gaussian Adaptive Error Model (AEM). The first panel shows the initial state before adaptation, where the coarse likelihoods $\mathcal{L}_\ell(d|\theta)$ (red/orange isolines) approximate the fine likelihood $\mathcal{L}_L(d|\theta)$ (blue contours) poorly. The second panel shows the effect of shifting the likelihoods by the mean of the bias. The third panel shows the effect of additionaly incorporating estimates of the covariance of the bias. (Adapted from [9].)

Let $\mathcal{F}_\ell$ denote a coarse forward map of level ℓ and $\mathcal{F}_L$ denote the forward map on the finest level L. To obtain a better approximation of the data d using $\mathcal{F}_\ell$, the two-level AEM suggested in [10, 11] and analysed in [12] is extended by adding a telescopic sum of the differences in the forward model output across all levels from ℓ to L:

$$d = \mathcal{F}_L(\theta) + \epsilon = \mathcal{F}_\ell(\theta) + \mathcal{B}_\ell(\theta) + \epsilon \quad \text{with} \quad \mathcal{B}_\ell(\theta) := \sum_{k=\ell}^{L-1} \underbrace{\mathcal{F}_{k+1}(\theta) - \mathcal{F}_k(\theta)}_{:=B_k(\theta)}, \tag{3}$$

denoting the bias on level ℓ at θ. The trick in the context of MLDA is that, since $\mathcal{B}_\ell$ is just a simple sum, the individual bias terms B_k from pairs of adjacent model levels can be estimated independently, so that new information can be exploited each time *any* set of adjacent levels are evaluated for the same parameter value θ. Approximating each individual bias term $B_k = \mathcal{F}_{k+1} - \mathcal{F}_k$ with a multivariate Gaussian $B_k^* \sim \mathcal{N}(\mu_k, \Sigma_k)$, the total bias $\mathcal{B}_\ell$ can be approximated by the Gaussian $\mathcal{B}_\ell^* \sim \mathcal{N}(\mu_{\mathcal{B},\ell}, \Sigma_{\mathcal{B},\ell})$ with $\mu_{\mathcal{B},\ell} = \sum_k \mu_k$ and $\Sigma_{\mathcal{B},\ell} = \sum_k \Sigma_k$.

The bias-corrected likelihood function for level ℓ is then proportional to

$$\mathcal{L}_\ell^*(d|\theta) \propto \exp\left(-\frac{1}{2}\big(d - \mathcal{F}_\ell(\theta) - \mu_\epsilon - \mu_{\mathcal{B},\ell}\big)^T \big(\Sigma_\epsilon + \Sigma_{\mathcal{B},\ell}\big)^{-1} \big(d - \mathcal{F}_\ell(\theta) - \mu_\epsilon - \mu_{\mathcal{B},\ell}\big)\right). \tag{4}$$

One way to construct the AEM is offline, by sampling from the prior before running the MCMC, as suggested in [10]. However, this approach requires a significant overhead prior to sampling, and may result in a suboptimal error model, since the bias in the posterior may differ substantially from the bias in the prior. Instead, as suggested by [11], an estimate for the B_k can be constructed iteratively during sampling, using the following recursive formulae for sample mean and sample covariance [13]:

$$\mu_{k,i+1} = \frac{1}{i+1}\Big(i\mu_{k,i} + B_k(\theta^{i+1})\Big) \quad \text{and} \tag{5}$$

$$\Sigma_{k,i+1} = \frac{i-1}{i}\Sigma_{k,i} + \frac{1}{i}\Big(i\mu_{k,i}\,\mu_{k,i}^T - (i+1)\mu_{k,i+1}\,\mu_{k,i+1}^T + B_k(\theta^{i+1})\,B_k(\theta^{i+1})^T\Big) \tag{6}$$

While this approach in theory compromises ergodicity in the strict sense, the recursively constructed sample moments exhibit *diminishing adaptation* [13].

3 Implementation and Demonstration

The Multilevel Delayed Acceptance MCMC algorithm (Alg. 2) has been implemented in PyMC3 [6], an open-source probabilistic programming package for Python built on top of the Theano library [14]. The code is available in the development version of PyMC3.[2]. In the following section, we present a numerical experiment, in which we compare the "vanilla" MLDA sampler to the AEM-activated MLDA sampler. To demonstrate the effect of the AEM, we have chosen models of very low resolution on the coarse levels. It is important to stress, however, that the AEM is not a strict requirement for MLDA in cases, where the coarse models are better approximations of the fine.

3.1 Example: Estimation of Soil Permeability in Subsurface Flow

In this example, a simple model problem arising in subsurface flow modelling is considered. Probabilistic uncertainty quantification is of interest in various situations, for example in risk assessment of radioactive waste repositories. Moreover, this simple PDE model is often used as a benchmark for MCMC algorithms in the applied mathematics literature. The classical equations which govern steady-state single-phase subsurface flow are Darcy's law coupled with an incompressibility constraint

$$w + k\nabla p = g \quad \text{and} \quad \nabla \cdot w = 0, \quad \text{in} \quad D \subset \mathbb{R}^d \tag{7}$$

for $d = 1, 2$ or 3, subject to suitable boundary conditions. Here p denotes the hydraulic head of the fluid, k the permeability tensor, w the flux and g is the source term.

A typical approach to treat the inherent uncertainty in this problem is to model the permeability as a random field $k = k(x, \omega)$ on $D \times \Omega$, for some probability space $(\Omega, \mathcal{A}, \mathbb{P})$. Therefore, (7) can be written as the following PDE with random coefficients:

$$-\nabla \cdot k(x,\omega)\nabla p(x,\omega) = f(x), \quad \text{for all} \quad x \in D, \tag{8}$$

[2] https://github.com/pymc-devs/pymc3

where $f := -\nabla \cdot g$. As a synthetic example, consider the domain $D := [0,1]^2$ with $f \equiv 0$ and deterministic boundary conditions

$$p|_{x_1=0} = 0, \quad p|_{x_1=1} = 1 \quad \text{and} \quad \partial_n p|_{x_2=0} = \partial_n p|_{x_2=1} = 0. \tag{9}$$

A widely used model for the prior distribution of the permeability in hydrology is a log-Gaussian random field, characterised by the mean of $\log k$, here chosen to be 0, and by its covariance function, here chosen to be

$$C(x,y) := \sigma^2 \exp\left(-\frac{\|x-y\|_2^2}{2\lambda^2}\right), \quad \text{for} \quad x, y \in D, \tag{10}$$

with $\sigma = 2$ and $\lambda = 0.3$. The log-Gaussian random field is parametrised using a truncated Karhunen-Loève (KL) expansion of $\log k$, i.e., an expansion in terms of a finite set of independent, standard Gaussian random variables $\theta_i \sim \mathcal{N}(0,1)$, $i = 1, \ldots, R$, given by

$$\log k(x,\omega) = \sum_{i=1}^{R} \sqrt{\mu_i}\phi_i(x)\theta_i(\omega). \tag{11}$$

Here, $\{\mu_i\}_{i\in\mathbb{N}}$ are the sequence of strictly decreasing real, positive eigenvalues, and $\{\phi_i\}_{i\in\mathbb{N}}$ the corresponding L^2-orthonormal eigenfunctions of the covariance operator with kernel $C(x,y)$. Thus, the prior distribution on the parameter $\theta = (\theta_i)_{i=1}^R$ in the stochastic PDE problem (8) is $\mathcal{N}(0, I_R)$.

The aim is to infer the posterior distribution of θ, conditioned on measurements of p at $M = 25$ discrete locations $x^j \in D$, $j = 1, \ldots, M$, stored in the vector $d_{obs} \in \mathbb{R}^M$. Thus, the forward operator is $\mathcal{F} : \mathbb{R}^R \to \mathbb{R}^M$ with $\mathcal{F}_j(\theta_\omega) = p(x^j, \omega)$.

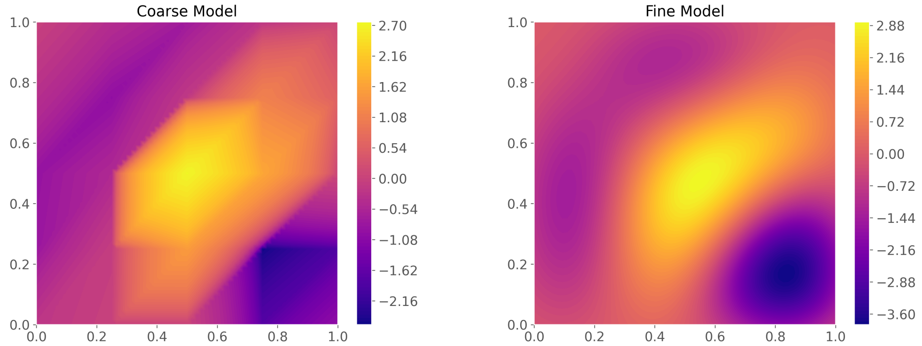


Figure 3: True log-conductivity field of the coarsest model with m_0 grid points (left) and the finest model with m_2 grid points (right).

All finite element (FE) calculations were carried out with `FEniCS` [15], using piecewise linear FEs on a uniform triangular mesh. The coarsest mesh $\mathcal{T}_0$ consisted of $m_0 = 5$ grid points in each direction, while subsequent levels were constructed by two steps of uniform refinement of $\mathcal{T}_0$, leading to $m_\ell = 4^\ell(m_0 - 1) + 1$ grid points in each direction on the three grids $\mathcal{T}_\ell$, $\ell = 0, 1, 2$ (Fig. 3).

To demonstrate the excellent performance of MLDA with the AEM, synthetic data was generated by drawing a sample θ^{ex} from the prior distribution and solving (8) with the resulting realisation of k on $\mathcal{T}_2$. To construct d_{obs}, the computed discrete hydraulic head values at $(x^j)_{j=1}^M$, were then perturbed by independent Gaussian random variables, i.e. by a sample $\epsilon^* \sim \mathcal{N}(0, \Sigma_\epsilon)$ with $\Sigma_\epsilon = 0.01^2 I_M$.

To compare the "vanilla" MLDA approach to the AEM-enhanced version, we sampled the same model using identical sampling parameters, with and without AEM activated. For each approach, we sampled four independent chains, each initialised at a random point from the prior. For each independent chain, we drew 5000 samples plus a burn-in of 2000. We used subchain lengths $J_0 = J_1 = 5$, since that produced the best trade-off between computation time and effective sample size for MLDA with the AEM. Note that the cost of computing the subchains on the coarser levels only leads to about a 50% increase in the total cost for drawing a sample on level L. The `PyMC3` non-blocked Random Walk Metropolis Hastings (RWMH) sampler was employed on the coarsest level with automatic step-size tuning during burn-in to achieve an acceptance rate between 0.2 and 0.5. All other sampling parameters were maintained at the default setting of the `MLDA` method.

To assess the performance of the two approaches the Effective Sample Size (ESS) for each parameter was computed [16]. Since the coarsest model was quite a poor approximation of the finest, running MLDA without the Adaptive Error Model (AEM), yielded very poor results. None of the four chains converged, there was poor mixing, a sub optimal acceptance rate of 0.019 on level L, and an ESS of 4 out of 20000 samples, meaning that each independent chain was only capable of producing a single independent sample. When the AEM was employed and otherwise using the exact same sampling parameters, we observed convergence for every chain, good mixing, an acceptance rate of 0.66 on level L and an ESS of 3319 out of 20000 samples (Fig. 4). In comparison, a single-level non-blocked RWMH sampler on grid $\mathcal{T}_2$ with automatic step-size tuning during burn-in produced an ESS of 19 out of 5000 samples with an acceptance rate of 0.26.

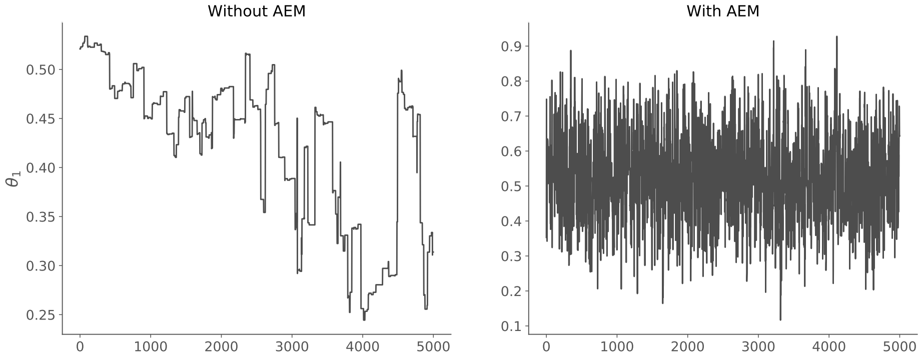


Figure 4: Traces of θ_1 on level $\ell = 2$, for MLDA without (left) and with AEM (right).

Note that the particular numerical experiment was chosen to demonstrate the dramatic effect that employing the AEM can have in MLDA. Thus, making it possible to use multilevel sampling strategies with very crude approximate models. A FE mesh with 25 degrees of freedom is extremely coarse for a Gaussian random field with correlation length $\lambda = 0.3$, yet using the AEM it still provides an excellent surrogate for delayed acceptance. Typically much finer models are used in real applications with longer subchains on the coarser levels (cf. [4]). The AEM will be less critical in that case and MLDA will also produce good ESS without the AEM. In a future journal paper, this topic will be carefully studied along with a comparison with other samplers on the finest level and an analysis of the multilevel variance reduction capabilities of MLDA.

Broader Impact

This research has the potential to make unbiased uncertainty quantification of expensive models available to a greater audience, including engineers employed in risk assessment and reliability engineering. Since many engineering problems involve solving PDEs, multi-level hierarchies can easily be introduced using grid refinement, making this method exceptionally well suited for engineering applications.

Acknowledgements

The work was funded by a Turing AI fellowship (2TAFFP\100007) and the Water Informatics Science and Engineering Centre for Doctoral Training (WISE CDT) under a grant from the Engineering and Physical Sciences Research Council (EPSRC), grant number EP/L016214/1.

References

[1] J. A. Christen and C. Fox. Markov chain Monte Carlo using an approximation. *J. Comput. Graph. Stat.*, 14(4):795–810, 2005.

[2] M. B. Giles. Multilevel Monte Carlo path simulation. *Oper. Res.*, 56(3):607–617, 2008.

[3] V. H. Hoang, C. Schwab, and A. M. Stuart. Complexity analysis of accelerated MCMC methods for Bayesian inversion. *Inverse Probl.*, 29(8):085010, 2013.

[4] T. J. Dodwell, C. Ketelsen, R. Scheichl, and A. L. Teckentrup. A hierarchical multilevel Markov chain Monte Carlo algorithm with applications to uncertainty quantification in subsurface flow. *SIAM/ASA J. Uncertain. Q.*, 3(1):1075–1108, 2015.

[5] A. Jasra, K. Kamatani, K. Law, and Y. Zhou. A multi-index Markov chain Monte Carlo method. *Int. J. Uncertain. Quant.*, 8(1):61–73, 2018.

[6] J. Salvatier, T. V. Wiecki, and C. Fonnesbeck. Probabilistic programming in Python using PyMC3. *PeerJ. Comput. Sci.*, 2:e55, 2016.

[7] Gareth O. Roberts and Jeffrey S. Rosenthal. General state space Markov chains and MCMC algorithms. *Probability Surveys*, 1(0):20–71, 2004.

[8] J. S. Liu. *Monte Carlo Strategies in Scientific Comuputing*. Springer, New York, 2004.

[9] M. B. Lykkegaard, T. J. Dodwell, and D. Moxey. Accelerating uncertainty quantification of groundwater flow modelling using deep neural networks. *arXiv:2007.00400*, 2020. Manuscript submitted for publication.

[10] J. Kaipio and E. Somersalo. Statistical inverse problems: Discretization, model reduction and inverse crimes. *J. Comput. Appl. Math.*, 198(2):493–504, 2007.

[11] T. Cui, C. Fox, and M. J. O'Sullivan. Bayesian calibration of a large-scale geothermal reservoir model by a new adaptive delayed acceptance Metropolis Hastings algorithm. *Water. Resour. Res.*, 47:W10521, 2011.

[12] T. Cui, C. Fox, and M. J. O'Sullivan. A posteriori stochastic correction of reduced models in delayed-acceptance MCMC, with application to multiphase subsurface inverse problems. *Int. J. Numer. Meth. Eng.*, 118(10):578–605, 2019.

[13] H. Haario, E. Saksman, and J. Tamminen. An adaptive Metropolis algorithm. *Bernoulli*, 7(2):223, 2001.

[14] Theano Development Team: Rami Al-Rfou et al. Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*, abs/1605.02688, 2016.

[15] H. P. Langtangen and A. Logg. *Solving PDEs in Python – The FEniCS tutorial Volume I*. Simula SpringerBriefs on Computing. Springer International Publishing, 2017.

[16] Aki Vehtari, Andrew Gelman, Daniel Simpson, Bob Carpenter, and Paul-Christian Bürkner. Rank-normalization, folding, and localization: An improved `$\widehat{R}$` for assessing convergence of MCMC. *arXiv:1903.08008 [stat]*, May 2020. arXiv: 1903.08008.

4. Multilevel Delayed Acceptance MCMC

This journal paper, currently under review in the SIAM/ASA Journal on Uncertainty Quantification, further develops the Multilevel Delayed Acceptance (MLDA) algorithm outlined in the previous conference paper. This paper includes a theoretical section where we prove detailed balance of the algorithm, an outline of how to exploit variance reduction with MLDA, the theory of a multilevel adaptive error model, and three examples demonstrating the various features of MLDA, including a gravity surveying problem, a predator-prey model and a groundwater flow problem.

The idea was conceived my Tim Dodwell and me. I developed the computer code in collaboration with Greg Mingas. I conceived of the examples and completed the experiments. I wrote the text with contributions from Tim Dodwell and Colin Fox. Tim Dodwell, Robert Scheichl and Colin Fox provided feedback during the research process. All authors contributed to the editing.

Multilevel Delayed Acceptance MCMC

A Preprint

Mikkel B. Lykkegaard[*]
Centre for Water Systems and
Institute for Data Science and AI
University of Exeter
EX4 4QF, United Kingdom
m.lykkegaard@exeter.ac.uk

Tim J. Dodwell
The Alan Turing Institute and
Institute for Data Science and AI
University of Exeter
EX4 4QF, United Kingdom
t.dodwell@exeter.ac.uk

Colin Fox
Department of Physics
University of Otago
Dunedin 9016, New Zealand
colin.fox@otago.ac.nz

Grigorios Mingas
The Alan Turing Institute
British Library, 96 Euston Road
NW1 2DB, United Kingdom
gmingas@turing.ac.uk

Robert Scheichl
Institute for Applied Mathematics and
Interdisciplinary Center for Scientific Computing
Heidelberg University
69120 Heidelberg, Germany
r.scheichl@uni-heidelberg.de

February 8, 2022

Abstract

We develop a novel Markov chain Monte Carlo (MCMC) method that exploits a hierarchy of models of increasing complexity to efficiently generate samples from an unnormalized target distribution. Broadly, the method rewrites the Multilevel MCMC approach of Dodwell *et al.* (2015) in terms of the Delayed Acceptance (DA) MCMC of Christen & Fox (2005). In particular, DA is extended to use a hierarchy of models of arbitrary depth, and allow subchains of arbitrary length. We show that the algorithm satisfies detailed balance, hence is ergodic for the target distribution. Furthermore, multilevel variance reduction is derived that exploits the multiple levels and subchains, and an adaptive multilevel correction to coarse-level biases is developed. Three numerical examples of Bayesian inverse problems are presented that demonstrate the advantages of these novel methods. The software and examples are available in `PyMC3`.

***K*eywords** Markov chain Monte Carlo · Bayesian Inverse Problems · Multilevel Methods · Model Hierarchies · Detailed Balance · Variance Reduction · Adaptive error model

[*] Corresponding author.

1 Introduction

Sampling from an unnormalised posterior distribution $\pi(\cdot)$ using Markov Chain Monte Carlo (MCMC) methods is a central task in computational statistics. This can be a particularly challenging problem when the evaluation of $\pi(\cdot)$ is computationally expensive and the parameters θ and/or data $\mathbf{d}$ defining $\pi(\cdot)$ are high-dimensional. The sequential (highly) correlated nature of a Markov chain and the slow converge rates of MCMC sampling, means that often many MCMC samples are required to obtain a sufficient representation of a posterior distribution $\pi(\cdot)$. Examples of such challenging problems frequently occur in Bayesian inverse problems, image reconstruction and probabilistic machine learning, where simulations of the measurements (required to calculate a likelihood function) depend on the evaluation of complex mathematical models (e.g. a system of partial differential equations) or the evaluation of prohibitively large data sets.

The topic of MCMC methods is a rich and active field of research. While the basic idea of the original Metropolis–Hastings algorithm [36, 24] is almost embarrassingly simple, it has given rise to a wide variety of algorithms tailored to different applications. Most notably, the Gibbs sampler [18], which samples each variable conditional on the other variables, the Metropolis Adjusted Langevin Algorithm (MALA, [42, 38]), Hamiltonian Monte Carlo (HMC, [16]) and the No-U-Turn Sampler (NUTS, [26]), which all exploit gradient information to improve the MCMC proposals. We would also like to highlight the seminal work of Haario et al. [21] on the Adaptive Metropolis sampler that launched a new paradigm of adaptive MCMC algorithms (see e.g. [2, 1, 41, 49, 50, 14]).

The most efficient MCMC methods cheaply generate candidate proposals, which have a high probability of being accepted, whilst being almost independent from the previous sample. In this paper, we define a MCMC approach capable of accelerating existing sampling methods, where a hierarchy (or sequence) $\pi_0(\cdot), \dots, \pi_{L-1}(\cdot)$ of computationally cheaper approximations to the exact posterior density $\pi(\cdot) \equiv \pi_L(\cdot)$ are available. As with the original Delayed Acceptance (DA) algorithm, proposed by Christen and Fox [8], short runs of MCMC subchains, generated using a computationally cheaper, approximate density $\pi_{\ell-1}(\cdot)$, are used to generate proposals for the Markov chain targeting $\pi_\ell(\cdot)$. The original DA method formulated the approach for just two levels and a single step on the coarse level. In this paper we extend the method by recursively applying DA across a hierarchy of model approximations for an arbitrary number of steps on the coarse levels – a method we term *Multilevel Delayed Acceptance* (MLDA). There are clear similarities with Multilevel Monte Carlo sampling methods, first proposed by Heinrich [25] and later by Giles [19], which have been widely studied for forward uncertainty propagation problems (see e.g. [9, 4, 7, 46]) and importantly have been extended to Bayesian inverse problems in the Multilevel Markov Chain Monte Carlo (MLMCMC) approach by Dodwell *et al.* [15] as well as to the Multi-Index setting [22, 27].

The fundamental idea of multilevel methods is simple: We let the cheaper (or *coarse*) model(s) do most of the work. In the context of sampling, be it Monte Carlo or MCMC, this entails drawing more samples on the coarser levels than on the finer, and use the entirety of samples across all model levels to improve our Monte Carlo estimates. Additionally, in the context of MCMC, the samplers on the coarser levels inform the samplers on the finer levels by filtering out poor MCMC proposals, effectively boosting the acceptance rate and hence computational efficiency on the finer levels.

While applying the multilevel idea to Monte Carlo sampling is straightforward, there are certain complications in the context of MCMC. The challenges are two-fold. Firstly, MCMC is inherently sequential and, following the Delayed Acceptance approach, proposals must be evaluated on the coarse levels before being passed to the fine, which precludes parallelisation across levels. Secondly, depending on the "quality" of the coarse models, MCMC proposals may not always be filtered well. The multilevel MCMC algorithm of Dodwell *et al.* [15] overcomes the first challenge by letting the coarser levels sample independently of the finer but at the expense of not being Markov (see Sec. 2.2). The sampler presented in this paper is Markov, so it differs from MLMCMC [15]. The second challenge can be addressed by using adequate coarse models, but the effect may also be alleviated by introducing an error model that accounts for discrepancies between levels. The error models introduced in [28] were constructed offline using samples from the prior. The sampler presented in this paper uses the *a posteriori* error model introduced in [13], constructed online and adaptively, which is much more efficient; see [14, 17] for comparisons of *a priori* and *a posteriori* error models.

In the following section we present the MLDA algorithm. We prove detailed balance of MLDA, working through each of the constituent elements separately. In this process, we develop two additional algorithms, namely *Randomised-Length-Subchain Surrogate Transition* (RST) and *Two Level Delayed Acceptance* (TLDA), each of which are valid MCMC samplers in their own respect. We extend the MLDA algorithm (1) by showing that we can achieve multilevel variance reduction with respect to some quantity of interest (as for MLMCMC), and (2) by developing a multilevel error approximation model to adaptively correct the coarse level biases at runtime based on samples. In Section 3, we demonstrate the algorithm using three examples of different Bayesian inverse problems. First, we show that extended subchains on the coarse level can significantly increase the effective sample size compared to an equivalent single-level sampler on the fine level, using an example from gravitational surveying. Second, we demonstrate multilevel variance reduction on a predator-prey model, where coarse models are constructed by restricting the length of the time window over which the differential equation model is fitted to data. Third, we demonstrate the multilevel error model in the context of a subsurface flow problem. We show that when we utilize the error model, we can achieve high effective sample sizes on the finest level, even when a very crude approximation is employed as the coarsest model.

2 Multilevel Delayed Acceptance

In this section we first outline the theoretical foundations of vanilla Metropolis–Hastings based MCMC [36, 24] and the Delayed Acceptance (DA) method proposed by Christen and Fox [8]. We extend DA in two ways: horizontally, by allowing the coarse sampler to construct subchains of multiple coarse samples before proposing a sample on the fine level; and vertically, by recursively using DA on an entire hierarchy of models with increasing resolution/accuracy. This constitutes the Multilevel Delayed Acceptance (MLDA) sampler. From this we propose two extensions which each give significant gains in statistical efficiency, namely multilevel variance reduction and a multilevel adaptive error model – the inclusion of these extensions leads us to the *Adaptive* Multilevel Delayed Acceptance sampler demonstrated in Section 3.

2.1 Convergence of MLDA and Detailed Balance

We will show that MLDA correctly generates samples from the unnormalised target density $\pi(\cdot)$ by building on standard ergodicity results for Markov chains (see [39] and references therein). Each algorithm considered here defines a stochastic iteration on a well-defined state, so defines a Markov chain. Hence, we can apply classical ergodic theorems for Markov chains.

The ergodic theorems for Markov chains (see [39] and references therein) state that the chain is π-ergodic if the chain is π-irreducible, aperiodic, and reversible with respect to π. Essentially, irreducibility and aperiodicity guarantee that the Markov chain has a unique equilibrium distribution, while reversible with respect to π ensures that π is the density of that unique distribution. The condition of π-irreducibility is satisfied when the proposal distribution is chosen such that the standard Metropolis–Hasting algorithm is π-irreducible. For algorithms based on delayed acceptance, it is also necessary that the coarse-level approximation is chosen to maintain irreducibility; see [8, Thm. 1] for precise conditions on the approximation. Aperiodicity is a mild condition that is satisfied by any Metropolis–Hastings algorithm with a non-zero probability of rejection on any π-positive set; again see [8, Thm. 1]. We will assume that the proposal and approximations are chosen so that these conditions hold. Accordingly, we focus on establishing reversibility of algorithms, which is equivalent to the stochastic iteration being in detailed balance with the target density π; see [31].

2.1.1 Metropolis–Hastings MCMC

Consider first the plain vanilla Metropolis–Hastings algorithm for sampling from target density π_{t}. Given an initial state θ^0 and a proposal distribution with density function $q\left(\cdot|\theta\right)$, the Metropolis–Hastings algorithm for generating a chain of length N is given in Alg. 1.

```
Algorithm 1. Metropolis–Hastings (MH)
function: $\left[\theta^1, \ldots, \theta^N\right] = \mathbf{MH}\left(\pi_{\mathrm{t}}(\cdot), q(\cdot|\cdot), \theta^0, N\right)$
input:  density of target distribution $\pi_{\mathrm{t}}(\cdot)$, density of proposal distribution $q(\cdot|\cdot)$, initial state $\theta^0$,
        number of steps $N$
output: ordered list of states $\left[\theta^1, \ldots, \theta^N\right]$ (or just the final state $\theta^N$)
for $j = 0$ to $N - 1$ :
    • Given $\theta^j$, generate a proposal $\psi$ distributed as $q(\psi|\theta^j)$,
    • Accept proposal $\psi$ as the next state, i.e. set $\theta^{j+1} = \psi$, with probability
          $\alpha(\psi|\theta^j) = \min\left\{1, \frac{\pi_{\mathrm{t}}(\psi)q(\theta^j|\psi)}{\pi_{\mathrm{t}}(\theta^j)q(\psi|\theta^j)}\right\}$    (1)
      otherwise reject $\psi$ and set $\theta^{j+1} = \theta^j$.
```

For each j, Alg. 1 simulates a fixed stochastic iteration with θ^{j+1} being conditionally dependent only on θ^j, the state at step j, which can be represented by a fixed (stationary) transition kernel $K\left(y|x\right)$ that generates a (homogeneous) Markov chain. For target density π_{t}, detailed balance may be written

$$\pi_{\mathrm{t}}\left(x\right) K\left(y|x\right) = \pi_{\mathrm{t}}\left(y\right) K\left(x|y\right),$$

which, in general, is the property that K is self-adjoint in the measure π_{t}. See [31, Sec. 5.3] for a nice method for showing that K simulated by **MH** Alg. 1 is in detailed balance with π_{t}, and also for a more general class of acceptance probabilities.

Hence, under mild conditions on the proposal density q and the initial state θ^0, the ergodic theorem for Markov chains applies, which guarantees that the j-step density converges to π_{t}, asymptotically as $j \to \infty$. Hence, the Markov chain is π_{t}-ergodic.

A common choice of proposal distributions for inverse problems in multiple dimensions are random-walk proposals, though these typically lead to adjacent states of the chain being highly correlated, resulting in high computational cost to estimate posterior expectations with a desired accuracy. In the following we do not discuss the choice of proposal q, though in some sense our primary concern is how to improve a proposal once chosen. We also do not discuss the choice of initial state.

The following lemma gives an alternative form of the acceptance probability in Eq. (1) used later.

Lemma 1. *If the proposal transition kernel $q(\cdot|\cdot)$ in Alg. 1 is in detailed balance with some distribution π^*, then the acceptance probability* (1) *may be written*

$$\alpha(\psi|\theta^j) = \min\left\{1, \frac{\pi_t(\psi)\pi^*(\theta^j))}{\pi_t(\theta^j)\pi^*(\psi)}\right\} \tag{2}$$

Proof. Substitute the detailed balance statement $\pi^*(\psi)q(\theta^j|\psi) = \pi^*(\theta^j))q(\psi|\theta^j)$ into (1) to get (2), almost everywhere. □

2.1.2 MCMC for Hierarchical Bayesian Models

A hierarchical Bayesian model of some problem, including inverse problems, leads to the posterior distribution for unknown parameters θ conditioned on measured data $\mathbf{d}$, given by Bayes' rule

$$\pi(\theta|\mathbf{d}) = \frac{\pi(\mathbf{d}|\theta)\pi_{\mathrm{p}}(\theta)}{\pi(\mathbf{d})}. \tag{3}$$

In the Bayesian framework, solving the inverse problem is performed by exploring the posterior distribution $\pi(\theta|\mathbf{d})$ defined by (3) and evaluating statistics with respect to that distribution. Sample-based inference does this by drawing samples from the posterior distribution to evaluate sample-based Monte Carlo estimates of expected values. The plain vanilla route to drawing samples from $\pi(\theta|\mathbf{d})$ is to invoke **MH** Alg. 1 with $\pi_{\mathrm{t}}(\cdot) = \pi(\cdot|\mathbf{d})$ such that

$$\left[\theta^1, \ldots, \theta^N\right] = \mathbf{MH}\left(\pi(\theta|\mathbf{d}), q(\cdot|\cdot), \theta^0, N\right).$$

Asymptotically, the density of the jth state θ^j converges to the posterior density $\pi(\cdot|\mathbf{d})$ and averages over this chain converge to expectations with respect to $\pi(\cdot|\mathbf{d})$, asymptotically in N.

The following lemma formalises the usual observation that the unnormalised posterior density $\pi(\mathbf{d}|\theta)\pi_{\mathrm{p}}(\theta) \propto \pi(\theta|\mathbf{d})$ may be used to evaluate the Metropolis ratio in Eq. (1).

Lemma 2. *When $\pi(\mathbf{d})$ in* (3) *is finite, the Metropolis ratio $\pi_t(\psi)/\pi_t(\theta^j)$ in Alg. 1 Eq.* (1) *may be evaluated as a ratio of unnormalized densities*

$$\frac{\pi(\mathbf{d}|\psi)\pi_p(\psi)}{\pi(\mathbf{d}|\theta^j)\pi_p(\theta^j)}. \tag{4}$$

Proof. Substitute $\pi_{\mathrm{t}}(\cdot) = \pi(\cdot|\mathbf{d})$ from Eq. (3) into the Metropolis ratio and note that the normalisation constants $1/\pi(\mathbf{d})$ in the numerator and in the denominator cancel. □

Hereafter, for brevity we typically write the acceptance probability using the ratio of normalized posterior densities, as in Eq. (1), but actually compute with unnormalized densities, as in Eq. (4).

2.1.3 Delayed Acceptance MCMC

The Delayed Acceptance (DA) algorithm was introduced by Christen and Fox in [8], with the goal of reducing the computational cost per iteration by utilizing a computationally cheaper approximation of the forward map, and thus also of the posterior density, for evaluating the acceptance probability in Alg. 1. One may also view DA as a way to improve the proposal kernel q, since DA modifies the proposal kernel using a Metropolis–Hastings accept-reject step to give an effective proposal that is in detailed balance with an (approximate) distribution that is hopefully closer to the target than is the equilibrium distribution of the original proposal kernel.

The delayed acceptance algorithm is given in Alg. 2, for target (fine) density π_{F} and approximate (coarse) density π_{C}. Delayed acceptance first performs a standard Metropolis–Hastings accept/reject step (as given in Alg. 1) with the approximate/coarse density π_{C}. If accepted, a second accept reject/step is used, with acceptance probability chosen such that the composite iteration satisfies detailed balance with respect to the desired target π_{F}.

In Alg. 2 Eq. (6), $q_{\mathrm{C}}(\cdot|\cdot)$ is the effective proposal density from the first Metropolis–Hastings step with coarse density $\pi_{\mathrm{C}}(\cdot)$ as target; see [8] for details. The acceptance probability in Eq. (6) is the standard Metropolis–Hastings rule for proposal density q_{C}, targeting $\pi_{\mathrm{F}}(\cdot)$, hence Alg. 2 simulates a kernel in detailed balance with $\pi_{\mathrm{F}}(\cdot)$ and produces a chain that is ergodic with respect to $\pi_{\mathrm{F}}(\cdot)$; see [8] for conditions on the approximation that ensure that the ergodic theorem applies. Computational cost per iteration is reduced because for proposals that are rejected in the first **MH** step in Eq. (5), and thus result in $\psi = \theta^j$, the second acceptance ratio in Eq. (6) involving the more expensive, fine target density $\pi_{\mathrm{F}}(\cdot)$ does not need to be evaluated again.

```
Algorithm 2. Delayed Acceptance (DA)
function: $\left[\theta^1, \dots, \theta^N\right] = \mathbf{DA}\left(\pi_{\mathrm{F}}(\cdot), \pi_{\mathrm{C}}(\cdot), q(\cdot|\cdot), \theta^0, N\right)$
input:  target (fine) density $\pi_{\mathrm{F}}(\cdot)$, approximate (coarse) density $\pi_{\mathrm{C}}(\cdot)$, proposal kernel $q(\cdot|\cdot)$,
        initial state $\theta^0$, number of steps $N$
output: ordered list of states $\left[\theta^1, \dots, \theta^N\right]$ $\left(\text{or just the final state } \theta^N\right)$
for $j = 0$ to $N - 1$ :
    • Given $\theta^j$, generate proposal $\psi$ by invoking one step of MH Alg. 1 for coarse target $\pi_{\mathrm{C}}$:
          $\psi = \mathbf{MH}\left(\pi_{\mathrm{C}}(\cdot), q(\cdot|\cdot), \theta^j, 1\right)$ .        (5)
    • Accept proposal $\psi$ as the next state, i.e. set $\theta^{j+1} = \psi$, with probability
          $\alpha(\psi|\theta^j) = \min\left\{1, \frac{\pi_{\mathrm{F}}(\psi) q_{\mathrm{C}}(\theta^j|\psi)}{\pi_{\mathrm{F}}(\theta^j) q_{\mathrm{C}}(\psi|\theta^j)}\right\}$        (6)
      otherwise reject proposal $\psi$ and set $\theta^{j+1} = \theta^j$.
```

In the multilevel context with levels indexed by ℓ, the original **DA** Alg. 2 is a two-level method. Denote the more accurate forward map that defines the fine posterior distribution $\pi_\ell(\theta_\ell|\mathbf{d}_\ell)$ by $\mathcal{F}_\ell$, and the less accurate forward map that defines the approximate (coarse) posterior distribution $\pi_{\ell-1}(\theta_\ell|\mathbf{d}_{\ell-1})$ by $\mathcal{F}_{\ell-1}$. Note that we also allow a possibly altered or reduced data set $\mathbf{d}_{\ell-1}$ on level $\ell - 1$, but that the states in the two forward maps and in the two distributions are the same. Then setting $\pi_{\mathrm{F}}(\cdot) = \pi_\ell(\cdot|\mathbf{d}_\ell)$ and $\pi_{\mathrm{C}}(\cdot) = \pi_{\ell-1}(\cdot|\mathbf{d}_{\ell-1})$ in the call to **DA** Alg. 2, such that

$$\left[\theta_\ell^1, \dots, \theta_\ell^N\right] = \mathbf{DA}\left(\pi_\ell(\cdot|\mathbf{d}_\ell), \pi_{\ell-1}(\cdot|\mathbf{d}_{\ell-1}), q(\cdot|\cdot), \theta^0, N\right),$$

computes a chain that is ergodic with respect to $\pi_\ell(\cdot|\mathbf{d}_\ell)$, asymptotically as $N \to \infty$.

DA Alg. 2 actually allows for the approximate, coarse posterior distribution to depend on the state of the chain. Denote the state-dependent, approximate forward map at state θ by $\mathcal{F}_{\ell-1,\theta}$ and the resulting approximate posterior density by $\pi_{\ell-1,\theta}(\cdot|\mathbf{d}_{\ell-1})$. For state-dependent approximations it is always desirable and easy to achieve (see [14]) that $\mathcal{F}_{\ell-1,\theta}(\theta) = \mathcal{F}_\ell(\theta)$, so that $\pi_{\ell-1,\theta}(\theta|\mathbf{d}_{\ell-1}) = k\pi_\ell(\theta|\mathbf{d}_\ell)$ with the normalising constant k independent of state θ. The acceptance probability Eq. (6) then has the explicit form

$$\alpha(\psi|\theta^j) = \min\left\{1, \frac{\min\left\{\pi_{\mathrm{F}}(\psi)q(\theta^j|\psi), \pi_{\mathrm{C},\psi}(\theta^j)q(\psi|\theta^j)\right\}}{\min\left\{\pi_{\mathrm{F}}(\theta^j)q(\psi|\theta^j), \pi_{\mathrm{C},\theta_\ell^j}(\psi)q(\theta^j|\psi)\right\}}\right\}. \tag{7}$$

For technical reasons, as explained later, we will not use state-dependent approximations, but rather restrict ourselves to fixed approximate forward maps that do not depend on the current state.

2.1.4 Randomised-Length-Subchain Surrogate Transition MCMC

When the approximate forward map does not depend on the current state – for example, when using a fixed coarse discretization for a PDE – the resulting approximate posterior density is a fixed *surrogate* for the true posterior density, and Alg. 2 coincides with the surrogate transition method introduced by Liu [31]. Lemma 1 then implies that the acceptance probability in Eq. (6) is

$$\alpha(\psi|\theta^j) = \min\left\{1, \frac{\pi_{\mathrm{F}}(\psi)\pi_{\mathrm{C}}(\theta^j)}{\pi_{\mathrm{F}}(\theta^j)\pi_{\mathrm{C}}(\psi)}\right\}, \tag{8}$$

since the Metropolis–Hastings step in Eq. (5) ensures that the effective proposal kernel $q_{\mathrm{C}}(\cdot|\cdot)$ is in detailed balance with the approximate density $\pi_{\mathrm{C}}(\cdot)$.

We extend the surrogate transition method in two ways. As noted by Liu [31], multiple steps can be made with the surrogate, i.e. iterating the proposal and first accept/reject step Eq. (5) before performing the second accept/reject step with acceptance probability in Eq. (8). We call the sequence of states generated by multiple steps of Eq. (5) a *subchain*. Further, we consider subchains of random length, set according to a probability mass function (pmf) $p(\cdot)$ on the positive integers. In practice we set $J \in \mathbb{Z}^+$ and then set $p = \mathcal{U}(\{1, 2, \dots, J\})$, though note that a deterministic choice of subchain length is another special case. The utility of randomising the subchain length will become apparent in Section 2.3. These extensions are included in Alg. 3.

```
Algorithm 3. Randomised-Length-Subchain Surrogate Transition (RST)
function: $\left[\theta^1, \dots, \theta^N\right] = \mathbf{RST}\left(\pi_\mathrm{F}(\cdot), \pi_\mathrm{C}(\cdot), q(\cdot|\cdot), p(\cdot), \theta^0, N\right)$
input:   target (fine) density $\pi_\mathrm{F}(\cdot)$, surrogate (coarse) density $\pi_\mathrm{C}(\cdot)$, proposal kernel $q(\cdot|\cdot)$, prob-
         ability mass function $p(\cdot)$ over subchain length, initial state $\theta^0$, number of steps $N$
output:  ordered list of states $\left[\theta^1, \dots, \theta^N\right]$ (or just the final state $\theta^N$)
for $j = 0$ to $N - 1$ :
    • Draw the subchain length $n \sim p(\cdot)$.
    • Starting at $\theta^j$, generate subchain of length $n$ using MH Alg. 1 to target $\pi_\mathrm{C}(\cdot)$:
          $\psi = \mathbf{MH}\left(\pi_\mathrm{C}(\cdot), q(\cdot|\cdot), \theta^j, n\right)$    (9)
    • Accept the proposal $\psi$ as the next sample, i.e. set $\theta^{j+1} = \psi$, with probability
          $\alpha(\psi|\theta^j) = \min\left\{1, \frac{\pi_\mathrm{F}(\psi)\pi_\mathrm{C}(\theta^j)}{\pi_\mathrm{F}(\theta^j)\pi_\mathrm{C}(\psi)}\right\}.$    (10)
      otherwise reject and set $\theta^{j+1} = \theta^j$.
```

We will show that Alg. 3 satisfies detailed balance using the following lemma, needed also later.

Lemma 3. *Let $K_1(x|y)$ and $K_2(x|y)$ be two transition kernels that are in detailed balance with a density π and that commute. Then their composition $(K_1 \circ K_2)$ is also in detailed balance with π.*

Proof.

$$\begin{aligned}\pi(\psi)(K_1 \circ K_2)(\theta|\psi) &= \pi(\psi)\int K_1(\theta|\phi)K_2(\phi|\psi)\mathrm{d}\phi = \pi(\psi)\int K_2(\theta|\phi)K_1(\phi|\psi)\mathrm{d}\phi \\ &= \pi(\psi)\int K_2(\phi|\theta)\frac{\pi(\theta)}{\pi(\phi)}K_1(\psi|\phi)\frac{\pi(\phi)}{\pi(\psi)}\mathrm{d}\phi \\ &= \pi(\theta)\int K_2(\phi|\theta)K_1(\psi|\phi)\mathrm{d}\phi = \pi(\theta)(K_1 \circ K_2)(\psi|\theta)\end{aligned}$$

□

Lemma 4. *Alg. 3 simulates a Markov chain that is in detailed balance with $\pi_\mathrm{F}(\cdot)$.*

Proof. Recall that the effective density $q_\mathrm{C}(\cdot|\cdot)$ for proposals drawn according to Alg. 2 Eq. (5) is in detailed balance with $\pi_\mathrm{C}(\cdot)$. Since q_C clearly commutes with itself, using Lemma 3, it follows by induction that $q_\mathrm{C}^n(\cdot|\cdot)$, i.e., q_C composed n times with itself) is in detailed balance with $\pi_\mathrm{C}(\cdot)$ for any n. Hence, the effective proposal density induced by Alg. 3 Eq. (9), namely the mixture kernel $\sum_{n\in\mathbb{Z}^+} p(n)q_\mathrm{C}^n(\cdot|\cdot)$ is also in detailed balance with $\pi_\mathrm{C}(\cdot)$.

Finally, the acceptance probability in Alg. 3 Eq. (10) for target density $\pi_\mathrm{F}(\cdot)$ follows from Lemma 1, since the proposal kernel is in detailed balance with $\pi_\mathrm{C}(\cdot)$. Consequently, Alg. 3 produces a chain in detailed balance with $\pi_\mathrm{F}(\cdot)$. □

Remark 1. (a) Choosing a multinomial pmf over the subchain length, with $p(J) = 1$ and $p(\neg J) = 0$, implies that Lemma 4 is also valid for the special case of a fixed subchain length J_C.

(b) We do not yet have a version of Lemma 4 for fully state-dependent approximations, which is why we restrict here to state-independent surrogates.

When both posterior distributions are with respect to the same prior distribution, the acceptance probability in Eq. (10) simplifies to a ratio of likelihood functions. The proof is obvious.

Lemma 5. *If the densities of the coarse and fine posterior distributions in Alg. 3 are with respect to the same prior distribution, i.e.* $\pi_{\mathrm{F}}(\theta) = \pi_\ell(\theta|\mathbf{d}_\ell) \propto \pi_\ell(\mathbf{d}_\ell|\theta)\pi_{\mathrm{p}}(\theta)$ *and* $\pi_{\mathrm{C}}(\theta) = \pi_{\ell-1}(\theta|\mathbf{d}_{\ell-1}) \propto \pi_{\ell-1}(\mathbf{d}_{\ell-1}|\theta)\pi_{\mathrm{p}}(\theta)$*, the acceptance probability in Alg. 3 Eq.* (10) *is equal to*

$$\alpha(\psi|\theta^j) = \min\left\{1, \frac{\pi_\ell\big(\mathbf{d}_\ell|\psi\big)\pi_{\ell-1}\big(\mathbf{d}_{\ell-1}|\theta^j\big)}{\pi_\ell\big(\mathbf{d}_\ell|\theta^j\big)\pi_{\ell-1}\big(\mathbf{d}_{\ell-1}|\psi\big)}\right\}. \tag{11}$$

2.1.5 Different Fine and Coarse States

In delayed acceptance Alg. 2, and hence also in the randomised surrogate transition Alg. 3, the state in the fine and coarse target distributions is the same. In the MLMCMC of Dodwell *et al.* [15] the (fine) state θ_ℓ at level ℓ is partitioned into "coarse modes" (or "components") denoted $\theta_{\ell,\mathrm{C}}$ and "fine modes" $\theta_{\ell,\mathrm{F}}$, so that $\theta_\ell = (\theta_{\ell,\mathrm{F}}, \theta_{\ell,\mathrm{C}})$. The coarse modes are the components of the state vector on the coarse, approximate level $\ell - 1$, i.e., $\theta_{\ell,\mathrm{C}} = \theta_{\ell-1}$, and the target π_{C} at the coarse level is a function of the coarse modes only, while the fine target distribution additionally depends also on the fine modes.

The randomised surrogate transition Alg. 3 is easily extended to allow this structure, as shown in Alg. 4 below, where surrogate transition is only used to propose the states of the coarse modes, while the fine modes are drawn from some additional proposal distribution. The composite of the fine and coarse proposals then forms the proposed state at the fine level. For this extension it is important that the fine modes are proposed independently of the coarse modes to ensure detailed balance, as shown below.

Lemma 6. *Two Level Delayed Acceptance in Alg. 4 generates a chain in detailed balance with* π_{F}.

Proof. As noted in the proof of Lemma 4, the proposal density q_{C} induced by the surrogate transition step in Alg. 4 Eq. (12) is in detailed balance with the coarse target density $\pi_{\mathrm{C}}(\cdot)$ over θ_{C}. As a kernel on the composite state $\theta = (\theta_{\mathrm{F}}, \theta_{\mathrm{C}})$ we can write the coarse proposal as

$$K_{\mathrm{C}} = \left[\begin{array}{c:c} I & 0 \\ \hdashline 0 & q_{\mathrm{C}} \end{array}\right]$$

where I denotes the identity of appropriate dimension. Similarly, the fine proposal Eq. (13) on the composite state has kernel

$$K_{\mathrm{F}} = \left[\begin{array}{c:c} q_{\mathrm{F}} & 0 \\ \hdashline 0 & I \end{array}\right].$$

Since K_{F} does not change the coarse modes, it trivially is in detailed balance with $\pi_{\mathrm{C}}(\cdot)$. Further, it is easy to check that K_{C} and K_{F} commute. Hence, by Lemma 3 the composition $(K_{\mathrm{F}} \circ K_{\mathrm{C}}^n)$ is also in detailed balance with $\pi_{\mathrm{C}}(\cdot)$ and so is the effective proposal kernel $\sum_{n \in \mathbb{Z}^+} p(n)(K_{\mathrm{F}} \circ K_{\mathrm{C}}^n)$ for drawing $\psi = (\psi_{\mathrm{F}}, \psi_{\mathrm{C}})$ according to Alg. 4 Eqs. (12) and (13). The acceptance probability in Alg. 4 Eq. (14) then follows again from Lemma 1 and the chain produced by Alg. 4 is in detailed balance with $\pi_{\mathrm{F}}(\cdot)$, as desired. □

Note that the Randomised Surrogate Transition Alg. 3 is a special case of Alg. 4 with $\theta^j = \theta^j_{\mathrm{C}}$, i.e. θ^j_{F} is empty, and correspondingly $q_{\mathrm{F}}(\cdot|\cdot)$ is the (trivial) proposal on the empty space.

```
Algorithm 4. Two Level Delayed Acceptance (TLDA)
function: $\left[\theta^1, \ldots, \theta^N\right] = \mathbf{TLDA}\left(\pi_{\mathrm{F}}(\cdot), \pi_{\mathrm{C}}(\cdot), q(\cdot|\cdot), q_{\mathrm{F}}(\cdot|\cdot), p(\cdot), \theta^0, N\right)$
input:  target (fine) density $\pi_{\mathrm{F}}(\cdot)$, surrogate (coarse) density $\pi_{\mathrm{C}}(\cdot)$, proposal kernel $q(\cdot|\cdot)$ on
        coarse modes, proposal kernel $q_{\mathrm{F}}(\cdot|\cdot)$ on fine modes, probability mass function $p(\cdot)$
        over subchain length, initial state $\theta^0$, number of steps $N$
output: ordered list of states $\left[\theta^1, \ldots, \theta^N\right]$ (or just the final state $\theta^N$)
for $j = 0$ to $N - 1$ :
    • Draw the subchain length $n \sim p(\cdot)$.
    • Starting at $\theta^j_{\mathrm{C}}$, generate subchain of length $n$ using MH Alg. 1 to target $\pi_{\mathrm{C}}(\cdot)$:
          $\psi_{\mathrm{C}} = \mathbf{MH}\left(\pi_{\mathrm{C}}(\cdot), q(\cdot|\cdot), \theta^j_{\mathrm{C}}, n\right)$    (12)
    • Draw the fine-mode proposal
          $\psi_{\mathrm{F}} \sim q_{\mathrm{F}}(\cdot|\theta^j_{\mathrm{F}})$    (13)
    • Accept proposal $\psi = (\psi_{\mathrm{F}}, \psi_{\mathrm{C}})$ as next sample, i.e., set $\theta^{j+1} = \psi$, with probability
          $\alpha(\psi|\theta^j) = \min\left\{1, \frac{\pi_{\mathrm{F}}(\psi)\pi_{\mathrm{C}}(\theta^j_{\mathrm{C}})}{\pi_{\mathrm{F}}(\theta^j)\pi_{\mathrm{C}}(\psi_{\mathrm{C}})}\right\}.$    (14)
      otherwise reject and set $\theta^{j+1} = \theta^j$.
```

2.1.6 Multilevel Delayed Acceptance

The multilevel delayed acceptance algorithm is a recursive version of **TLDA** in which instead of invoking Metropolis–Hastings to generate a subchain at the coarser levels the algorithm is recursively invoked again (except for the coarsest level $\ell = 0$).

To be more precise, **MLDA** Alg. 5 below is called on the most accurate, finest level L. Then, for levels $1 \leq \ell \leq L$ it generates a subchain at level $\ell - 1$ as in **TLDA**, by recursively invoking **MLDA** on level $\ell - 1$, until the coarsest level $\ell = 0$ is reached where plain **MH** in invoked. Required for **MLDA** are the hierarchy of density functions $\pi_0(\cdot), \ldots, \pi_L(\cdot)$ along with a coarsest-level proposal q_0, partitions into coarse and fine modes at each level, fine-level proposals $q_{1,\mathrm{F}}, \ldots, q_{L,\mathrm{F}}$ and probability mass functions $p_1(\cdot), \ldots, p_L(\cdot)$ over the subchain lengths on levels 0 to $L - 1$.

A chain of length N at level L is then produced by calling

$$\left[\theta^1_L, \ldots, \theta^N_L\right] = \mathbf{MLDA}\left(\{\pi_k\}^L_{k=0}, q_0, \{q_{k,\mathrm{F}}\}^L_{k=1}, \{p_k\}^L_{k=1}, \theta^0_L, L, N\right). \tag{15}$$

We can now state the main theoretical result of paper.

Theorem 1. *Multilevel Delayed Acceptance in Alg. 5, invoked as in* (15)*, generates a Markov chain that is in detailed balance with π_L.*

Proof. The proof follows essentially by induction on the level ℓ from the proof of Lemma 6. At level $\ell = 1$, **MLDA** is equivalent to **TLDA**, and so the base step follows immediately from Lemma 6. Let us now assume that the proposal kernel for $\psi = (\psi_{\mathrm{F}}, \psi_{\mathrm{C}})$ on level ℓ simulated using **MLDA** on level $\ell - 1$ is in detailed balance with $\pi_{\ell-1}$. Then it follows from Lemma 1 that the acceptance probability in Alg. 5 Eq. (16) produces a Markov chain that is in detailed balance with $\pi_\ell(\cdot)$, which concludes the induction step. □

```
Algorithm 5. Multilevel Delayed Acceptance (MLDA):
function: $\left[\theta_\ell^1, \dots, \theta_\ell^N\right] = \mathbf{MLDA}\left(\{\pi_k\}_{k=0}^{\ell}, q_0, \{q_{k,\mathrm{F}}\}_{k=1}^{\ell}, \{p_k\}_{k=1}^{\ell}, \theta_\ell^0, \ell, N\right)$
input:  target densities $\pi_0(\cdot), \dots \pi_\ell(\cdot)$, proposal densities $q_0(\cdot|\cdot)$ and $q_{1,\mathrm{F}}(\cdot|\cdot), \dots, q_{\ell,\mathrm{F}}$, probability
        mass functions $p_1(\cdot), \dots, p_\ell(\cdot)$ over subchain lengths on levels 0 to $\ell - 1$, initial
        state $\theta_\ell^0$, current level index $\ell$, number of steps $N$
output: ordered list of states $[\theta_\ell^1, \dots, \theta_\ell^N]$ at level $\ell$ (or just the final state $\theta_\ell^N$)
for $j = 0$ to $N - 1$ :
    • Draw the subchain length $n_\ell \sim p_\ell(\cdot)$ for level $\ell - 1$.
    • Starting at $\theta_{\ell,\mathrm{C}}^j$, generate a subchain of length $n_\ell$ on level $\ell - 1$:
        – If $\ell = 1$, use the Metropolis–Hastings algorithm to generate the subchain
              $\psi_\mathrm{C} = \mathbf{MH}\left(\pi_0(\cdot), q_0(\cdot,\cdot), \theta_{1,\mathrm{C}}^j, n_1\right)$ .
        – If $\ell > 1$, generate the subchain by (recursively) calling MLDA
              $\psi_\mathrm{C} = \mathbf{MLDA}\left(\{\pi_k(\cdot)\}_{k=0}^{\ell-1}, q_0(\cdot|\cdot), \{q_{k,\mathrm{F}}\}_{k=1}^{\ell-1}, \{p_k\}_{k=1}^{\ell-1}, \theta_{\ell,\mathrm{C}}^j, \ell - 1, n_\ell\right)$ .
    • Draw the fine-mode proposal $\psi_\mathrm{F} \sim q_{\ell,\mathrm{F}}\left(\cdot\, |\theta_{\ell,\mathrm{F}}^j\right)$.
    • Accept proposal $\psi = (\psi_\mathrm{F}, \psi_\mathrm{C})$ as next sample, i.e., set $\theta_\ell^{j+1} = \psi$, with probability
              $\alpha(\psi|\theta^j) = \min\left\{1, \frac{\pi_\ell(\psi)\pi_{\ell-1}\left(\theta_{\ell,\mathrm{C}}^j\right)}{\pi_\ell\left(\theta_\ell^j\right)\pi_{\ell-1}(\psi_\mathrm{C})}\right\}$        (16)
      otherwise reject and set $\theta_\ell^{j+1} = \theta_\ell^j$.
```

2.2 Comparison with MLMCMC

The generalisation of Delayed Acceptance to an extended multilevel setting leads to clear similarities with the Multilevel Markov Chain Monte Carlo (MLMCMC) Method proposed by Dodwell *et al.* [15]. The more subtle difference between the two approaches is illustrated in Fig. 1.

The MLDA algorithm can be seen as a recursive application of the surrogate transition method over multiple levels. If a proposal ψ from level $\ell - 1$ for level ℓ at state θ_ℓ^j is rejected, the initial state for the coarse subchain $\theta_{\ell-1}^0$ is set back to θ_ℓ^j. Hence, the new coarse subchain, which will generate the next proposal for level ℓ, is initialised from the same state as the previous subchain.

For MLMCMC [15], even if the coarse proposal is rejected, the coarse chain continues independently of the fine chain. In analogy to the subchain picture in MLDA, this corresponds to initialising the subchain on level $\ell - 1$ with the coarse state ψ_C that has just been rejected on level ℓ. As a result, coarse and fine chains will separate and only re-coalesce once a coarse proposal is accepted at the fine level. This choice provides better mixing at coarse levels and allows for efficient parallelisation of the MLMCMC algorithm [44], but it does entail one important caveat; The practical algorithm in [15, Alg. 3] does not necessarily define a Markov process unless coarse proposals passed to the next finer level are independent, as in [15, Alg. 2]. The practical implication of violating this requirement is that we do not have a proof of convergence of MLMCMC with finite subchains because we cannot apply the theorems that guarantee convergence for homogeneous Markov chains. Indeed, numerical

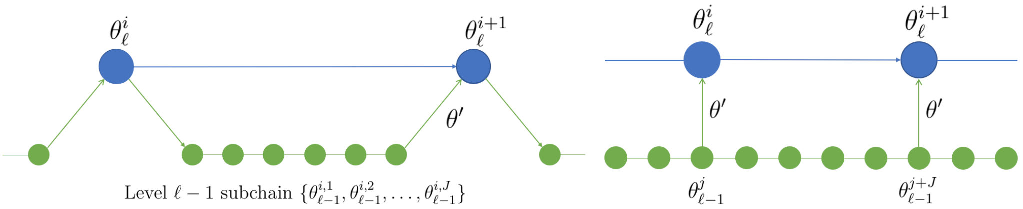


Figure 1: Schematic for generating a proposal θ' on level ℓ for MLDA (left) and MLMCMC (right) using a fixed length subchain of length J. The key difference is that for MLMCMC the coarse chain on level $\ell-1$ is generated independently of the chain on level ℓ.

experiments (not shown) indicate that estimates using MLMCMC with finite subchains are biased and that the underlying chains do not converge to the desired target distributions.

Accordingly, in theory the practical multilevel estimator proposed by Dodwell *et al.* [15, Alg. 3] is only unbiased if the coarse proposal is an independent sample from $\pi_{\ell-1}$; therefore only at infinite computational cost (i.e. when the subchain length goes to infinity). However, if the fixed subchain length is chosen to be greater than twice the integrated autocorrelation length of the chain at that level, in practice this bias disappears. This imposes the constraint that the subchain length might have to be fairly long. If the acceptance rate is also relatively low, the method becomes computationally inefficient, i.e. a lot of computational effort has to be put into generating independent proposals from a coarse distribution which are then rejected with high probability.

2.3 Extension 1: Exploiting Variance Reduction with a Multilevel Estimator

Using the MLDA sampler proposed above, it is in fact possible to define an asymptotically unbiased multilevel estimator that retains most of the computational benefits of both Multilevel Monte Carlo [19] and MLMCMC [15]. Let $Q_\ell(\theta_\ell)$ define some quantity of interest computed on level $\ell = 0, \dots, L$. The aim is to estimate $\mathbb{E}_{\pi_L}[Q_L]$ – the expectation of Q_L with respect to the posterior distribution π_L on the finest level L – using as little computational effort as possible.

The idea of Multilevel Monte Carlo is, at its heart, very simple. The key is to avoid estimating the expected value $\mathbb{E}_\ell[Q_\ell]$ directly on level ℓ, but instead to estimate the correction with respect to the next lower level. Under the assumption that samples on level $\ell-1$ are cheaper to compute than on level ℓ and that the variance of the correction term is smaller than the variance of Q_ℓ itself, the cost of computing this estimator is much lower than an estimator defined solely on samples from level ℓ. In the context of MLDA and MLMCMC, the target density π_ℓ depends on ℓ, so that we write

$$\mathbb{E}_{\pi_L}[Q_L] = \mathbb{E}_{\pi_0}[Q_0] + \sum_{\ell=1}^{L} \Big(\mathbb{E}_{\pi_\ell}[Q_\ell] - \mathbb{E}_{\pi_{\ell-1}}[Q_{\ell-1}]\Big), \tag{17}$$

which is achieved by adding and subtracting $\mathbb{E}_{\pi_\ell}[Q_\ell]$ for all levels $\ell = 0, \dots, L-1$. Note that for the particular case where the densities $\{\pi_\ell\}_{\ell=0}^L$ are all equal, this reduces to the simple telescoping sum forming the basis of standard Multilevel Monte Carlo [19].

The practical MLMCMC algorithm in [15, Alg. 3] now proceeds by estimating the first term in Eq. (17) using the MCMC estimator $E_{\pi_0}[Q_0] \approx \frac{1}{N_0}\sum_{i=1}^{N_0} Q_0(\theta_0^i)$ with a Markov chain $\big[\theta_0^1, \dots, \theta_0^{N_0}\big]$ produced with a standard **MH** on the coarsest level. Each of the correction terms for $\ell \geq 1$ is estimated by

$$\mathbb{E}_{\pi_\ell}[Q_\ell] - \mathbb{E}_{\pi_{\ell-1}}[Q_{\ell-1}] \approx \frac{1}{N_\ell}\sum_{i=1}^{N_\ell} Q_\ell\big(\theta_\ell^i\big) - Q_{\ell-1}\big(\theta_{\ell-1}^{J_\ell i}\big), \tag{18}$$

where N_ℓ is the total number of samples on level ℓ after subtracting burn-in, J_ℓ is the subchain length on level $\ell - 1$ and $\theta^{J_\ell i}_{\ell-1}$ is the state of the coarse chain used as the proposal for the ith state of the fine chain in the MLMCMC algorithm.

As mentioned in Section 2.2, this multilevel estimator is only unbiased for MLMCMC as $J_\ell \to \infty$ or, in practice, for coarse subchains with J_ℓ greater than twice the integrated autocorrelation length.

An unbiased multilevel estimator can be produced using MLDA, without this constraint on the subchain lengths. We achieve this by employing a particular form of RST Alg. 3 in the MLDA Alg. 5. For all $\ell = 1, \dots, L$, we set the probability mass function over the subchain length on level $\ell - 1$ to the discrete uniform distribution $p_\ell = \mathcal{U}(1, J_\ell)$, where J_ℓ is the maximum subchain length. Hence, the jth proposal $\psi_{\mathrm{C}} = \psi^j_{\ell-1}$ for the coarse modes on level ℓ in this version of MLDA constitutes an independent, uniformly-at-random draw from a subchain of length J_ℓ on level $\ell - 1$. Crucially, we let the coarse sampler continue sampling beyond the proposed state to produce subchains of fixed length J_ℓ for each state of the fine chain. Moreover, we also evaluate and store the quantity of interest at each state of each of those subchains on level $\ell - 1$.

Thus, using MLDA in this way to compute a chain $[\theta^1_L, \dots, \theta^N_L]$ on the finest level L. In addition to the

$$N_L = N \;\text{ samples }\quad Q_L\big(\theta^1_L\big), \dots, Q_L\big(\theta^{N_L}_L\big) \quad\text{ on level } L,$$

we obtain also

$$N_\ell = N \times \prod_{k=\ell}^{L-1} J_{k+1} \;\text{ samples }\quad Q_\ell\big(\theta^1_\ell\big), \dots, Q_\ell\big(\theta^{N_\ell}_\ell\big) \quad\text{ on levels }\; \ell = 0, \dots, L-1.$$

Using those samples the following asymptotically unbiased MLDA estimator of the posterior expectation $\mathbb{E}_{\pi_L}[Q_L]$ can be defined:

$$\widehat{Q}_L := \frac{1}{N_0} \sum_{i=1}^{N_0} Q_0\big(\theta^i_0\big) + \sum_{\ell=1}^{L} \frac{1}{N_\ell} \sum_{j=1}^{N_\ell} Q_\ell\big(\theta^j_\ell\big) - Q_{\ell-1}\big(\psi^j_{\ell-1}\big). \tag{19}$$

Here, $\psi^j_{\ell-1}$ denotes the proposal ψ_{C} for the coarse modes of the jth state θ^j_ℓ of the Markov chain on level ℓ produced by MLDA in Alg. 5.

Let us first discuss, why this estimator is asymptotically unbiased. For each j, the proposals ψ^j_{l-1} are independently and uniformly drawn from the subchain $[\theta^k_{l-1} : (j-1)J_\ell < k \leq jJ_\ell]$. Thus, the ensemble $\big\{\psi^1_{l-1}, \dots, \psi^{N_\ell}_{l-1}\big\}$ is a random draw from $\big\{\theta^1_{l-1}, \dots, \theta^{N_{\ell-1}}_{l-1}\big\}$ and thus identically distributed. As a consequence, in the limit as $N_\ell \to \infty$ for all ℓ, most terms on the right hand side of Eq. (19) cancel. What remains, is $\sum_{j=1}^{N_L} Q_L\big(\theta^j_L\big)$, which due to Theorem 1 is an unbiased estimator for $\mathbb{E}_{\pi_L}[Q_L]$ in the limit as $N_L \to \infty$.

Since the coarse subsamplers in MLDA are repeatedly realigned with the next finer distribution by way of the MLDA transition kernel, the samples on the coarse levels are in fact not distributed according to the "vanilla" densities $\{\pi_\ell\}_{\ell=0}^{L-1}$, but come from some "hybrid" mixture distributions.With the particular choice for p_ℓ, the density of the mixture distribution arising from subsampling the coarse density on level $\ell - 1 < L$ can be written

$$\tilde{\pi}_{\ell-1} = \frac{1}{J_\ell} \sum_{n=1}^{J_\ell} K^n_{\ell-1}\, \tilde{\pi}_{\ell,C} \tag{20}$$

where $\tilde{\pi}_{\ell,C}$ is the marginal density of the coarse modes of the next finer density, $K_{\ell-1}$ is the transition kernel simulated by each step of subsampling on level $\ell - 1$, and $K^n_{\ell-1}$ is that kernel composed with itself n times. Recall again that according to Theorem 1 the finest sampler targets the exact posterior, so that $\tilde{\pi}_L = \pi_L$. Thus, the MLDA estimator in Eq. (19) approximates the following telescoping sum:

$$\mathbb{E}_{\pi_L}[Q_L] = \mathbb{E}_{\tilde{\pi}_0}[Q_0] + \sum_{\ell=1}^{L} \Big(\mathbb{E}_{\tilde{\pi}_\ell}[Q_\ell] - \mathbb{E}_{\tilde{\pi}_{\ell-1}}[Q_{\ell-1}] \Big), \tag{21}$$

which is a small but crucial difference to the sum in Eq. (17) that forms the basis of MLMCMC [15].

The computational gains due to multilevel variance reduction remain. In fact, since the mixture densities $\tilde{\pi}_{\ell-1}$ are conditioned every J_ℓ steps on the next finer chain, they are even closer and thus, the variances of the correction terms in Eq. (19) will be further reduced compared to the variances of the estimates in Eq. (18). The fixed subchain lengths J_ℓ and thus the numbers of samples N_ℓ on the coarser levels can then be chosen as usual in multilevel Monte Carlo approaches to minimise the total variance for a fixed computational budget, or to minimise the cost to achieve the smallest variance. We are not going to go into more depth with respect to this estimator in this paper, but refer to e.g. [9, 15, 20] for detailed analyses of Multilevel (Markov Chain) Monte Carlo estimators.

2.4 Extension 2: Adaptive Correction of the Approximate Posteriors

While the algorithm outlined in Section 2.1 does guarantee sampling from the exact posterior, there are situations where convergence can be prohibitively slow. When the coarse model approximations are poor, the second-stage acceptance probability can be low, and many proposals will be rejected. This will result in suboptimal acceptance rates, poor mixing and low effective sample sizes. The leftmost panel in Fig. 2 shows a contrived example where the approximate likelihood function (red isolines) is offset from the exact likelihood function (blue contours) and its scale, shape and orientation are incorrect.

One way to alleviate this problem is through *tempering*, where the variance in the likelihood function $\mathbf{\Sigma}_\epsilon$ on levels $\ell < L$ is inflated, resulting in a wider approximate posterior distribution. While this approach would allow the approximate posterior to encapsulate the exact posterior, it does not tackle the challenge in an intelligent fashion, and the inflation factor introduces an additional tuning parameter.

In place of tempering, an enhanced Adaptive Error Model (AEM) can be employed to account for discrepancies between model levels. Let $\mathcal{F}_\ell$ denote the coarse forward map on level ℓ and $\mathcal{F}_L$ denote the forward map on the finest level L. To obtain a better approximation of the data d using $\mathcal{F}_\ell$, the two-level AEM suggested in [13] and analysed in [14, 17] is extended here by adding a telescopic sum of the differences in the forward model output across all levels from ℓ to L:

$$d = \mathcal{F}_L(\theta) + \epsilon = \mathcal{F}_\ell(\theta) + \mathcal{B}_\ell(\theta) + \epsilon \quad \text{with} \quad \mathcal{B}_\ell(\theta) := \sum_{k=\ell}^{L-1} \underbrace{\mathcal{F}_{k+1}(\theta) - \mathcal{F}_k(\theta)}_{:=B_k(\theta)} \tag{22}$$

denoting the bias on level ℓ at θ. The trick in the context of MLDA is that, since $\mathcal{B}_\ell$ is just a simple sum, the individual bias terms B_k from pairs of adjacent model levels can be estimated independently, so that new information can be exploited each time *any* set of adjacent levels are evaluated for the same parameter value θ.

Approximating each individual bias term $B_k = \mathcal{F}_{k+1} - \mathcal{F}_k$ with a multivariate Gaussian $B_k^* \sim \mathcal{N}(\mu_k, \mathbf{\Sigma}_k)$, the total bias $\mathcal{B}_\ell$ can be approximated by the Gaussian $\mathcal{B}_\ell^* \sim \mathcal{N}(\mu_{\mathcal{B},\ell}, \mathbf{\Sigma}_{\mathcal{B},\ell})$ with $\mu_{\mathcal{B},\ell} = \sum_{k=\ell}^{L-1} \mu_k$ and $\mathbf{\Sigma}_{\mathcal{B},\ell} = \sum_{k=\ell}^{L-1} \mathbf{\Sigma}_k$.

The bias-corrected likelihood function for level ℓ is then proportional to

$$\mathcal{L}_\ell(\mathbf{d}|\theta) = \exp\left(-\frac{1}{2}(\mathcal{F}_\ell(\theta) + \mu_{\mathcal{B},\ell} - \mathbf{d})^T (\mathbf{\Sigma}_{\mathcal{B},\ell} + \mathbf{\Sigma}_e)^{-1} (\mathcal{F}_\ell(\theta) + \mu_{\mathcal{B},\ell} - \mathbf{d}) \right). \tag{23}$$

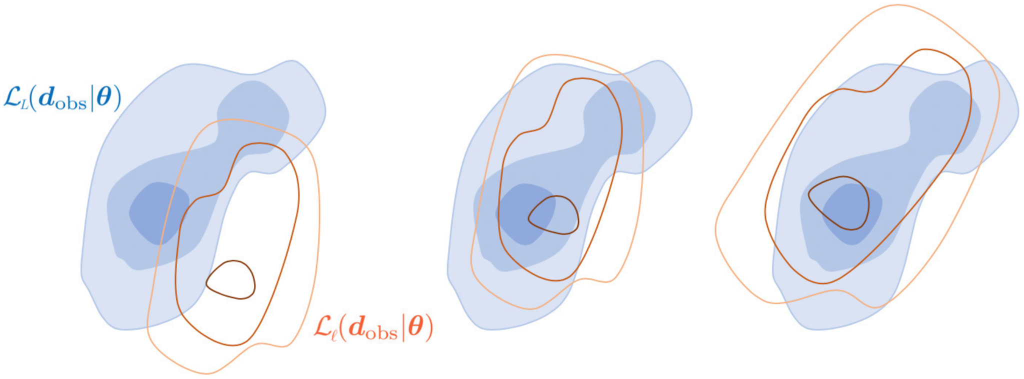

$\mathcal{L}_L(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta})$
$\mathcal{L}_\ell(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta})$

Figure 2: Effect of applying the Gaussian Adaptive Error Model (AEM). The first panel shows the initial state before adaptation, where the coarse likelihood function ($\mathcal{L}_\ell(\mathbf{d}_{obs}|\theta)$, red isolines) approximates the fine likelihood function ($\mathcal{L}_L(\mathbf{d}_{obs}|\theta)$, blue contours) poorly. The second panel shows the effect of adding the mean of the bias to the likelihood functional, resulting in an offset of the coarse model likelihood function. The third panel shows the effect of also adding the covariance of the bias to the likelihood functional, resulting in a scaling and rotation of the coarse likelihood function. Adapted from [33].

The *Approximation* Error Model, suggested by [28], is constructed offline, by sampling from the prior distribution before running the MCMC; We simply sample N parameter sets from the prior and compute the sample moments according to

$$\mu_k = \frac{1}{N}\sum_{i=1}^{N} B_k(\theta^{(i)}) \quad \text{and} \quad \mathbf{\Sigma}_k = \frac{1}{N-1}\sum_{i=1}^{N}(B_k(\theta^{(i)}) - \mu_k)(B_k(\theta^{(i)}) - \mu_k)^T. \tag{24}$$

However, this approach requires significant investment prior to sampling, and may result in a sub-optimal error model, since the bias in the posterior distribution is very different from the bias in the prior when the data is informative. Instead, as suggested in [13], an estimate for B_k can be constructed iteratively during sampling, using the following recursive formulae for sample means and sample covariances [21]:

$$\mu_{k,i+1} = \frac{1}{i+1}\Big(i\mu_{k,i} + B_k(\theta^{i+1})\Big) \quad \text{and} \tag{25}$$

$$\mathbf{\Sigma}_{k,i+1} = \frac{i-1}{i}\mathbf{\Sigma}_{k,i} + \frac{1}{i}\Big(i\mu_{k,i}\,\mu_{k,i}^T - (i+1)\mu_{k,i+1}\,\mu_{k,i+1}^T + B_k(\theta^{i+1})\,B_k(\theta^{i+1})^T\Big). \tag{26}$$

While this approach in theory results in a MCMC algorithm that is not Markov, the recursively constructed sample moments converge as sampling proceeds and hence the approach exhibits *diminishing adaptation* and *bounded convergence* which is sufficient to ensure ergodicity for adaptive MCMC schemes, [40, 41]. As shown in [14], it is also possible to construct a *state-dependent* AEM, where the coarse samples are corrected only according to the bias of the state of the MCMC, rather than the mean of the bias. This approach, however, may require a different form of the multilevel acceptance probability (16), which we have not yet established, as discussed in Section 2.1.

3 Examples

In this section, we consider three inverse problems which demonstrate the efficiency gains obtained by using MLDA, as well as by the extensions outlined above. The algorithm has been included in the free and open source probabilistic programming library PyMC[2] as the MLDA step method since version 3.10.0, and the examples below were all completed using this implementation.

3.1 Gravitational Survey

In this example, we consider a 2-dimensional gravity surveying problem, adapted from the 1-dimensional problem presented in [23]. Our aim is to recover an unknown two-dimensional mass density distribution $f(\mathbf{t})$ at a known depth d below the surface from measurements $g(\mathbf{s})$ of the vertical component of the gravitational field at the surface. The contribution to $g(\mathbf{s})$ from infinitesimally small areas of the subsurface mass distribution are given by:

$$\mathrm{d}g(\mathbf{s}) = \frac{\sin\theta}{r^2} f(\mathbf{t})\,\mathrm{d}\mathbf{t} \tag{27}$$

where θ is the angle between the vertical plane and a straight line between two points $\mathbf{t}$ and $\mathbf{s}$, and $r = \|\mathbf{s}-\mathbf{t}\|_2$ is the Eucledian distance between the points. We exploit that $\sin\theta = d/r$, so that

$$\frac{\sin\theta}{r^2} f(\mathbf{t})\,\mathrm{d}\mathbf{t} = \frac{d}{r^3} f(\mathbf{t})\,\mathrm{d}\mathbf{t} = \frac{d}{\|\mathbf{s}-\mathbf{t}\|_2^3} f(\mathbf{t})\,\mathrm{d}\mathbf{t} \tag{28}$$

This yields the integral equation

$$g(\mathbf{s}) = \iint_T \frac{d}{\|\mathbf{s}-\mathbf{t}\|_2^3} f(\mathbf{t})\,\mathrm{d}\mathbf{t} \tag{29}$$

where $T = [0,1]^2$ is the domain of the function $f(\mathbf{t})$. This constitutes our forward model. We solve the integral numerically using midpoint quadrature. For simplicity, we use m quadrature points along each dimension, so that in discrete form our forward model becomes

$$g(\mathbf{s}_i) = \sum_{l=1}^{m} \omega_l \sum_{k=1}^{m} \omega_k \frac{d}{\|\mathbf{s}_i - \mathbf{t}_{k,l}\|_2^3} \hat{f}(\mathbf{t}_{k,l}) = \sum_{j=1}^{m^2} \omega_j \frac{d}{\|\mathbf{s}_i - \mathbf{t}_j\|_2^3} \hat{f}(\mathbf{t}_j) \tag{30}$$

where $\omega_j = 1/m^2$ are the quadrature weights, $\hat{f}(\mathbf{t}_j)$ is the approximate subsurface mass at the quadrature points $\mathbf{t}_j,\ j = 1,\dots,m^2$, and $g(\mathbf{s}_i)$ is the surface measurement at the collocation point $\mathbf{s}_i,\ i = 1,\dots,n^2$. Hence, when $n > m$, we are dealing with an overdetermined problem and vice versa. This can be expressed as a linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$, where

$$a_{ij} = \omega_j \frac{d}{\|\mathbf{s}_i - \mathbf{t}_j\|_2^3}, \quad x_j = \hat{f}(\mathbf{t}_j), \quad b_i = g(\mathbf{s}_i). \tag{31}$$

Due to the ill-posedness of the underlying, continuous inverse problem, the matrix $\mathbf{A}$ is very ill-conditioned, which entails numerical instability and spurious, often oscillatory, naive solutions for noisy right hand sides. A problem of this type is traditionally solved by way of *regularisation*, but it can also be handled in a more natural and elegant fashion as a Bayesian inverse problem.

For the exerimental set-up, a "true" mass density distribution $f(t)$ was assigned on T at a depth of $d = 0.1$ (Fig. 3, left panel). The modelled signal was then discretised with $m = n = 100$ and perturbed with white noise with standard deviation $\sigma_\epsilon = 0.1$ (Fig. 3, right panel) to be used as synthetic data in the numerical experiment.

The unknown mass density distribution was modelled as a Gaussian Random Process with a Matérn 3/2 covariance kernel [37]:

$$C_{3/2}(x,y) = \sigma^2 \left(1 + \frac{\sqrt{3}\|x-y\|_2}{\lambda}\right) \exp\left(-\frac{\sqrt{3}\|x-y\|_2}{\lambda}\right), \quad \text{for} \quad x,y \in D, \tag{32}$$

[2] https://docs.pymc.io/en/v3/

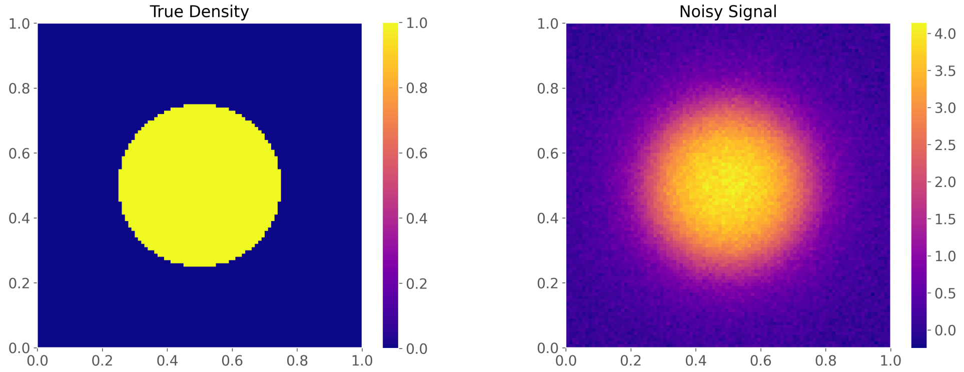


Figure 3: (Left) The "true" mass density $f(t)$ and (right) the noisy signal at $d = 0.1$, with $\sigma_\epsilon = 0.1$.

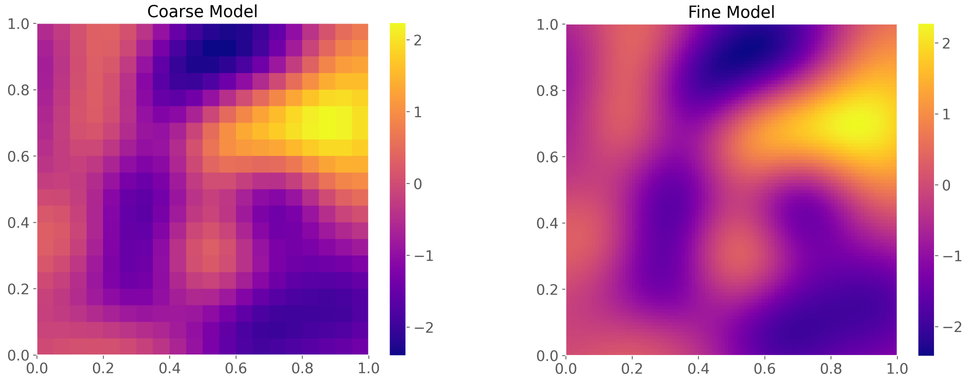


Figure 4: Random realisations of the Matérn 3/2 random process prior, used to model the unknown mass density for the coarse model with $m = 20$ (left) and the fine model with $m = 100$ (right).

where λ is the covariance length scale and σ^2 is the variance. The random field was parametrised using a truncated Karhunen-Loève (KL) expansion of $f(t)$, i.e. an expansion in terms of a finite set of independent, standard Gaussian random variables $\theta_i \sim \mathcal{N}(0, 1)$, $i = 1, \ldots, R$, given by

$$f(t, \omega) = \sum_{i=1}^{R} \sqrt{\mu_i}\phi_i(t)\theta_i(\omega). \tag{33}$$

Here, $\{\mu_i\}_{i\in\mathbb{N}}$ are the sequence of strictly decreasing real, positive eigenvalues, and $\{\phi_i\}_{i\in\mathbb{N}}$ the corresponding L^2-orthonormal eigenfunctions of the covariance operator with kernel $C_{3/2}(x, y)$.

A model hierarchy consisting of two model levels, with $m = 100$ and $m = 20$ respectively, was created. A Matern 3/2 random process with $l = 0.2$ and $\sigma^2 = 1$ was initialised on the fine model level and parametrised using KL decomposition, which was then truncated to encompass its $R = 32$ highest energy eigenmodes. It was then projected to the coarse model space (Fig. 4).

Thus, the prior distribution of the model parameters $(\theta_i)_{i=1}^R$ is $\mathcal{N}(0, I_R)$. To sample from the posterior distribution of these parameters and thus to estimate the posterior mean conditioned on the synthetic data, we used the TLDA sampler with a Random Walk Metropolis Hastings (RWMH) sampler on the coarse level. We ran 2 independent chains, each with 20000 draws, a burn-in of 5000

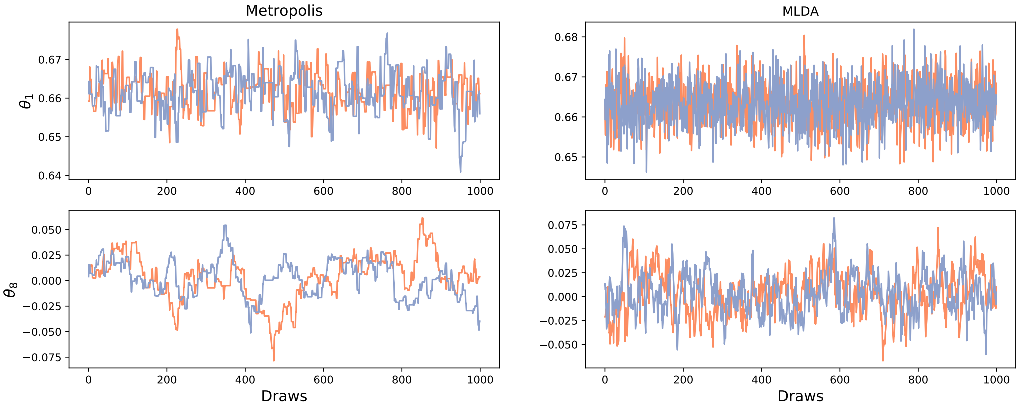


Figure 5: Traces of θ_1 (top row) and θ_8, for RWMH (left column) and MLDA (right column), respectively. Different colors represent the independent chains.

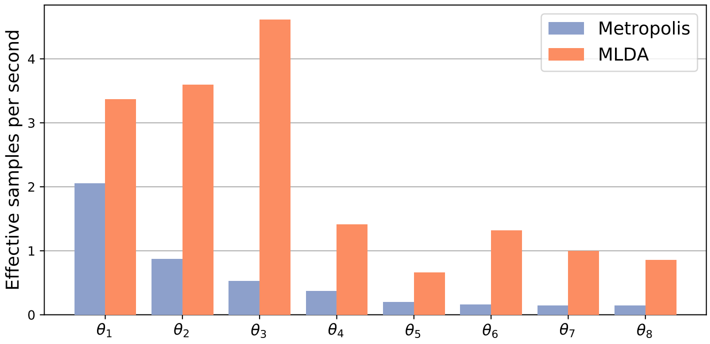


Figure 6: Algorithmic performance measured in ES/s (effective samples per second), for the eight highest energy KL coefficients $\theta_k, k = 1, \dots, 8$, for both RWMH (blue) and MLDA (red).

and a subchain length on the coarse level of 10. We also ran 2 chains using a single level RWMH sampler on the fine level with otherwise identical settings, but with no subchains. Each chain was initialised at the MAP (Maximum a Posteriori) point.

While RWMH converged to the same parameter estimates as MLDA, RWMH exhibited inferior mixing (Fig. 5) and fewer effective samples per second (Fig. 6), particularly for the higher KL coefficients.

3.2 Predator-Prey Model

The Lotka-Volterra model describes the interaction between populations of prey (N) and predators (P) over time [43]. Their interaction is described by the system of nonlinear, first order, ordinary differental equations (ODEs)

$$\frac{dN}{dt} = aN - bNP \quad \text{and} \quad \frac{dP}{dt} = cNP - dP, \quad \text{for } t > 0. \tag{34}$$

The model outputs are fully described by the parameters

$$\theta = \{N_0, P_0, a, b, c, d\},$$

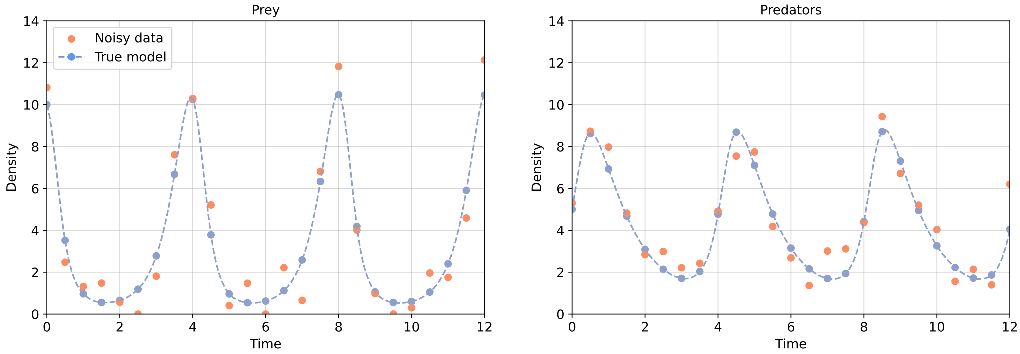


Figure 7: The true (blue) and measured (red) densities of prey (left) and predators (right).

which include the initial densities of prey and predators at time $t = 0$, and ecological parameters a, b, c, d, where broadly a is the birth rate of the prey, b is the encounter rate between prey and predators, c is the growth rate for the predators and d is the death rate of the predators. For further details on their physical interpretation see for example [3].

In this example, we wish to infer the distribution of θ, given noisy observations of prey and predator densities at discrete time intervals, i.e. $N(t^\star)$ and $P(t^\star)$ for $t^\star \in \mathcal{T}$, where $\mathcal{T} = [0, 12]$ is the domain. The observations are again synthetically generated by solving Eq. (34) with the "true" parameters

$$\theta^\star = \{10.0, 5.0, 3.0, 0.7, 0.3, 1.0\}$$

and perturbing the calculated values $N(t^\star)$ and $P(t^\star)$ with independent Gaussian noise $\epsilon \sim \mathcal{N}(0, 1)$ (Fig. 7). Our aim is to predict the mean density of predators $\mathbb{E}(P)$ over the same period.

The solutions of the ODE system in Eq. (34) can be approximated by a suitable numerical integration scheme. We use an explicit, adaptive Runge-Kutta method of order 5(4) [45]. For the fine level, we integrate over the entire time domain $\mathcal{T}_L = [0, 12]$ and use the entire dataset to compute the likelihood function, while for the coarse level, we stop integration halfway through, so that $\mathcal{T}_{L-1} = [0, 6]$, and use only the corresponding subset of the data to compute the likelihood function.

We assume that we possess some prior knowledge about the parameters, and use informed priors $N_0 \sim \mathcal{N}(10.8, 1)$, $P_0 \sim \mathcal{N}(5.3, 1)$, $a \sim \mathcal{N}(2.5, 0.5)$, $b \sim$ Inv-Gamma$(1.0, 0.5)$, $c \sim$ Inv-Gamma$(1.0, 0.5)$ and $d \sim \mathcal{N}(1.2, 0.3)$.

To demonstrate the multilevel variance reduction feature, we ran the TLDA sampler with randomisation of the subchain length as described in Section 2.3 and then compared the (multilevel) MLDA estimator in Eq. (19), which uses both the coarse and fine samples, with a standard MCMC estimator based only on the samples produced by TLDA on the fine level. In both cases, we used the two–level model hierarchy as described above and employed the Differential Evolution Markov Chain (DE-MC$_Z$) proposal [47] on the coarse level. The coarse level proposal kernel was automatically tuned during burn-in to achieve an acceptance rate between 0.2 and 0.5. The subchain length of $J_L = 15$ was chosen to balance the variances of the two contributions to the multilevel estimator (Eq. (19)), as for MLMC and MLMCMC.

Figure 8 shows the development of the total sampling error as the sampling progresses, for the sampler with and without variance reduction. Employing variance reduction clearly leads to a lower sampling error than the standard approach. Figure 9 shows the true prey and predator densities along with samples from the posterior distribution, demonstrating that the true model is encapsulated by the posterior samples, as desired.

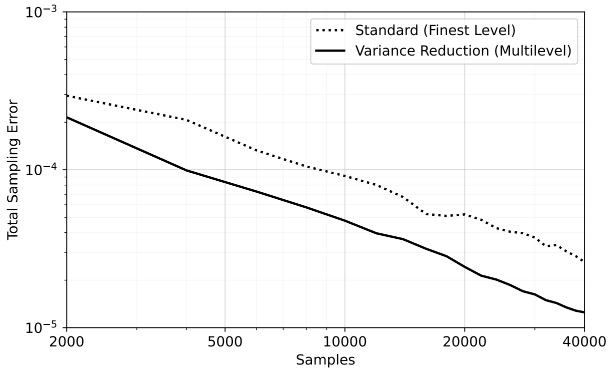


Figure 8: Development of the total sampling error as sampling progresses for the sampler with (solid red) and without (dashed, blue) variance reduction.

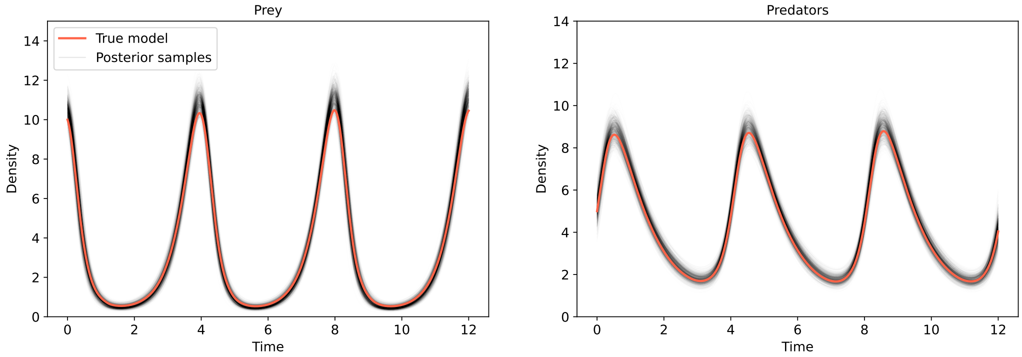


Figure 9: True model (red) and posterior samples (black).

3.3 Subsurface Flow

In this example, a simple model problem arising in subsurface flow modelling is considered. Probabilistic uncertainty quantification is of interest in various situations, for example in risk assessment of radioactive waste repositories. Moreover, this simple PDE model is often used as a benchmark for MCMC algorithms in the applied mathematics literature [35, 34, 15, 11, 10, 5]. The classical equations which govern steady-state single-phase subsurface flow in a confined aquifer are Darcy's law coupled with an incompressibility constraint

$$w + k\nabla p = g \quad \text{and} \quad \nabla \cdot w = 0, \quad \text{in} \quad D \subset \mathbb{R}^d \tag{35}$$

for $d = 1, 2$ or 3, subject to suitable boundary conditions. Here p denotes the hydraulic head of the fluid, k the permeability tensor, w the flux and g is the source term.

A typical approach to treat the inherent uncertainty in this problem is to model the permeability as a random field $k = k(x, \omega)$ on $D \times \Omega$, for some probability space $(\Omega, \mathcal{A}, \mathbb{P})$. Therefore, Eq. (35) can be written as the following PDE with random coefficients:

$$-\nabla \cdot k(x, \omega)\nabla p(x, \omega) = f(x), \quad \text{for all} \quad x \in D, \tag{36}$$

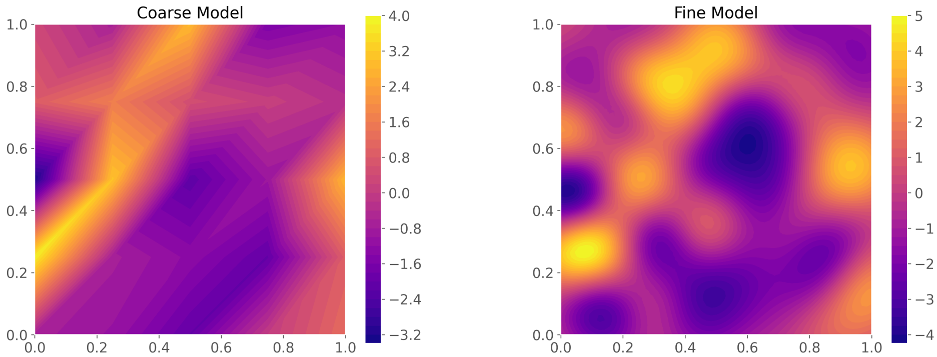


Figure 10: True log-conductivity field of the coarsest model with m_0 grid points (left) and the finest model with m_2 grid points (right).

where $f := -\nabla \cdot g$. As a synthetic example, consider the domain $D := [0,1]^2$ with $f \equiv 0$ and deterministic boundary conditions

$$p|_{x_1=0} = 0, \quad p|_{x_1=1} = 1 \quad \text{and} \quad \partial_n p|_{x_2=0} = \partial_n p|_{x_2=1} = 0. \tag{37}$$

A widely used model for the prior distribution of the permeability in hydrology is a log-Gaussian random field [15, 12, 11, 5, 29], characterised by the mean of $\log k$, here chosen to be 0, and by its covariance function, here chosen to be

$$C(x,y) := \sigma^2 \exp\left(-\frac{\|x-y\|_2^2}{2\lambda^2}\right), \quad \text{for} \quad x, y \in D, \tag{38}$$

with $\sigma = 2$ and $\lambda = 0.1$. Again, the log-Gaussian random field is parametrised using a truncated Karhunen-Loève (KL) expansion of $\log k$, i.e., an expansion in terms of a finite set of independent, standard Gaussian random variables $\theta_i \sim \mathcal{N}(0,1)$, $i = 1, \dots, R$, given by

$$\log k(x,\omega) = \sum_{i=1}^{R} \sqrt{\mu_i}\phi_i(x)\theta_i(\omega). \tag{39}$$

Again, $\{\mu_i\}_{i\in\mathbb{N}}$ are the sequence of strictly decreasing real, positive eigenvalues, and $\{\phi_i\}_{i\in\mathbb{N}}$ the corresponding L^2-orthonormal eigenfunctions of the covariance operator with kernel $C(x,y)$. Thus, the prior distribution on the parameter $\theta = (\theta_i)_{i=1}^R$ in the stochastic PDE problem (Eq. (36)) is $\mathcal{N}(0, I_R)$. In this example we chose $R = 64$.

The aim is to infer the posterior distribution of θ, conditioned on measurements of p at $M = 25$ discrete locations $x^j \in D$, $j = 1, \dots, M$, stored in the vector $d_{obs} \in \mathbb{R}^M$. Thus, the forward operator is $\mathcal{F} : \mathbb{R}^R \to \mathbb{R}^M$ with $\mathcal{F}_j(\theta_\omega) = p(x^j, \omega)$.

All finite element (FE) calculations were carried out with FEniCS [30], using piecewise linear FEs on a uniform triangular mesh. The coarsest mesh $\mathcal{T}_0$ consisted of $m_0 = 5$ grid points in each direction, while subsequent levels were constructed by two steps of uniform refinement of $\mathcal{T}_0$, leading to $m_\ell = 4^\ell(m_0 - 1) + 1$ grid points in each direction on the three grids $\mathcal{T}_\ell$, $\ell = 0, 1, 2$ (Fig. 10).

To demonstrate the excellent performance of MLDA with the AEM, synthetic data was generated by drawing a sample from the prior distribution and solving Eq. (36) with the resulting realisation of k on $\mathcal{T}_2$. To construct d_{obs}, the computed discrete hydraulic head values at $(x^j)_{j=1}^M$ were then perturbed by independent Gaussian noise, i.e. by a sample $\epsilon^* \sim \mathcal{N}(0, \Sigma_\epsilon)$ with $\Sigma_\epsilon = 0.01^2 I_M$.

To compare the "vanilla" MLDA approach to the AEM-enhanced version, we sampled the same model using identical sampling parameters, with and without AEM activated. For each approach, we

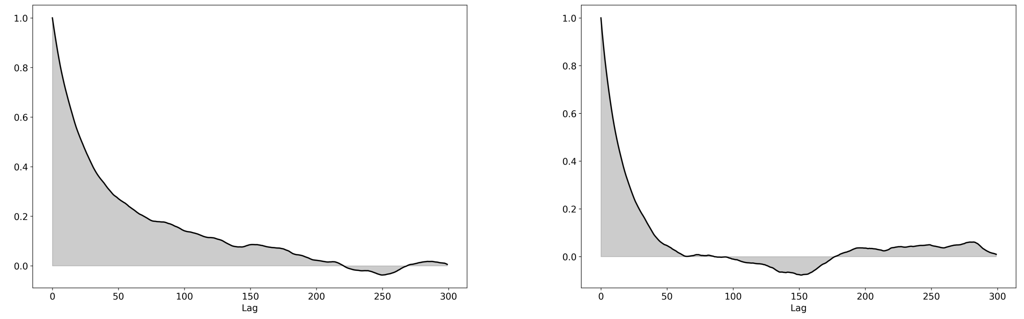


Figure 11: Autocorrelation function for θ_1 for samples without AEM (left) and with AEM (right).

sampled two independent chains, each initialised at a random point from the prior. For each chain, we drew 20000 samples plus a burn-in of 5000. We used subchain lengths $J_0 = J_1 = 5$, since that produced the best trade-off between computation time and effective sample size for MLDA with the AEM. Note that the cost of computing the subchains on the coarser levels only leads to about a 50% increase in the total cost for drawing a sample on level L. The DE-MC$_Z$ proposal [47] was employed on the coarsest level with automatic step-size tuning during burnin to achieve an acceptance rate between 0.2 and 0.5.

To assess the performance of the two approaches, the autocorrelation function (Fig. 11) and the Effective Sample Size (ESS) for each parameter were computed [48]. Since the coarsest model was quite a poor approximation of the finest, running MLDA without the Adaptive Error Model (AEM) yielded relatively poor results, with an average ESS of 326 out of 40000 samples, and strong autocorrelation. However, when the AEM was employed and otherwise using the exact same sampling parameters, we obtained an average ESS of 1012 out of 40000 samples, with correspondingly weaker autocorrelation.

Note that this particular numerical experiment was chosen to demonstrate the dramatic effect that employing the AEM can have in MLDA, thus making it possible to use multilevel sampling strategies with very crude approximate models. A FE mesh with 25 degrees of freedom is extremely coarse for a Gaussian random field with correlation length $\lambda = 0.1$, yet using the AEM it still provides an excellent surrogate for delayed acceptance. Typically much finer models are used in real applications with longer subchains on the coarser levels (cf. [15]). The AEM will be less critical in that case and MLDA will also produce good ESS without the AEM.

4 Conclusions and Future Work

In this paper, we have presented an extension of state-independent Delayed Acceptance MCMC [8], where a hierarchy of coarse MCMC samplers inform the finest sampler in a cascading fashion. If the models on the coarse levels are carefully designed, the approach can lead to significant computational savings, compared to standard single-level MCMC. A possible direction for future research would be to extend this approach further to the general Delayed Acceptance context, where also state-dependent approximations are supported. We would like to highlight that the choice of proposal on the coarsest level is free, as long as it achieves irreducibility for the coarsest distribution. We have chosen relatively simple proposals for the coarsest level, but if e.g. the gradient of the likelihood function is available, one can also employ more advanced gradient-informed proposals, such as MALA, HMC or NUTS.

The presented MLDA algorithm has clear similarities with Multilevel MCMC [15], in that it allows for any number of coarse levels and extended subchains on the coarse levels, but unlike MLMCMC, it is Markov and asymptotically unbiased, also for finite-length subchains. To achieve this quality, the algorithm must be sequential, which complicates parallelisation considerably. One remedy for this challenge, and a possible direction for future research, would be to employ pre-fetching of proposals [6]. The central idea of pre-fetching is to precompute proposal "branches" and evaluate those in parallel, since for each proposal there are only two options, namely *accept* or *reject*. Pre-fetching and evaluating entire proposal branches is significantly more computationally demanding than the strictly sequential approach and generates more waste, similar to Multiple-Try Metropolis [32], since entire branches will effectively be rejected at each step. Minimising the waste of pre-fetching while maintaining the computational gains of parallelisation constitutes a complex, probabilistic optimisation problem. This could be addressed by controlling the pre-fetching length, e.g., using a reinforcement learning agent to learn an optimal policy, and to then hedge bets on valuable pre-fetching lengths, based on the latest sampling history.

A question that remains is the optimal choice of the subchain lengths $\{J_\ell\}_{\ell=1}^L$ for the coarse levels, which is essentially the only tuning parameter in the MLDA algorithm. A good rule of thumb may be to choose the length for any level such that the cost of creating the subchain corresponds to the cost of evaluating a single proposal on the next finer level, but this is not the most rigorous approach. The question has previously been studied in the context of Multilevel Monte Carlo [9] and MLMCMC [15], and involves either computing the optimal (effective) sample size for each level for a fixed acceptable sampling error, or computing the sampling error corresponding to a fixed computational budget. A similar approach can be taken for MLDA, but with some caveats. First, the number of samples on each level is determined, not only by the subchain length on that level, but by the number of samples on the next finer level. Hence, care must be taken when choosing the subchain lengths. Second, it is non-trivial to determine the effective sample size of a level *a priori*, because of the direct correspondence with the distribution on the next finer level by way of the MLDA acceptance criterion. One possible workaround would be to determine the optimal subchain lengths adaptively by empirically determining the effective sample sizes and variances on each level during burn-in. Similarly to the pre-fetching approach outlined above, these decisions could also be outsourced to a reinforcement learning agent that would adaptively learn the optimal policy for minimising either cost or sampling error. We emphasize this question as a potential direction for future research.

Acknowledgements

MCMC sampling was completed using the `MLDA` sampler of the free and open source probabilistic programming library `PyMC`. The code is available at GitHub: https://github.com/pymc-devs/pymc.

ML was funded as part of the Water Informatics Science and Engineering Centre for Doctoral Training (WISE CDT) under a grant from the Engineering and Physical Sciences Research Council (EPSRC), grant number EP/L016214/1. TD and GM were funded by a Turing AI Fellowship (2TAFFP\100007). CF was partially funded by MBIE contract UOOX2106. The work of RS is supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany's Excellence Strategy EXC 2181/1 - 390900948 (the Heidelberg STRUCTURES Excellence Cluster).

The authors have no competing interests.

References

[1] C. Andrieu and J. Thoms. A tutorial on adaptive MCMC. *Statistics and Computing*, 18(4):343–373, 2008.

[2] Y. F. Atchadé. An Adaptive Version for the Metropolis Adjusted Langevin Algorithm with a Truncated Drift. *Methodology and Computing in Applied Probability*, 8(2):235–254, 2006.

[3] N. Bacaër. *A Short History of Mathematical Population Dynamics*. Springer, London, 2011.

[4] A. Barth, C. Schwab, and N. Zollinger. Multi-level Monte Carlo Finite Element method for elliptic PDEs with stochastic coefficients. *Numerische Mathematik*, 119(1):123–161, Sept. 2011.

[5] A. Beskos, M. Girolami, S. Lan, P. E. Farrell, and A. M. Stuart. Geometric MCMC for infinite-dimensional inverse problems. *Journal of Computational Physics*, 335:327–351, 2017.

[6] A. E. Brockwell. Parallel Markov chain Monte Carlo Simulation by Pre-Fetching. *Journal of Computational and Graphical Statistics*, 15(1):246–261, 2006.

[7] J. Charrier, R. Scheichl, and A. L. Teckentrup. Finite Element Error Analysis of Elliptic PDEs with Random Coefficients and Its Application to Multilevel Monte Carlo Methods. *SIAM Journal on Numerical Analysis*, 51(1):322–352, 2013.

[8] J. A. Christen and C. Fox. Markov chain Monte Carlo Using an Approximation. *Journal of Computational and Graphical Statistics*, 14(4):795–810, 2005.

[9] K. A. Cliffe, M. B. Giles, R. Scheichl, and A. L. Teckentrup. Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients. *Computing and Visualization in Science*, 14(1):3–15, 2011.

[10] P. R. Conrad, A. Davis, Y. M. Marzouk, N. S. Pillai, and A. Smith. Parallel local approximation MCMC for expensive models. *SIAM/ASA Journal on Uncertainty Quantification*, 6(1):339–373, 2018.

[11] P. R. Conrad, Y. M. Marzouk, N. S. Pillai, and A. Smith. Accelerating Asymptotically Exact MCMC for Computationally Intensive Models via Local Approximations. *Journal of the American Statistical Association*, 111(516):1591–1607, 2016.

[12] P. G. Constantine, C. Kent, and T. Bui-Thanh. Accelerating Markov Chain Monte Carlo with Active Subspaces. *SIAM Journal on Scientific Computing*, 38(5):A2779–A2805, 2016.

[13] T. Cui, C. Fox, and M. J. O'Sullivan. Bayesian calibration of a large-scale geothermal reservoir model by a new adaptive delayed acceptance Metropolis Hastings algorithm: Adaptive Delayed Acceptance Metropolis-Hastings algorithm. *Water Resources Research*, 47(10), 2011.

[14] T. Cui, C. Fox, and M. J. O'Sullivan. A posteriori stochastic correction of reduced models in delayed-acceptance MCMC, with application to multiphase subsurface inverse problems: Stochastic correction of reduced models in delayed-acceptance MCMC. *International Journal for Numerical Methods in Engineering*, 118(10):578–605, 2019.

[15] T. J. Dodwell, C. Ketelsen, R. Scheichl, and A. L. Teckentrup. A Hierarchical Multilevel Markov Chain Monte Carlo Algorithm with Applications to Uncertainty Quantification in Subsurface Flow. *SIAM/ASA Journal on Uncertainty Quantification*, 3(1):1075–1108, 2015.

[16] S. Duane, A. D. Kennedy, B. J. Pendleton, and D. Roweth. Hybrid Monte Carlo. *Physics Letters B*, 195(2):216–222, 1987.

[17] C. Fox, T. Cui, and M. Neumayer. Randomized reduced forward models for efficient Metropolis–Hastings MCMC, with application to subsurface fluid flow and capacitance tomography. *GEM-International Journal on Geomathematics*, 11(1):1–38, 2020.

[18] S. Geman and D. Geman. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-6(6):721–741, 1984.

[19] M. B. Giles. Multilevel Monte Carlo Path Simulation. *Operations Research*, 56(3):607–617, 2008.

[20] M. B. Giles. Multilevel Monte Carlo methods. *Acta Numerica*, 24:259 – 328, 2015.

[21] H. Haario, E. Saksman, and J. Tamminen. An Adaptive Metropolis Algorithm. *Bernoulli*, 7(2):223, 2001.

[22] A.-L. Haji-Ali, F. Nobile, L. Tamellini, and R. Tempone. Multi-index Stochastic Collocation Convergence Rates for Random PDEs with Parametric Regularity. *Foundations of Computational Mathematics*, 16(6):1555–1605, 2016.

[23] P. C. Hansen. *Discrete Inverse Problems: Insight and Algorithms*. Society for Industrial and Applied Mathematics, 2010.

[24] W. K. Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, 57(1):97–109, 1970.

[25] S. Heinrich. Multilevel Monte Carlo Methods. In *Proceedings of the Third International Conference on Large-Scale Scientific Computing-Revised Papers*, LSSC '01, pages 58–67, London, UK, 2001. Springer-Verlag.

[26] M. D. Hoffman and A. Gelman. The No-U-Turn Sampler: Adaptively setting path lengths in Hamiltonian Monte Carlo. *Journal of Machine Learning Research*, 15(1):1593–1623, 2014.

[27] A. Jasra, K. Kamatani, K. Law, and Y. Zhou. A Multi-Index Markov Chain Monte Carlo Method. *International Journal for Uncertainty Quantification*, 8(1):61–73, 2018.

[28] J. Kaipio and E. Somersalo. Statistical inverse problems: Discretization, model reduction and inverse crimes. *Journal of Computational and Applied Mathematics*, 198(2):493–504, 2007.

[29] S. Lan. Adaptive Dimension Reduction to Accelerate Infinite-Dimensional Geometric Markov Chain Monte Carlo. *Journal of Computational Physics*, 392:71–95, 2019.

[30] H. P. Langtangen and A. Logg. *Solving PDEs in Python – The FEniCS Tutorial Volume I*. Simula SpringerBriefs on Computing. Springer, 2017.

[31] J. S. Liu. *Monte Carlo Strategies in Scientific Computing*. Springer Series in Statistics. Springer, New York, 2004.

[32] J. S. Liu, F. Liang, and W. H. Wong. The Multiple-Try Method and Local Optimization in Metropolis Sampling. *Journal of the American Statistical Association*, 95(449):121–134, 2000.

[33] M. B. Lykkegaard, T. J. Dodwell, and D. Moxey. Accelerating uncertainty quantification of groundwater flow modelling using a deep neural network proxy. *Computer Methods in Applied Mechanics and Engineering*, 383:113895, 2021.

[34] Y. M. Marzouk and H. N. Najm. Dimensionality reduction and polynomial chaos acceleration of Bayesian inference in inverse problems. *Journal of Computational Physics*, 228(6):1862–1902, 2009.

[35] Y. M. Marzouk, H. N. Najm, and L. A. Rahn. Stochastic spectral methods for efficient Bayesian solution of inverse problems. *Journal of Computational Physics*, 224(2):560–586, 2007.

[36] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of State Calculations by Fast Computing Machines. *The Journal of Chemical Physics*, 21(6):1087–1092, 1953.

[37] C. E. Rasmussen and C. K. I. Williams. *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press, Cambridge, MA, 2006.

[38] G. O. Roberts and J. 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.

[39] G. O. Roberts and J. S. Rosenthal. General state space Markov chains and MCMC algorithms. *Probability Surveys*, 1:20–71, 2004.

[40] G. O. Roberts and J. S. Rosenthal. Coupling and Ergodicity of Adaptive Markov Chain Monte Carlo Algorithms. *Journal of Applied Probability*, 44(2):458–475, 2007.

[41] G. O. Roberts and J. S. Rosenthal. Examples of Adaptive MCMC. *Journal of Computational and Graphical Statistics*, 18(2):349–367, 2009.

[42] G. O. Roberts and R. L. Tweedie. Exponential Convergence of Langevin Distributions and Their Discrete Approximations. *Bernoulli*, 2(4):341, 1996.

[43] L. L. Rockwood and J. W. Witt. *Introduction to population ecology*. Wiley Blackwell, Chichester, West Sussex, UK, 2nd edition, 2015.

[44] L. Seelinger, A. Reinarz, L. Rannabauer, M. Bader, P. Bastian, and R. Scheichl. High performance uncertainty quantification with parallelized multilevel Markov chain Monte Carlo. SC '21, New York, NY, USA, 2021. Association for Computing Machinery.

[45] S. Strogatz. *Nonlinear dynamics and chaos: With applications to physics, biology, chemistry, and engineering*. Studies in Nonlinearity. Westview Press, Cambridge, MA, 2007.

[46] A. L. Teckentrup, R. Scheichl, M. B. Giles, and E. Ullmann. Further analysis of multilevel Monte Carlo methods for elliptic PDEs with random coefficients. *Numerische Mathematik*, 125(3):569–600, 2013.

[47] C. J. F. ter Braak and J. A. Vrugt. Differential Evolution Markov Chain with snooker updater and fewer chains. *Statistics and Computing*, 18(4):435–446, 2008.

[48] A. Vehtari, A. Gelman, D. Simpson, B. Carpenter, and P.-C. Bürkner. Rank-normalization, folding, and localization: An improved $\widehat{R}$ for assessing convergence of MCMC. *Bayesian Analysis*, 16(2):667–718, 2020.

[49] J. A. Vrugt, C. ter Braak, C. Diks, B. A. Robinson, J. M. Hyman, and D. Higdon. Accelerating Markov Chain Monte Carlo Simulation by Differential Evolution with Self-Adaptive Randomized Subspace Sampling. *International Journal of Nonlinear Sciences and Numerical Simulation*, 10(3):273–290, 2009.

[50] Q. Zhou, Z. Hu, Z. Yao, and J. Li. A Hybrid Adaptive MCMC Algorithm in Function Spaces. *SIAM/ASA Journal on Uncertainty Quantification*, 5(1):621–639, 2017.

5. Accelerating Uncertainty Quantification of Groundwater Flow Modelling Using a Deep Neural Network Proxy

This journal paper (Lykkegaard, Dodwell and Moxey, 2021), published in Computer Methods in Applied Mechanics and Engineering in May 2021, explores a novel approach to multilevel MCMC sampling for Bayesian inverse problems. Our technique exploits a deep neural network as a fast proxy model to generate MCMC proposals with very low computational cost. The presented modified Delayed Acceptance algorithm is strongly related to the MLDA algorithm presented in Chapter 4. In fact, it can be formulated as a special case of the *Randomised-Length-Subchain Surrogate Transition* (RST, Algorithm 3), where the probability mass function over subchain length is the negative binomial distribution (see e.g. Grinstead and Snell (1997)), i.e. $n \sim \mathbf{NB}(l_\alpha, \alpha)$, where n is the (randomised) MLDA subchain length, α is the acceptance rate and l_α is the predefined *offset length*, describing the required number of accepted proposals before terminating the subchain. The methodology of using a deep neural network as a proxy model in the context of the MLDA sampler is demonstrated in the context of two synthetic groundwater flow examples.

The idea was conceived by Tim Dodwell and me. I developed the computer code, conducted the experiments and wrote the paper. Tim Dodwell and David Moxey provided feedback during the research process. All authors contributed to the editing.

Available online at www.sciencedirect.com

ScienceDirect

Comput. Methods Appl. Mech. Engrg. 383 (2021) 113895

Computer methods in applied mechanics and engineering

www.elsevier.com/locate/cma

Accelerating uncertainty quantification of groundwater flow modelling using a deep neural network proxy

Mikkel B. Lykkegaard[a,*], Tim J. Dodwell[a,b], David Moxey[a]

[a] *College of Engineering, Mathematics and Physical Sciences, University of Exeter, UK*
[b] *The Alan Turing Institute, London, NW1 2DB, UK*

Received 22 June 2020; received in revised form 21 December 2020; accepted 26 April 2021
Available online 15 May 2021

Abstract

Quantifying the uncertainty in model parameters and output is a critical component in model-driven decision support systems for groundwater management. This paper presents a novel algorithmic approach which fuses Markov Chain Monte Carlo (MCMC) and Machine Learning methods to accelerate uncertainty quantification for groundwater flow models. We formulate the governing mathematical model as a Bayesian inverse problem, considering model parameters as a random process with an underlying probability distribution. MCMC allows us to sample from this distribution, but it comes with some limitations: it can be prohibitively expensive when dealing with costly likelihood functions, subsequent samples are often highly correlated, and the standard Metropolis–Hastings algorithm suffers from the curse of dimensionality. This paper designs a Metropolis–Hastings proposal which exploits a deep neural network (DNN) approximation of a groundwater flow model, to significantly accelerate MCMC sampling. We modify a delayed acceptance (DA) model hierarchy, whereby proposals are generated by running short subchains using an inexpensive DNN approximation, resulting in a decorrelation of subsequent fine model proposals. Using a simple adaptive error model, we estimate and correct the bias of the DNN approximation with respect to the posterior distribution on-the-fly. The approach is tested on two synthetic examples; a isotropic two-dimensional problem, and an anisotropic three-dimensional problem. The results show that the cost of uncertainty quantification can be reduced by up to 50% compared to single-level MCMC, depending on the precomputation cost and accuracy of the employed DNN.

Keywords: Groundwater flow; Uncertainty quantification; Markov chain Monte Carlo; Surrogate models; Deep neural networks

1. Introduction

Modelling of groundwater flow and transport is an important decision support tool when, for example, estimating the sustainable yield of an aquifer or remediating groundwater pollution. However, the input parameters for mathematical models of groundwater flow (such as subsurface transmissivity and boundary conditions) are often impossible to determine fully or accurately, and are hence subject to various uncertainties. In order to make informed decisions, it is of critical importance to decision makers to obtain robust and unbiased estimates of the total model

[*] Corresponding author.
E-mail addresses: m.lykkegaard@exeter.ac.uk (M.B. Lykkegaard), t.dodwell@exeter.ac.uk (T.J. Dodwell), d.moxey@exeter.ac.uk (D. Moxey).

https://doi.org/10.1016/j.cma.2021.113895

uncertainty, which in turn is a product of the uncertainty of these input parameters [1]. A popular way to achieve this, in relation to groundwater flow or any inverse problem in general, is stochastic or Bayesian modelling [2–4]. In this context, a probability distribution, the *prior*, is assigned to the input parameters, in accordance with any readily available information. Given some real-world measurements corresponding to the model outputs (e.g. sparse spatial measurements of hydraulic head, Darcy flow or concentration of pollutants), it is possible to reduce the overall uncertainty and obtain a better representation of the model by conditioning the prior distribution on this data. The result is a distribution of the model input parameters given data, which is also referred to as the *posterior*.

Obtaining samples from the posterior distribution directly is not possible for all but the simplest of problems. A popular approach for generating samples is the Metropolis–Hastings type *Markov Chain Monte Carlo* (MCMC) method [5]. Samples are generated by a sequential process. First, given a current sample, a new proposal for the input parameters is made using a so-called proposal distribution. Evaluating the model with this new set of parameters, a *likelihood* is computed — a measure of misfit between the model outputs and the data. The likelihoods of the proposed and current samples are then compared. Based on this comparison, the proposal is either accepted or rejected, and the whole process is repeated, generating a Markov chain of probabilistically feasible input parameters. The key point is that the distribution of samples in the chain converges to the *posterior* – the distribution of input parameters given the data [5]. This relatively simple algorithm can lead to extremely expensive Bayesian computations for three key reasons. First, each step of the chain requires the evaluation of (often) an expensive mathematical model. Second, the sequential nature of the algorithm means subsequent samples are often highly correlated — even repeated if a step is rejected. Therefore the chains must often be very long to obtain good statistics on the distribution of outputs of the model. Third, without special care, the approach does not generally scale well to large numbers of uncertain input parameters; the so-called curse of dimensionality. Addressing these scientific challenges is at the heart of modern research in MCMC algorithms. As with this paper there is a particular focus on developing novel and innovative proposal distributions, which seek to de-correlate adjacent samples and limit the computational burden of evaluating expensive models.

Broadly in the literature, simple Darcy type models and other variants of the diffusion equation have long been a popular toy example problems for demonstrating MCMC methodologies in the applied mathematics community (see e.g. [6–8]). There appears to be much less interest in MCMC in the applied groundwater modelling community. This may be because of the computational cost of running MCMC on highly parametrised, expensive models, or the lack of an easy-to-use MCMC software framework, akin to the parameter estimation toolbox PEST [9].

An exciting approach to significantly reduce the computational cost has been proposed in multi-level, multi-fidelity and Delayed Acceptance (DA) MCMC methods. In each case, to alleviate computational cost, a hierarchy of models is established, consisting of a fine model and (possibly multiple) coarse, computationally cheap approximations. Typically, the coarser models are finite element solutions of the PDE on a mesh with a coarser resolution, but as we show in this paper, can be taken to be any general approximation similar to the multi-fidelity philosophy [10]. Independent of the approach, the central idea is the same: to obtain significant efficiency gains by exploiting approximate coarse models to generate 'good' proposals cheaply, using additional accept/reject steps to filter out highly unlikely proposals before evaluating the fine, expensive model. Previous studies of two-stage approaches include [11] who modelled multi-phase flow with coarse level proposals evaluated by a coarse-mesh single-phase flow model (an idea that was developed further in [12]), [13] and [14]. We note that the latter of which, instead of simply using a coarser discretisation, implemented a data-driven polynomial chaos expansion as a surrogate model. We intend to demonstrate how the development of novel techniques in MCMC and machine learning can be combined to help realise the potential of MCMC in this field.

In this work, we propose a combination of multiple cutting-edge MCMC techniques to allow for efficient inversion and uncertainty quantification of groundwater flow. We propose an improved delayed acceptance (DA) MCMC algorithm, adapted from the approach proposed by [15]. In our case, similarly to multi-level MCMC [7], proposals are generated by computing a subchain using a Deep Neural Network (DNN) as an approximate model — leading to cheaply computed, decorrelated proposals passed on to the fine model. For our first example, the subchain is driven by the preconditioned Crank–Nicolson (pCN) proposal distribution [16] to ensure the proposed Metropolis–Hastings algorithm is robust with respect to the dimension of the uncertain parameter space. For our second example, proposals for the subchains are generated using the Adaptive Metropolis (AM) proposal [17], since the posterior distribution in this case is highly non-spherical and multiple parameters are correlated. Finally, we propose an enhanced error model, in which the DNN is trained by sampling the prior distribution, yet the bias of the approximation is adaptively estimated and corrected on-the-fly by testing the approximations against the full model in an adaptive delayed acceptance setting [18].

2. Preliminaries

In this section we briefly introduce the forward model, defining the governing equations underpinning groundwater flow and their corresponding weak form, enabling us to solve the equations using FEM methods. We then formulate our model as a Bayesian inverse problem with random input parameters, effectively resulting in a stochastic model, which can be accurately characterised by sampling from the posterior distribution of parameters using MCMC. The simple Metropolis–Hastings MCMC algorithm is then introduced and extended with the preconditioned Crank–Nicolson (pCN) and Adaptive Metropolis (AM) transition kernels.

2.1. Governing equations for groundwater flow

Consider steady groundwater flow in a confined, inhomogeneous aquifer which occupies the domain Ω with boundary Γ. Assuming that water is incompressible, the governing equations for groundwater flow can be written as the scalar elliptic partial differential equation:

$$-\nabla \cdot (-T(\mathbf{x})\nabla h(\mathbf{x})) = g(\mathbf{x}) \quad \text{for all} \quad \mathbf{x} \in \Omega \tag{1}$$

subject to boundary conditions on $\Gamma = \Gamma_N \cup \Gamma_D$ defined by the constraint equations

$$h(\mathbf{x}) = h_D(\mathbf{x}) \quad \text{on } \Gamma_D \quad \text{and} \quad (-T(\mathbf{x})\nabla h(\mathbf{x})) \cdot \boldsymbol{n} = q_N(\mathbf{x}) \quad \text{on } \Gamma_N. \tag{2}$$

Here $T(\mathbf{x})$ is the heterogeneous, depth-integrated transmissivity, $h(\mathbf{x})$ is hydraulic head, $h_D(\mathbf{x})$ is fixed hydraulic head at boundaries with Dirichlet constraints, $g(\mathbf{x})$ is fluid sources and sinks, $q(\mathbf{x})$ is Darcy velocity, $q_N(\mathbf{x})$ is Darcy velocity across boundaries with Neumann constraints and $\Gamma_D \subset \partial\Omega$ and $\Gamma_N \subset \partial\Omega$ define the boundaries comprising of Dirichlet and Neumann conditions, respectively. Following standard FEM practice (see e.g. [19]), Eq. (1) is converted into weak form by multiplying by an appropriate test function $w \in H^1(\Omega)$ and integrating by parts, so that

$$\int_\Omega \nabla w \cdot (T(\mathbf{x})\nabla h) \, d\mathbf{x} + \int_{\Gamma_N} w \, q_N(\mathbf{x}) \, ds = \int_\Omega w \, g(\mathbf{x}) \, d\mathbf{x}, \quad \forall w \in H^1(\Omega), \tag{3}$$

where $H^1(\Omega)$ is the Hilbert space of weakly differentiable functions on Ω. To approximate the hydraulic head solution $h(\boldsymbol{x})$, a finite element space $V_\tau \subset H^1(\Omega)$ on a finite element mesh $\mathcal{Q}_\tau(\Omega)$. This is defined by a basis of piecewise linear Lagrange polynomials $\{\phi_i(\mathbf{x})\}_{i=1}^M$, associated with each of the M finite element nodes. As a result (3) can be rewritten as a system of sparse linear equations

$$\mathbf{Ah} = \mathbf{b} \quad \text{where} \quad A_{ij} = \int_\Omega \nabla\phi_i \cdot T(\mathbf{x})\nabla\phi_j(\mathbf{x}) \, d\mathbf{x} \quad \text{and} \tag{4}$$

$$b_i = \int_\Omega \phi_i(\mathbf{x}) \, g(\mathbf{x}) \, d\mathbf{x} - \int_{\Gamma_N} \phi_i(\mathbf{x}) q_N(\mathbf{x}) \, ds, \tag{5}$$

where $\mathbf{A} \in \mathbb{R}^{M\times M}$ and $\mathbf{b} \in \mathbb{R}^M$ are the global stiffness matrix and load vector, respectively. The vector $\mathbf{h} := [h_1, h_2, \dots, h_M] \in \mathbb{R}^M$ is the solution vector of hydraulic head at each node within the finite element mesh so that $h(\mathbf{x}) = \sum_{i=1}^M h_i\phi_i(\mathbf{x})$. In our numerical experiments, these equations are solved using the open source general-purpose FEM framework `FEniCS` [20]. While there are well-established groundwater simulation software packages available, such as MODFLOW [21] and FEFLOW [19], FEniCS was chosen because of its flexibility and ease of integration with other software and analysis codes.

2.2. Aquifer transmissivity

The aquifer transmissivity $T(\mathbf{x})$ is not known everywhere on the domain, therefore a typical approach is to model it as a log-Gaussian random field. There exists extensive literature on modelling groundwater flow transmissivity using log-Gaussian random fields (see e.g. [22,23,14]). Whilst this may not always prove a good model, particularly in cases with highly correlated extreme values and/or preferential flow paths [24,25] as seen when considering faults and other discontinuities [26,27], the log-Gaussian distribution remains relevant for modelling transmissivity in a range of aquifers [28,29,14].

Our starting point is a covariance operator with kernel $C(\mathbf{x}, \mathbf{y})$, which defines the correlation structure of the uncertain transmissivity field. For our numerical experiments, we consider the ARD (Automatic Relevance Determination) squared exponential kernel, a generalisation of the 'classic' squared exponential kernel, which allows for handling directional anisotropy:

$$C(\mathbf{x}, \mathbf{y}) = \exp\left(-\frac{1}{2}\sum_{j=1}^{d}\left(\frac{x_j - y_j}{l_j}\right)^2\right), \tag{6}$$

where d is the spatial dimensionality of the problem and $\boldsymbol{l} \in \mathbb{R}^d$ is a vector of lengths scales corresponding to each spatial dimension. We emphasise that the covariance kernel is a *modelling choice*, and that different options are available, such as the Matern kernel which offers additional control over the smoothness of the field.

In our work, transmissivity was modelled as a discrete log-Gaussian random field expanded in an orthogonal eigenbasis with k Karhunen–Loève (KL) eigenmodes. To achieve this we construct a covariance matrix $\mathbf{C} \in \mathbb{R}^{M\times M}$, where entries are given by $C_{ij} = C(\mathbf{x}_i, \mathbf{x}_j)$ for each pair of nodal coordinates within the finite element mesh $i, j = 1, \dots, M$. Once constructed, the largest k eigenvalues $\{\lambda_i\}_{i=1}^k$ and associated eigenvectors $\{\boldsymbol{\psi}_i\}_{i=1}^k$ of $\mathbf{C}$ can be computed. The transmissivity at the nodes $\mathbf{t} := [t_1, t_2, \dots, t_M]$, is given by

$$\log \boldsymbol{t} = \boldsymbol{\mu} + \sigma\, \boldsymbol{\Psi}\boldsymbol{\Lambda}^{\frac{1}{2}}\boldsymbol{\theta}, \quad \text{where} \quad \boldsymbol{\Lambda} = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_k \end{bmatrix} \quad \text{and} \quad \boldsymbol{\Psi} = [\boldsymbol{\psi}_1, \boldsymbol{\psi}_2, \dots, \boldsymbol{\psi}_k], \tag{7}$$

where $\boldsymbol{\mu}$ defines the log of the mean transmissivity field, σ is a scalar parametrising the variance and $\boldsymbol{\theta}$ is a vector of Gaussian random variables such that $\boldsymbol{\theta} \sim \mathcal{N}(0, \mathbb{I}_k)$ as in [30]. The random field can be interpolated from nodal values across Ω, using the shape functions $\{\phi_i(\mathbf{x})\}_{i=1}^M$ so that $T(\mathbf{x}) = \sum_{i=1}^M t_i\phi_i(\mathbf{x})$.

Truncating the KL eigenmodes at the kth mode limits the amount of small scale features that can be represented. This, along with interpolating the field, has a smoothing effect on the recovered transmissivity fields, which may or may not be desirable, depending on the application. Fig. 1 shows some examples of realisations of Gaussian random fields with a square exponential kernel, which illustrates the effect of the covariance length scale l and the number of admitted KL eigenmodes k. For relatively large length scales l, there is a limit to k, above which adding higher frequency eigenvalues does not provide any additional information. In this context, the proportion of signal energy encompassed by the truncation can be understood as the ratio between the sum of truncated eigenvalues and the sum of all eigenvalues: $\sum_{i=1}^k \lambda_i / \sum_{j=1}^M \lambda_j$.

2.3. The Bayesian inverse problem

To setup the Bayesian inverse problem and thereby quantify the uncertainty in the transmissivity field $T(\mathbf{x})$, the starting point is to define a statistical model which describes distribution of the mismatch between observations and model predictions. The observations are expressed in a single vector $\mathbf{d}_{obs} \in \mathbb{R}^m$ and for a given set of model input parameters $\boldsymbol{\theta}$, the model's prediction of the data is defined by the *forward map*, $\mathcal{F}(\boldsymbol{\theta}) : \mathbb{R}^k \to \mathbb{R}^m$. The statistical model assumes the connection between model and observations through the relationship

$$\boldsymbol{d}_{\text{obs}} = \mathcal{F}(\boldsymbol{\theta}) + \boldsymbol{\epsilon} \tag{8}$$

where we take $\boldsymbol{\epsilon} \sim N(0, \boldsymbol{\Sigma}_\epsilon)$ which represents the uncertainty of the connection between model and data, capturing both model mis-specification and measurement noise as sources of this uncertainty.

The backbone of a Bayesian approach is Bayes' theorem, which allows for computing *posterior* beliefs of model parameters using both *prior* beliefs and *observations*. Bayes' theorem states that the posterior probability of a parameter realisation $\boldsymbol{\theta}$ given data $\boldsymbol{d}_{\text{obs}}$ can be computed as

$$\pi(\boldsymbol{\theta}|\boldsymbol{d}_{\text{obs}}) = \frac{\pi_0(\boldsymbol{\theta})\mathcal{L}(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta})}{\pi(\boldsymbol{d}_{\text{obs}})} \tag{9}$$

where $\pi(\boldsymbol{\theta}|\boldsymbol{d}_{\text{obs}})$ is referred to as the *posterior distribution*, $\mathcal{L}(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta})$ is called the *likelihood*, $\pi_0(\boldsymbol{\theta})$ the *prior distribution* and

$$\pi(\boldsymbol{d}_{\text{obs}}) = \int_\Theta \pi(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta})d\boldsymbol{\theta} = \int_\Theta \pi_0(\boldsymbol{\theta})\mathcal{L}(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta})d\boldsymbol{\theta} \tag{10}$$

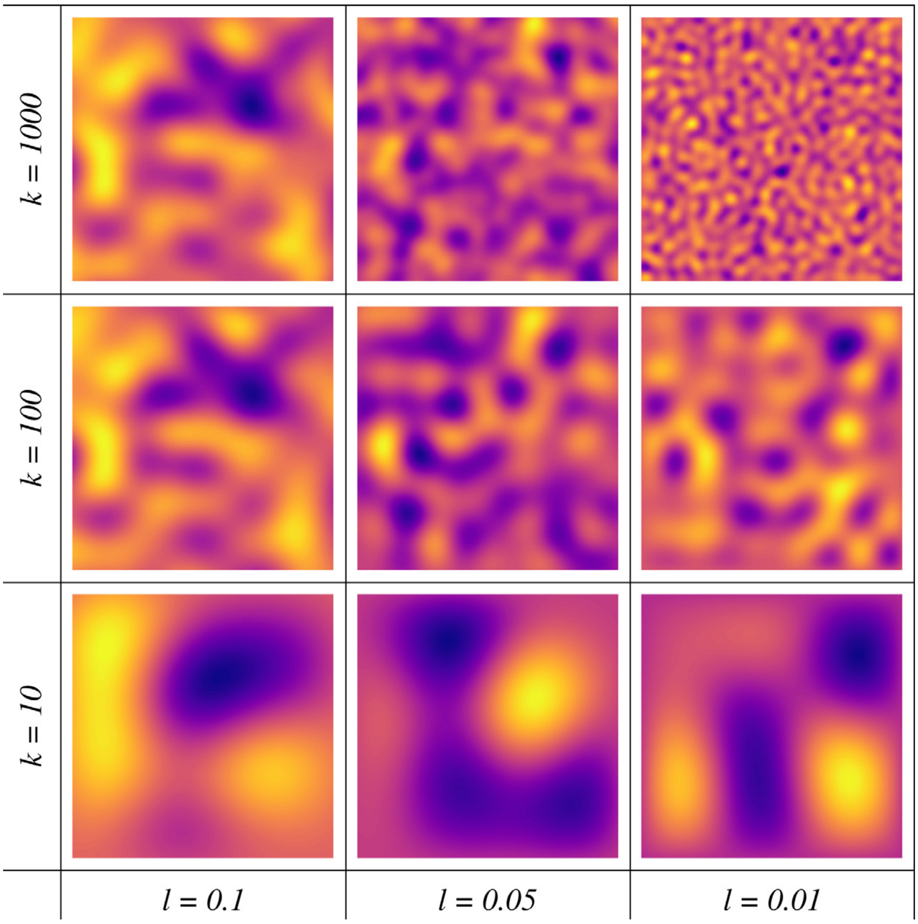


Fig. 1. A selection of Gaussian random process realisations for $\boldsymbol{x} \in [0, 1]^2$, with a square exponential kernel using different covariance length scales l and number of KL eigenmodes k. All displayed realisations were generated using the same appropriately truncated random vector $\boldsymbol{\xi}$ with identical eigenvectors for each l.

is a normalising constant, sometimes referred to as the *evidence*. In most cases this integral does not have a closed-form solution and is infeasible to estimate numerically in most real-world applications, particularly when the dimension of the unknown parameter space is large and the evaluation of the model (required to compute $\mathcal{L}(\mathbf{d}_{obs}|\boldsymbol{\theta})$) is computationally expensive. A family of methods called Markov Chain Monte Carlo (MCMC) are often employed to approximate the solution [31]. Importantly MCMC, whilst computationally expensive, allows indirect sampling from the posterior distribution and avoids the explicit need to estimate (10). Moreover, it can be designed to be independent of the dimension of the parameter space and has no embedded unquantifiable bias. In this paper we consider a subclass of MCMC methods called the Metropolis–Hastings [32,33,5] algorithm, which is described in Algorithm 1. The algorithm generates a Markov chain $\{\boldsymbol{\theta}^{(n)}\}_{n\in\mathbb{N}}$ with a distribution converging to $\pi(\boldsymbol{d}_{obs}|\boldsymbol{\theta})$. It is difficult (often impossible) to sample directly from the posterior, hence at each step, at position $\boldsymbol{\theta}^{(i)}$ in the chain, a proposal is made $\boldsymbol{\theta}'$ from a simpler known (proposal) distribution $q(\boldsymbol{\theta}'|\boldsymbol{\theta}^{(i)})$. An accept/reject step then determines whether the proposal comes from (probabilistically) the posterior distribution or not. This accept/reject step is a achieved by essentially computing the ratio of the densities of the current state to the proposal. To do this we exploit Bayes's Theorem. The key observation in MCMC is that the normalising constant $\pi(\boldsymbol{d}_{\text{obs}})$ is independent of $\boldsymbol{\theta}$, and so

$$\pi(\boldsymbol{\theta}|\boldsymbol{d}_{\text{obs}}) \propto \pi_0(\boldsymbol{\theta})\mathcal{L}(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta}). \tag{11}$$

Therefore when comparing the ratio of the densities, the normalising constant (since independent of $\boldsymbol{\theta}$) cancels.

Algorithm 1: Metropolis–Hastings Algorithm

```
1. Given a parameter realisation $\boldsymbol{\theta}_i$ and a transition kernel $q(\boldsymbol{\theta}'|\boldsymbol{\theta}_i)$, generate a proposal $\boldsymbol{\theta}'$.
2. Compute the likelihood ratio between the proposal and the previous realisation:
       $\alpha = \min \left\{ 1, \frac{\pi_0(\boldsymbol{\theta}')\mathcal{L}(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta}')}{\pi_0(\boldsymbol{\theta}^{(i)})\mathcal{L}(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta}^{(i)})} \frac{q(\boldsymbol{\theta}^{(i)}|\boldsymbol{\theta}')}{q(\boldsymbol{\theta}'|\boldsymbol{\theta}^{(i)})} \right\}$
3. If $u \sim U(0,1) > \alpha$ then set $\boldsymbol{\theta}^{(i+1)} = \boldsymbol{\theta}^{(i)}$, otherwise, set $\boldsymbol{\theta}^{(i+1)} = \boldsymbol{\theta}'$.
```

In our model problem, the prior density of the parameters $\pi_0(\boldsymbol{\theta})$ represents the available *a priori* knowledge about the transmissivity of the aquifer. From our statistical model (8) we see that our $\boldsymbol{d}_{obs} - \mathcal{F}(\boldsymbol{\theta}) \sim \mathcal{N}(0, \boldsymbol{\Sigma}_\epsilon)$, hence

$$\mathcal{L}(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta}) = \exp\left(-\frac{1}{2}(\mathcal{F}(\boldsymbol{\theta}) - \boldsymbol{d}_{\text{obs}})^{\intercal} \boldsymbol{\Sigma}_e^{-1} (\mathcal{F}(\boldsymbol{\theta}) - \boldsymbol{d}_{\text{obs}})\right). \tag{12}$$

Importantly we note that for each step of the Metropolis–Hastings algorithms we are required to compute $\mathcal{L}(\boldsymbol{d}_{obs}|\boldsymbol{\theta}')$. This requires the evaluation of the forward mapping $\mathcal{F}(\boldsymbol{\theta}')$ which can be computationally expensive. Moreover, due to the sequential nature of MCMC-based approaches, consecutive samples are correlated and hence many samples are required to obtain good statistics on the outputs.

The proposal distribution $q(\boldsymbol{\theta}'|\boldsymbol{\theta}^{(n)})$ is the key element which drives the Metropolis–Hastings algorithm and control the effectiveness of the algorithm. A common choice is a simple random walk, for which $q_{\text{RW}}(\boldsymbol{\theta}'|\boldsymbol{\theta}^{(i)}) = \mathcal{N}(\boldsymbol{\theta}^{(i)}, \boldsymbol{\Sigma})$, yet as shown in [34], the basic random walk does not lead to a convergence that is independent of the input dimension m. Better choices would be the *preconditioned Crank–Nicolson* proposal (pCN, [16]), which has dimension independent acceptance probability, or the *Adaptive Metropolis* algorithm (AM, [17]), which adaptively aligns the proposal distribution to the posterior during sampling. Moreover, unlike the Metropolis-Adjusted Langevin Algorithm (MALA), No-U-Turn Sampler (NUTS) and Hamiltonian Monte Carlo, none of these proposals rely on gradient information, which can be infeasible to compute for expensive forward models.

To generate a proposal using the pCN transition kernel, one computes

$$\boldsymbol{\theta}' = \sqrt{1-\beta^2}\,\boldsymbol{\theta}^{(i)} + \beta\boldsymbol{\xi} \tag{13}$$

where $\boldsymbol{\xi}$ is a random sample from the prior distribution, $\boldsymbol{\xi} \sim \mathcal{N}(0, \boldsymbol{\Sigma})$. This expression corresponds to the transition kernel $q_{\text{pCN}}(\boldsymbol{\theta}'|\boldsymbol{\theta}^{(i)}) = \mathcal{N}(\sqrt{1-\beta^2}\boldsymbol{\theta}^{(i)}, \beta^2\boldsymbol{\Sigma})$. Moreover, for the pCN transition kernel, the acceptance probability simplifies to

$$\alpha = \min\left\{1, \frac{\mathcal{L}(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta}')}{\mathcal{L}(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta}^{(i)})}\right\} \quad \text{following the identity} \quad \frac{p_0(\boldsymbol{\theta}^{(i)})}{p_0(\boldsymbol{\theta}')} = \frac{q_{\text{pCN}}(\boldsymbol{\theta}^{(i)}|\boldsymbol{\theta}')}{q_{\text{pCN}}(\boldsymbol{\theta}'|\boldsymbol{\theta}^{(i)})} \tag{14}$$

as given in [7]. Additional details of derivation of the pCN proposal are provided in Appendix A.

Similarly, to generate a proposal using the AM transition kernel, we draw a random sample

$$\boldsymbol{\theta}' \sim \mathcal{N}(\boldsymbol{\theta}^{(i)}, \boldsymbol{\Sigma}^{(i)}) \tag{15}$$

where $\boldsymbol{\Sigma}^{(i)}$ is an iteratively updated covariance structure

$$\boldsymbol{\Sigma}^{(i)} = \begin{cases} \boldsymbol{\Sigma}^{(0)}, & \text{if } i \leq i_0, \\ s_d \operatorname{Cov}(\boldsymbol{\theta}^{(0)}, \boldsymbol{\theta}^{(1)} \dots \boldsymbol{\theta}^{(i)}) + s_d\,\gamma\,\mathbb{I}_d, & \text{otherwise.} \end{cases}$$

Hence, proposals are drawn from a distribution with an initial covariance $\boldsymbol{\Sigma}^{(0)}$ for a given period i_0, after which adaptivity is 'switched on', and used for the remaining samples. The adaptive covariance $\boldsymbol{\Sigma}^{(i)} = s_d \operatorname{Cov}(\boldsymbol{\theta}^{(0)}, \boldsymbol{\theta}^{(1)} \dots \boldsymbol{\theta}^{(i)}) + s_d\,\gamma\,\mathbb{I}_d$ can be constructed iteratively during sampling using the following recursive formula:

$$\boldsymbol{\Sigma}^{(i+1)} = \frac{i-1}{i}\boldsymbol{\Sigma}^{(i)} + \frac{s_d}{i}(i\bar{\boldsymbol{\theta}}^{(i-1)}\bar{\boldsymbol{\theta}}^{(i-1)\intercal} - (i+1)\bar{\boldsymbol{\theta}}^{(i)}\bar{\boldsymbol{\theta}}^{(i)\intercal} + \boldsymbol{\theta}^{(i)}\boldsymbol{\theta}^{(i)\intercal} + \gamma\mathbb{I}_d) \tag{16}$$

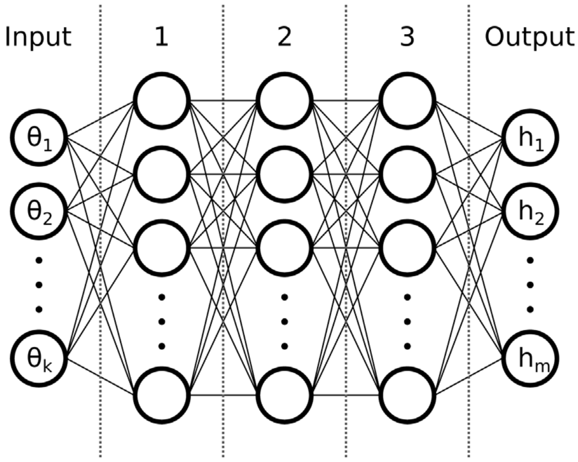


Fig. 2. Graph showing the structure of a feedforward DNN.

where $\bar{\cdot}$ is the arithmetic mean, $s_d = 2.4^2/d$ is a scaling parameter, d is the dimension of the proposal distribution and γ is a parameter which prevents $\boldsymbol{\Sigma}_i$ from becoming singular [17]. This, on the other hand, corresponds to the transition kernel $q_{\mathrm{AM}}(\boldsymbol{\theta}'|\boldsymbol{\theta}^{(0)}, \boldsymbol{\theta}^{(1)} \ldots \boldsymbol{\theta}^{(i)}) = \mathcal{N}(\boldsymbol{\theta}^{(i)}, \boldsymbol{\Sigma}^{(i)})$, which is not guaranteed to be ergodic, since it will depend on the history of the chain. However, the Diminishing Adaptation condition [35] holds, as adaptation will naturally decrease as sampling progresses.

2.4. Deep neural network

The approximate/surrogate model in our experiments is a feed-forward deep neural network (DNN), a type of artificial neural network with multiple hidden layers, as implemented in the open-source neural-network library `Keras` [36] utilising the `Theano` backend [37].

Artificial neural networks have previously been successfully applied as fast model proxies in inverse geophysics problems. Examples include [38], who used a neural network with two hidden layers for Monte Carlo sampling in the context of a crosshole traveltime inversion, and [39] who used a neural network with a single hidden layer and a Differential Evolution Adaptive Metropolis sampler for electromagnetic inversion.

The DNN approximates the forward map, accepting a vector of KL coefficients $\boldsymbol{\theta} \in \mathbb{R}^k$, and returning an approximation of the vector of approximate model output $\hat{\mathcal{F}}(\boldsymbol{\theta}) \in \mathbb{R}^m$ – in this paper a vector of hydraulic heads at given sampling points, i.e. $\hat{\mathcal{F}}(\boldsymbol{\theta}) : \mathbb{R}^k \mapsto \mathbb{R}^m$. Fig. 2 shows the graph of one particular DNN employed in our experiments.

Each edge in Fig. 2 is equipped with a weight $w^l_{i,j}$ where l is index of the layer that the weight feeds into, i is the index of nodes in the same layer and j is the index of nodes in the previous layer. These weights can be arranged in $n \times m$ matrices $\boldsymbol{W}_l$ for each layer l. Similarly, each node is equipped with a bias b^l_i where l is index of its layer and i is the index of node, and these biases can be arranged in vectors $\boldsymbol{b}_l$. Data is propagated through the network such that the output $\boldsymbol{y}_l$ of a layer l with activation function $\mathcal{A}_l(\cdot)$ is

$$\boldsymbol{y}_l = \mathcal{A}_l\left(\boldsymbol{b}_l + \boldsymbol{W}_l\, \boldsymbol{y}_{l-1}\right). \tag{17}$$

Activation functions $A(\cdot)$ are applied element-wise on their input vectors $\boldsymbol{x}$ so that

$$\mathcal{A}(\boldsymbol{x}) = (A(x_1),\ A(x_2)\ \ldots\ A(x_n))^{\intercal}$$

Many different activation functions are available for artificial neural networks, and we here give a short description of the ones employed in our experiments: the *sigmoid* and the *rectified linear unit* ('ReLU'). The transfer function of the nodes in the first layer of each DNN was of the type *sigmoid*:

$$S(x) = \frac{1}{1+e^{-x}} \tag{18}$$

squashing the input vector into the interval (0, 1), effectively resulting in a strictly positive output from the first hidden layer. The remaining hidden layers consisted of nodes with the *de facto* standard hidden layer activation function for deep neural networks, the *rectified linear unit* ('ReLU'):

$$R(x) = \begin{cases} x, & \text{if } x > 0, \\ 0, & \text{otherwise.} \end{cases}$$

To fit an artificial neural network to a given set of data, the network is initially compiled using random weights and biases and then trained using a dataset of known inputs and their corresponding outputs. The weights and biases are updated iteratively during training by way of an appropriate optimisation algorithm and a loss function, and if appropriately set up, will converge towards a set of optimal values, allowing the DNN to predict the response of the forward model to some level of accuracy [40]. Our particular DNNs were trained using the mean squared error (MSE) loss function

$$\text{MSE} = \frac{1}{m}\sum_{i=1}^{m}(h_i - \hat{h}_i)^2$$

for m output variables, and the RMSprop optimiser, a stochastic, gradient based and adaptive algorithm, suggested by [41] and widely used for training DNNs.

3. Adaptive delayed acceptance proposal using a deep neural network

In this section we describe a modified adaptive delayed acceptance proposal for MCMC, using ideas from multi-level MCMC [7]. The general approach generates proposals by running Markov subchains driven by an approximate model. In our case this approximation is constructed from a DNN of the forward map $\mathcal{F}(\boldsymbol{\theta})$ trained from offline samples of the prior distribution. Finally, we show how the approximate map can be corrected online, by adaptively learning a simple multi-variant Gaussian correction to the outputs of the neural network.

3.1. Modified delayed acceptance MCMC

Delayed Acceptance (DA) [15] is a technique that exploits a model hierarchy consisting of an expensive fine model and relatively inexpensive coarse approximation. The idea is simple: a proposal is first evaluated (pre-screened) by an approximate model and immediately discarded if it is rejected. Only if accepted, it is subjected to a second accept/reject step using the fine model. In this context, the likelihood of observations given a parameter set is henceforth denoted $\hat{\mathcal{L}}(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta})$ when evaluated on the approximate model and remains $\mathcal{L}(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta})$ when evaluated on the fine model. This simple screening mechanism cheaply filters out poor proposals, wasting minimal time evaluating unlikely proposals on the expensive, fine model. Crucially, the coarse model need not evaluate every parameter, only a subset. The remaining fine parameters can then be sampled prior to the second accept/reject step. We denote the full parameter set $\boldsymbol{\theta}$, the coarse parameters $\hat{\boldsymbol{\theta}}$ and the fine parameters $\tilde{\boldsymbol{\theta}}$. so that $\boldsymbol{\theta} = [\hat{\boldsymbol{\theta}}, \tilde{\boldsymbol{\theta}}]$.

In this paper we extend this approach by not evaluating *every* accepted approximation proposal with the fine model. Instead, a proposal for the fine model is generated by running an approximate subchain until t approximate proposals have been accepted and only then evaluate using the fine model. We define the required number of accepted proposals in the approximate subchains as the *offset length*. This modified Delayed Acceptance MCMC algorithm is described in Algorithm 2 and an illustration of the process is given in Fig. 3.

This way, the autocorrelation of the fine chain is reduced, since proposals are 'more independent'. This approach is strongly related to a two-level version of multi-level MCMC. Since the fine model likelihood ratio is corrected by the inverse of the approximate likelihood ratio in step 6 of Algorithm 2, detailed balance is satisfied, the resulting Markov Chain is guaranteed to come from the true posterior and there is no loss of accuracy, even if the approximate model is severely wrong [15]. To demonstrate that this approach does indeed decrease the autocorrelation in our fine chain MCMC samples, we compute the Effective Sample Size N_{eff} of each MCMC simulation according to the procedures described in [42].

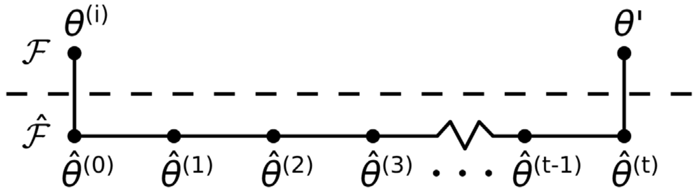


Fig. 3. Illustration of the principle used to offset fine level samples to reduce autocorrelation. The fine model $\mathcal{F}$ is only evaluated using the full set of proposed parameters $\boldsymbol{\theta}'$ after a prescribed number t (the *offset length*) of approximation parameter sets $\{\hat{\boldsymbol{\theta}}^{(1)}, \hat{\boldsymbol{\theta}}^{(2)}, \dots, \hat{\boldsymbol{\theta}}^{(t)}\}$ have been evaluated on the approximate model $\hat{\mathcal{F}}$ and accepted into the coarse chain.

Algorithm 2: Modified Delayed Acceptance MCMC

1. Given a realisation of the approximation parameters $\hat{\boldsymbol{\theta}}^{(j)}$ and the transition kernel $q(\hat{\boldsymbol{\theta}}'|\hat{\boldsymbol{\theta}}^{(j)})$, generate a proposal for the approximation $\hat{\boldsymbol{\theta}}'$.
2. Compute the likelihood ratio on the approximate model between the proposal and the previous realisation:
$$\alpha_1 = \min\left\{1, \frac{\pi_0(\hat{\boldsymbol{\theta}}')\hat{\mathcal{L}}(\boldsymbol{d}_{\text{obs}}|\hat{\boldsymbol{\theta}}')}{\pi_0(\hat{\boldsymbol{\theta}}^{(j)})\hat{\mathcal{L}}(\boldsymbol{d}_{\text{obs}}|\hat{\boldsymbol{\theta}}^{(j)})}\right\} \qquad (AM)$$
$$\alpha_1 = \min\left\{1, \frac{\hat{\mathcal{L}}(\boldsymbol{d}_{\text{obs}}|\hat{\boldsymbol{\theta}}')}{\hat{\mathcal{L}}(\boldsymbol{d}_{\text{obs}}|\hat{\boldsymbol{\theta}}^{(j)})}\right\} \qquad (pCN)$$
3. If $u \sim U(0,1) > \alpha_1$ then set $\hat{\boldsymbol{\theta}}^{(j+1)} = \hat{\boldsymbol{\theta}}^{(j)}$ and return to (1); otherwise set $\hat{\boldsymbol{\theta}}^{(j+1)} = \hat{\boldsymbol{\theta}}'$ and continue to (4).
4. If t proposals have been accepted in the approximation subchain, continue to (5), otherwise return to (1).
5. Given the latest realisation of the entire parameter set $\boldsymbol{\theta}^{(i)} = [\hat{\boldsymbol{\theta}}^{(i)}, \tilde{\boldsymbol{\theta}}^{(i)}]$ with fine parameters $\tilde{\boldsymbol{\theta}}^{(i)}$ and the transition kernel $q(\tilde{\boldsymbol{\theta}}'|\tilde{\boldsymbol{\theta}}^{(i)})$, generate a proposal for the fine parameters $\tilde{\boldsymbol{\theta}}'$ and set $\boldsymbol{\theta}' := [\hat{\boldsymbol{\theta}}', \tilde{\boldsymbol{\theta}}']$.
6. Compute the likelihood ratio on the fine model between the proposal and the previous realisation:
$$\alpha_2 = \min\left\{1, \frac{\pi_0(\boldsymbol{\theta}')\mathcal{L}(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta}')}{\pi_0(\boldsymbol{\theta}^{(i)})\mathcal{L}(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta}^{(i)})} \frac{\pi_0(\hat{\boldsymbol{\theta}}^{(i)})\hat{\mathcal{L}}(\boldsymbol{d}_{\text{obs}}|\hat{\boldsymbol{\theta}}^{(i)})}{\pi_0(\hat{\boldsymbol{\theta}}')\hat{\mathcal{L}}(\boldsymbol{d}_{\text{obs}}|\hat{\boldsymbol{\theta}}')}\right\} \qquad (AM)$$
$$\alpha_2 = \min\left\{1, \frac{\mathcal{L}(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta}')}{\mathcal{L}(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta}^{(i)})} \frac{\hat{\mathcal{L}}(\boldsymbol{d}_{\text{obs}}|\hat{\boldsymbol{\theta}}^{(i)})}{\hat{\mathcal{L}}(\boldsymbol{d}_{\text{obs}}|\hat{\boldsymbol{\theta}}')}\right\} \qquad (pCN)$$
7. If $u \sim U(0,1) > \alpha_2$ then set $\boldsymbol{\theta}^{(i+1)} = \boldsymbol{\theta}^{(i)}$, otherwise set $\boldsymbol{\theta}^{(i+1)} = \boldsymbol{\theta}'$.

3.2. Adaptive correction of the approximate posterior

Whilst in theory the modified delayed acceptance proposal described in Section 3.1 will provide a convergent Metropolis–Hastings algorithm, there are cases in which the rate of convergence will be extremely slow. To demonstrate this, the left-hand contour plot in Fig. 4 shows an artificially bad example. In this case the approximate model (red isolines) poorly captures the target likelihood distribution (blue density); there is a clear offset in the distributions, and the scale, shape and orientation of the approximate likelihood is incorrect. If using the modified delayed acceptance algorithm without alteration, it is easy to see that the proposal mechanism would struggle to traverse the whole of the target distribution, since much of it lies in the tails of the approximate likelihood

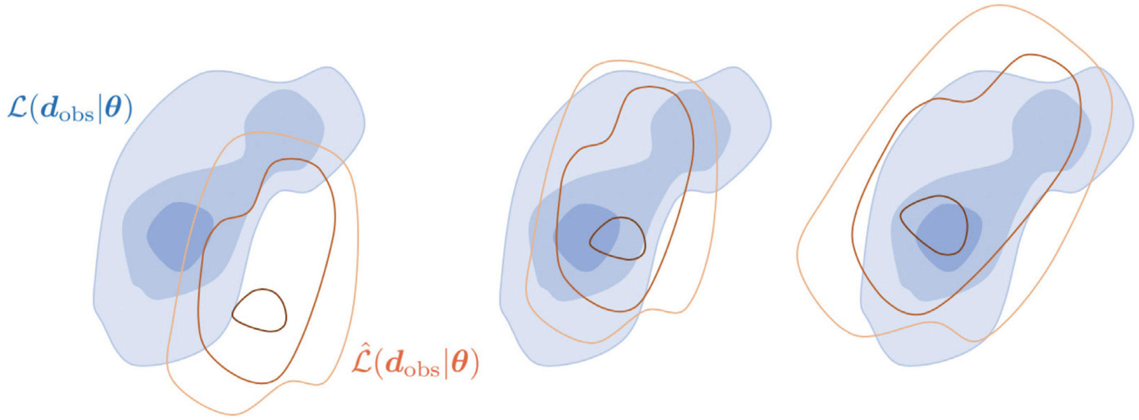

$\mathcal{L}(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta})$
$\hat{\mathcal{L}}(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta})$

Fig. 4. Fine/target likelihood (blue) and approximate likelihood (red). (Left) Original likelihood before correction, (middle) corrected likelihood by a constant shift $\boldsymbol{\mu}_{\text{bias}}$ and (right) corrected approximate likelihood by multivariate Gaussian. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

distribution. As a result, in practice, we would observe extremely slow convergence to the true posterior; in practice – at finite computational times – results would contain a significant bias.

An ad hoc way to overcome this is to apply so-called *tempering* on the statistical model which drives the subchain. In this technique, the variance of the misfit $\boldsymbol{\Sigma}_\epsilon$ on the subchain is artificially inflated to capture the uncertainty in the approximate model. The issue in adopting this approach is the difficulty in selecting a robust inflation factor for tempering, particularly in higher dimensions. Furthermore, an isotropic inflation of the approximate posterior will in general be sub-optimal.

In this paper we instead implement an adaptive enhanced error model (EEM), which overcomes many of these challenges. Moreover, it is easy to implement and has negligible additional computational cost. Let $\hat{\mathcal{F}}$ denote the approximate forward map of the fine/target model $\mathcal{F}$. Then, following [43,18], we apply a trick to the statistical model (8) where we add and subtract the coarse map $\hat{\mathcal{F}}$. With some rearrangement we obtain the expression

$$\boldsymbol{d}_{\text{obs}} = \mathcal{F}(\boldsymbol{\theta}) + \boldsymbol{\epsilon} = \mathcal{F}(\boldsymbol{\theta}) + \hat{\mathcal{F}}(\boldsymbol{\theta}) - \hat{\mathcal{F}}(\boldsymbol{\theta}) + \boldsymbol{\epsilon} = \hat{\mathcal{F}}(\boldsymbol{\theta}) + \underbrace{\left(\mathcal{F}(\boldsymbol{\theta}) - \hat{\mathcal{F}}(\boldsymbol{\theta})\right)}_{:=\mathcal{B}(\boldsymbol{\theta})} + \boldsymbol{\epsilon}. \tag{19}$$

Here $\mathcal{B}(\boldsymbol{\theta}) = \mathcal{F}(\boldsymbol{\theta}) - \hat{\mathcal{F}}(\boldsymbol{\theta})$ is the bias associated with the approximation at given parameter values $\boldsymbol{\theta}$. We approximate this bias using a multivariate Gaussian distribution, i.e. $\mathcal{B} \sim \mathcal{N}(\boldsymbol{\mu}_{\text{bias}}, \boldsymbol{\Sigma}_{\text{bias}})$, and therefore the likelihood function (12) can be rewritten as

$$\hat{\mathcal{L}}(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta}) = \exp\left(-\frac{1}{2}(\hat{\mathcal{F}}(\boldsymbol{\theta}) + \boldsymbol{\mu}_{\text{bias}} - \boldsymbol{d}_{\text{obs}})^\intercal(\boldsymbol{\Sigma}_{\text{bias}} + \boldsymbol{\Sigma}_e)^{-1}(\hat{\mathcal{F}}(\boldsymbol{\theta}) + \boldsymbol{\mu}_{\text{bias}} - \boldsymbol{d}_{\text{obs}})\right). \tag{20}$$

The influence of redefining the likelihood is best demonstrated geometrically, as shown in Fig. 4 (middle and right). Firstly, as shown in Fig. 4 (middle) we can make a better approximation by simply adding a shift of the mean bias $\boldsymbol{\mu}_{\text{bias}}$ to the original approximate model $\hat{\mathcal{F}}(\boldsymbol{\theta})$. This has the effect of aligning the 'centre of mass' of each of the distributions. Secondly, we can learn the covariance structure of the bias. This has the effect of stretching and rotating the approximate distribution to give an even better overall approximation, as shown in Fig. 4 (right). The final mismatch between the approximate and target distribution will be driven by the assumption that bias can be represented by a multivariate Gaussian, although more complex distributions could be constructed using, for example, Gaussian process regression. Whilst this is an avenue to explore in the future, any such approach would surrender the simplicity of this approach, which from the results appears particularly effective.

The idea of using an EEM when dealing with model hierarchies originates from [43], who suggested to use samples from the prior distribution of parameters to construct the EEM prior to Bayesian inversion, so that

$$\boldsymbol{\mu}_{\text{bias}} = \frac{1}{N}\sum_{i=1}^{N}\mathcal{B}(\boldsymbol{\theta}^{(i)}) \quad \text{and} \quad \boldsymbol{\Sigma}_{\text{bias}} = \frac{1}{N-1}\sum_{i=1}^{N}(\mathcal{B}(\boldsymbol{\theta}^{(i)}) - \boldsymbol{\mu}_{\text{bias}})(\mathcal{B}(\boldsymbol{\theta}^{(i)}) - \boldsymbol{\mu}_{\text{bias}})^\intercal \tag{21}$$

The estimates for $\boldsymbol{\mu}_{\text{bias}}$ and $\boldsymbol{\Sigma}_{\text{bias}}$ could be obtained by sampling the prior distribution and comparing the approximate forward map against the target forward map. This approach has previously been successfully applied to a geophysical inverse problem by [44], who compared the modelling error for a large number of crosshole tomography models. However, since the model output generated by parameter sets drawn from the prior distribution may be biased significantly differently than samples drawn from a (relatively concentrated) posterior distribution, this approach may lead to an EEM that poorly represents the model bias associated with the posterior. If the approximate model is a good approximation *on average*, constructing the EEM from the prior distribution would lead to an underestimation of the mean and an overestimation of the covariance of the bias, compared to an EEM constructed from the posterior. Furthermore, in our example where the approximate model is built from samples from the prior, it is expected that such an approach would further underestimate both the mean *and* covariance of the bias, since the neural network has been explicitly trained to minimise the error with respect to samples from the prior.

Instead of estimating the bias using the prior, the posterior bias can be constructed on-line by iteratively updating its mean $\boldsymbol{\mu}_{\text{bias}}$ and covariance $\boldsymbol{\Sigma}_{\text{bias}}$ using coarse/fine solution pairs from the MCMC samples as suggested by [45]. Another similar approach was employed to a Bayesian geophysical problem by [46], who collected model bias estimates while sampling, and used the bias estimates of the k-nearest-neighbours of each new coarse sample to construct a bias. In this case we select

$$\boldsymbol{\mu}_{\text{bias},i+1} = \frac{1}{i+1}\left(i\,\boldsymbol{\mu}_{\text{bias},i} + \mathcal{B}(\boldsymbol{\theta}^{(i+1)})\right) \quad \text{and} \tag{22}$$

$$\boldsymbol{\Sigma}_{\text{bias},i+1} = \frac{i-1}{i}\boldsymbol{\Sigma}_{\text{bias},i} + \frac{1}{i}(\mathcal{B}(\boldsymbol{\theta}^{(i+1)})\,\mathcal{B}(\boldsymbol{\theta}^{(i+1)})^{\mathsf{T}} - \boldsymbol{\mu}_{\text{bias},i+1}\,\boldsymbol{\mu}^{\mathsf{T}}_{\text{bias},i+1}) \tag{23}$$

While this approach does not in theory guarantee ergodicity of the chain (as is also the case with the Adaptive Metropolis proposal), the bias distribution will converge as the chain progresses and adaptation diminishes, resulting in a *de facto* ergodic process after an initial period of high adaptivity. This is a common feature of adaptive MCMC algorithms, as discussed in the classic paper on Adaptive Metropolis [17]. Our experiments showed that the bias distribution did indeed converge for every simulation, and that repeated experiments converged towards the same posterior bias distribution. Admitting a bias term in the inverse problem further introduces an issue of *identifiability*, as highlighted in [47]. Since observations are now modelled as a sum of coarse model output and multiple stochastic terms, the stochastic terms $\mathcal{B} \sim N(\boldsymbol{\mu}_{\text{bias}}, \boldsymbol{\Sigma}_{\text{bias}})$ and $\boldsymbol{\epsilon} \sim N(0, \sigma^2\mathbb{I}_n)$ are generally unidentifiable in the coarse model formulation, meaning that the bias $\mathcal{B}$ and the data modelling noise $\boldsymbol{\epsilon}$ are observationally equivalent, and not well-defined.

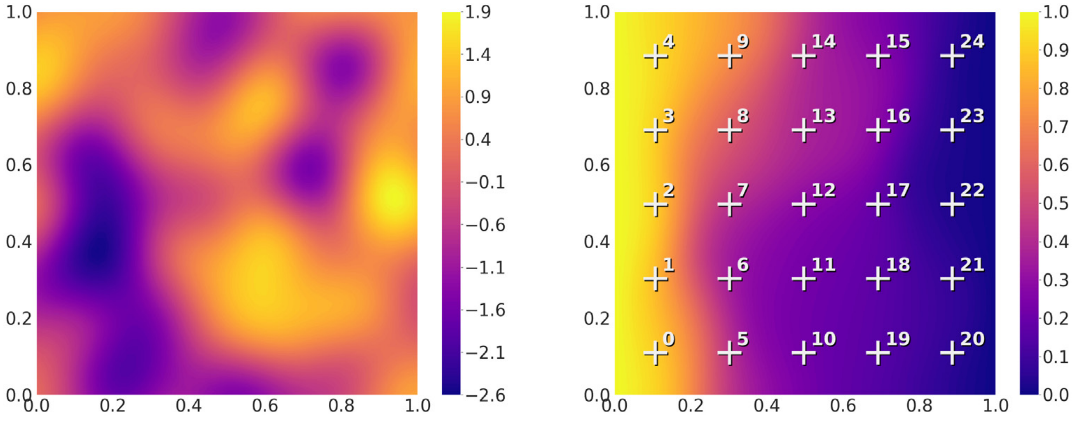
4. Results

In this section, we examine the effectiveness of our proposed strategy on two synthetic groundwater flow problems: a two-dimensional problem with an isotropic covariance kernel and a three-dimensional problem with an anisotropic covariance kernel. For both examples, we begin by outlining the model setup, for which we select a ‘true’ transmissivity field and a number of fixed observation points. For the first example, the influence of training size for the DNNs is examined, and the total cost of uncertainty quantification using a selection of DNNs is computed. For the second example we use a single DNN setup and analyse the resulting posterior marginal distributions and the quantity of interest. The first example was completed on commodity hardware — an HP Elitebook 840 G5 with an Intel Xeon E3-1200 quad-core processor, while the second example was completed on a TYAN Thunder FT48T-B7105 GPU server with two Intel Xeon Gold 6252 processors and an NVIDIA RTX 2080Ti GPU.

4.1. Example 1: 2D unit square

4.1.1. Model setup

This example was conducted on a unit square domain $\Omega = [0, 1]^2$, meshed using an unstructured triangular grid comprising 2,601 degrees of freedom. Dirichlet boundary conditions were imposed on the left and right boundaries with hydraulic heads of 1 and 0, respectively. The top and bottom edges impose homogeneous no-flow Neumann boundary conditions. To avoid committing an inverse crime, the covariance length scales of the ARD squared exponential kernel was set to $\boldsymbol{l} = (0.11, 0.11)^{\mathsf{T}}$ for data generation and $\boldsymbol{l} = (0.1, 0.1)^{\mathsf{T}}$ for the forward model used in sampling. The chosen length scales effectively resulted in an isotropic covariance kernel, equal to the ‘classic’



(a) Log-Transmissivity. (b) Hydraulic head.

Fig. 5. “True” transmissivity field, its corresponding solution and sampling points.

Table 1
Neural network layers and activation functions in the model approximation DNNs.

Layer	# Nodes	Activation functions			
		DNN1	DNN2	DNN3	DNN4
Input	k KL coefficients	–	–	–	–
1	$4k$	Sigmoid	Sigmoid	Sigmoid	Sigmoid
2	$8k$	ReLU	ReLU	–	–
3	$4k$	ReLU	ReLU	ReLU	ReLU
Output	m datapoints	Exponential	Linear	Exponential	Linear

square exponential kernel with $l = 0.1$. This resulted in a KL decomposition with $> 80\%$ of total signal energy in the 32 largest eigenvalues and $> 95\%$ of signal energy in the 64 largest eigenvalues. Hence, 32 modes were included in the approximate model whilst 64 modes where included in the fine model.

Fig. 5(a) shows the ‘true’ transmissivity field that we attempt to recover through our MCMC methodology and the modelled, corresponding hydraulic head. Synthetic samples for the likelihood function were extracted at 25 points on a regular grid with a horizontal and vertical spacing of 0.2 m (Fig. 5(b)), and these data were perturbed with white noise with covariance $\boldsymbol{\Sigma}_e = 0.001\, \mathbb{I}_m$.

4.1.2. Deep neural network design, training and evaluation

We evaluated a selection of different DNNs to investigate the impact of various network depths and activation functions on the DNN performance. Table 1 shows the layers of the employed DNNs, the number of nodes in each layer and their corresponding activation functions. DNN1 and DNN2 had three hidden layers, while DNN3 and DNN4 had only two, as the ReLU layer with $8k$ nodes was not included in these networks. The output layer of DNN1 and DNN3 consisted of nodes with an exponential activation function $E(x) = e^x$, resulting in a strictly positive output. The DNNs with an exponential activation function in the final layer tended overall to lead to the best performance.

Each DNN was trained on a set of samples from the prior distribution of parameters $\pi_0(\boldsymbol{\theta}) = \mathcal{N}(0, \mathbb{I}_k)$, in advance of running the MCMC. Hence, the DNN samples were drawn from a Latin Hypercube [48] in the interval [0, 1] and transformed to the standard normal distribution using the *probit*-function, such that $\boldsymbol{\theta}_{train} \sim \mathcal{N}(0, \mathbb{I}_k)$. The coarse, 32-mode FEM model was then run for every parameter sample, obtaining for each a vector of model outputs at sampling points given parameters. We trained and tested each DNN on a range of different sample sizes, namely $N_{\text{DNN}} = \{2000, 4000, 8000, 16000, 32000, 64000\}$, where $N_{\text{DNN}} = N_{train} + N_{test}$, with a 9:1 training/test splitting ratio. Each DNN was then trained for 200 epochs with a batch size of 50 using the `rmsprop` optimiser [41].

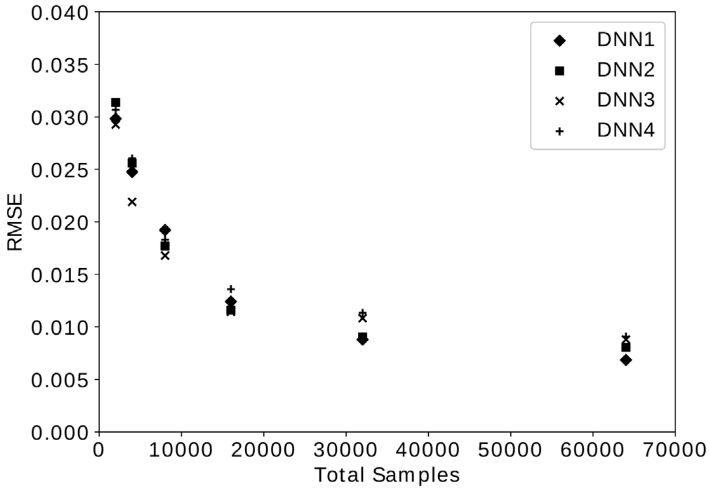


Fig. 6. Testing performance (RMSE) of each DNN against the total sample size ($N_{\mathrm{DNN}} = N_{train} + N_{test}$). Please refer to Table 1 for details in the structure of each DNN.

Deep Neural Networks performance was compared using the RMSE of their respective testing dataset

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(h_i - \hat{h}_i)^2} \tag{24}$$

The residual RMSE (24) of each DNN was computed to compare the network designs described in Table 1 and to investigate the influence of training dataset size on the DNN performance (Fig. 6). As expected, each DNN performed better as the number of samples in the training dataset were increased. In comparison, the DNN design had much less influence on the testing performance, suggesting that the main driver for constructing an accurate surrogate model, within the bounds of the examined DNN designs, was the number of training samples. For the remaining analysis, we chose the network design resulting in the overall lowest RMSE at $N_{\mathrm{DNN}} = 64000$, namely DNN1, and the sample sizes $N_{\mathrm{DNN}} = \{4000, 16000, 64000\}$.

Further performance analysis consisted of analysing the DNN error $e = h_{\mathrm{true}} - h_{\mathrm{pred}}$ for true and predicted heads (h_{true} and h_{pred}, respectively) for datapoints 0, 8, 16 and 24. (Fig. 7). All error distributions were approximately Gaussian, with the errors for the DNN with $N_{\mathrm{DNN}} = 4000$ exhibiting some right skew at sampling point 24. For all DNNs, the sampling points closer to the boundaries (at sampling points 0 and 24) had lower errors than those further away, since the heads close to the boundaries were more constrained by the model.

4.1.3. Uncertainty quantification

For inversion and uncertainty quantification, we chose a multivariate standard normal distribution as the prior parameter distribution, $\pi_0(\boldsymbol{\theta}) = \mathcal{N}(0, \mathbb{I}_k)$ and set the error covariance to $\boldsymbol{\Sigma}_e = 0.001\ \mathbb{I}_m$. While computationally convenient, the zero-centred prior in practice favours transmissivity field realisations capable of reproducing the observed heads with as little variation as possible. In total, eight different sampling strategies were investigated, namely single level ‘Vanilla’ MCMC, with no delayed acceptance, no adaptivity, and using only the 64-mode fine model; DA using three different DNNs trained and tested on $N_{\mathrm{DNN}} = \{4000, 16000, 64000\}$ samples as the coarse model and the 64-mode model as the fine; and DA with an enhanced error model (DA/EEM) using the same three DNNs. The offset length t for the DA strategies was manually tuned to achieve an acceptance rate of $a \in [0.2, 0.4]$. To investigate the effect of the offset length t independently of other factors, an additional simulation with $N_{\mathrm{DNN}} = 64000$ and $t = 1$ was also completed. In this first example, every simulation was completed using the pCN transition kernel, with $\beta = 0.15$. Each MCMC sampling strategy was repeated ($n = 32$) using randomly generated random seeds, to ensure that every starting point would converge towards the same stationary distribution and to allow for cross-chain statistics to be computed. Results given in this section pertain to these multi-chain samples rather than individual MCMC realisations, unless otherwise stated.

Our sampling strategies recovered the ground truth with good accuracy. Fig. 8 shows the mean and variance of the recovered field from the DA/EEM MCMC using the DNN with $N_{\mathrm{DNN}} = 64000$. All recovered fields exhibit

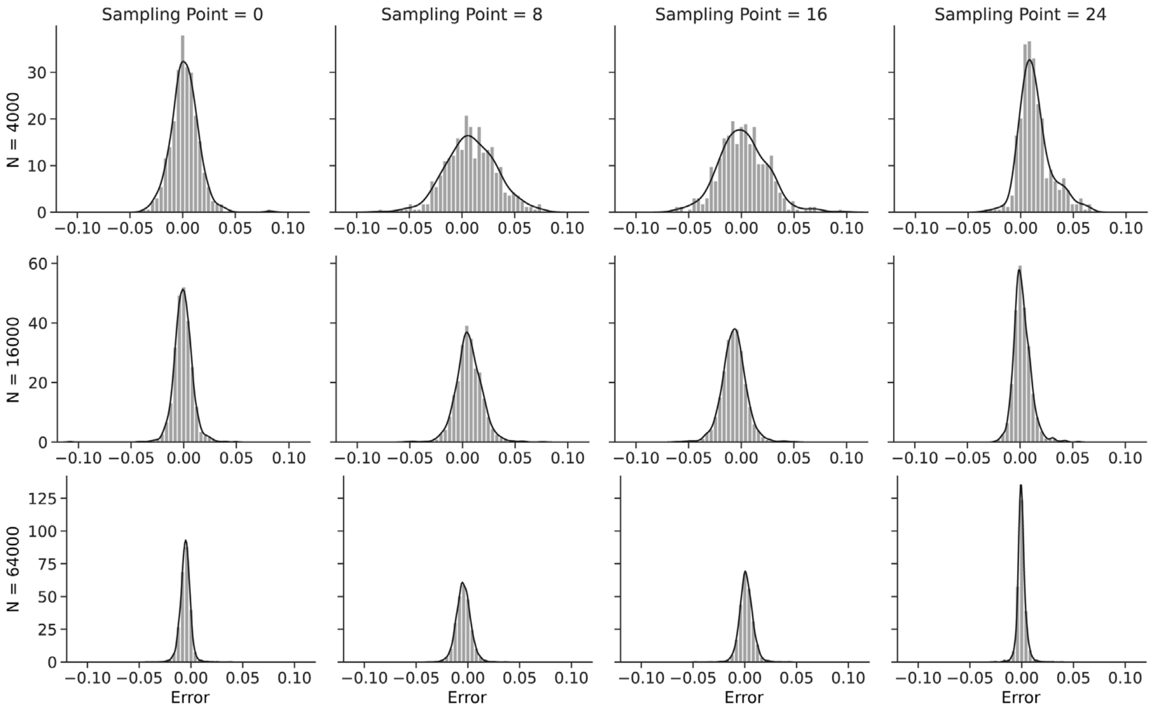


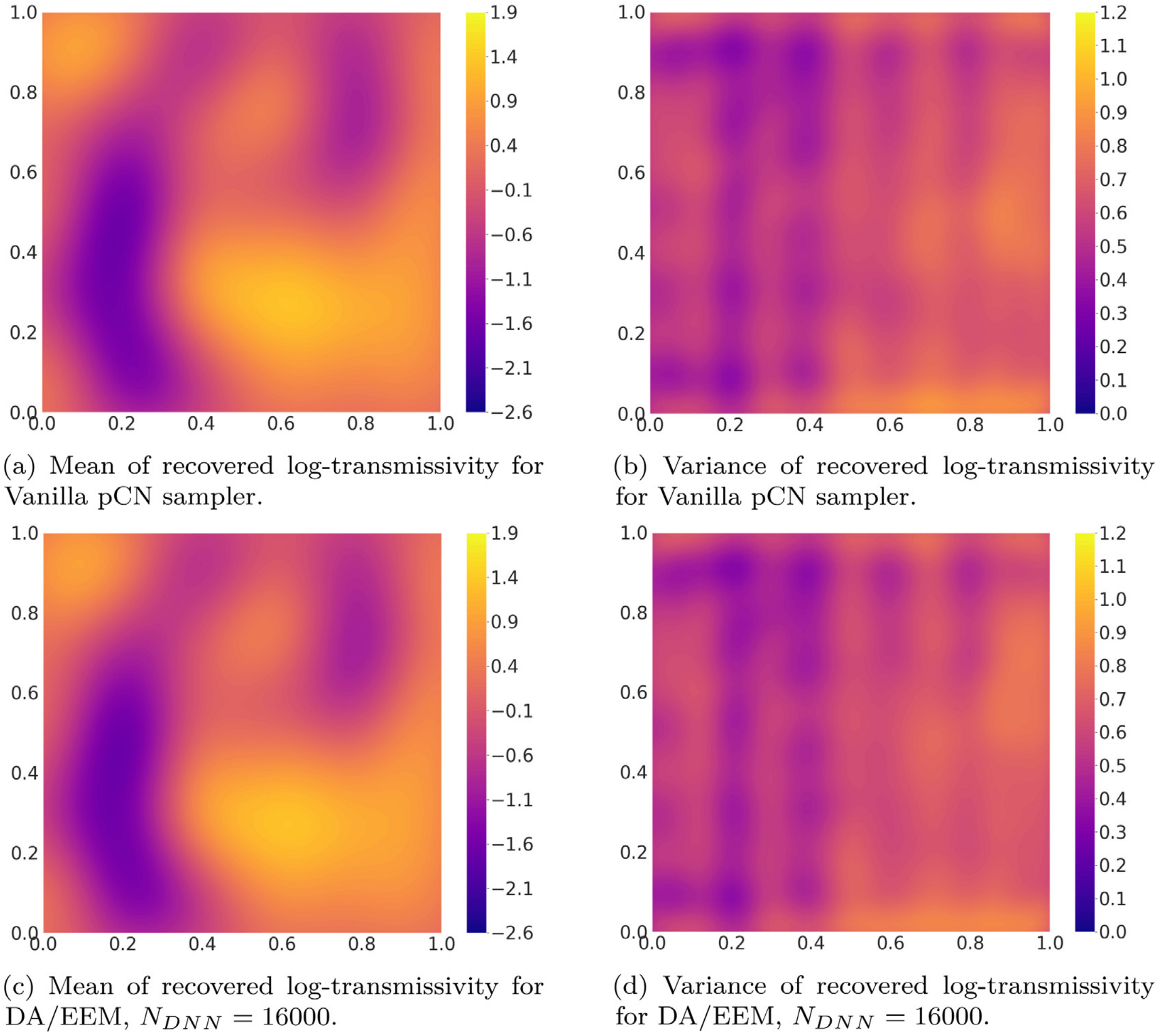
Fig. 7. Density plot of the error ($e = h_{\text{true}} - h_{\text{pred}}$) of the testing dataset for DNN1 trained and tested on $N_{\text{DNN}} = \{4000, 16000, 64000\}$ samples, for sampling points 0, 8, 16 and 24. Bars show density of each bin, while the curve shows Gaussian kernel density estimate.

Table 2
Results for various MCMC sampling strategies, means of multiple chains with $n = 32$. N_{DNN} is the number of total samples used to construct the DNN. t is the improved DA offset length. N_C/N_F is the final length of the coarse and fine chain, respectively, after subtracting burnin. *Acc. rate* is the fine chain acceptance rate. *Time* (min) is the total running time of the simulation in minutes. N_{eff} is the Effective Sample Size.

Strategy	N_{DNN}	t	N_C/N_F	Acc. Rate	Time (min)	N_{eff}
Vanilla	–	–	—/40000	0.33	32.1	85.6
DA	4000	2	85461.9/20000	0.27	16.2	64.5
DA/EEM	4000	2	78853.4/20000	0.31	15.2	79.0
DA	16000	4	172383.1/20000	0.27	18.2	116.3
DA/EEM	16000	4	178978.4/20000	0.30	18.4	143.6
DA	64000	8	336447.5/20000	0.24	30.1	196.5
DA/EEM	64000	8	377524.4/20000	0.30	29.9	235.7
DA/EEM	64000	1	56824.3/20000	0.57	15.3	68.6

higher smoothness than the ground truth, which can be attributed to the relatively low number of sampling points and their regular distribution on the domain, in combination with the regularisation introduced by the prior. Since the KL decomposition incorporated > 95% of the signal energy, the truncation would have contributed only marginally to the smoothing. None of the chains recovered the local peak in transmissivity on the right side of the domain, since there was too little data to discover this particular feature. However, this peak is clearly encapsulated by the posterior variance, as shown in Figs. 8(b) and 8(d).

While the recovered fields indicate that every MCMC sampling strategy converged towards the desired stationary distribution, they do not reveal the relative efficiency of each strategy. Hence, the Effective Sample Size (N_{eff}) was computed for each MCMC realisation. Every DA sampling strategy produced higher N_{eff} than the Vanilla pCN sampler, relative to the simulation time, with a clear correlation between DNN testing performance and N_{eff}. This was mainly because the better performing DNNs allowed for a longer coarse chain offset without diverging. Moreover, utilising the EEM produced even higher N_{eff} for every DA chain (Table 2).



(a) Mean of recovered log-transmissivity for Vanilla pCN sampler.

(b) Variance of recovered log-transmissivity for Vanilla pCN sampler.

(c) Mean of recovered log-transmissivity for DA/EEM, $N_{DNN} = 16000$.

(d) Variance of recovered log-transmissivity for DA/EEM, $N_{DNN} = 16000$.

Fig. 8. Mean and variance ($n = 32$) of recovered log-transmissivity fields using Vanilla pCN sampler (top) and DA/EEM MCMC with $N_{\text{DNN}} = 16000$ (bottom). Corresponding plots of every sampling strategy are shown in Figs. B.15–B.21 in Appendix B.

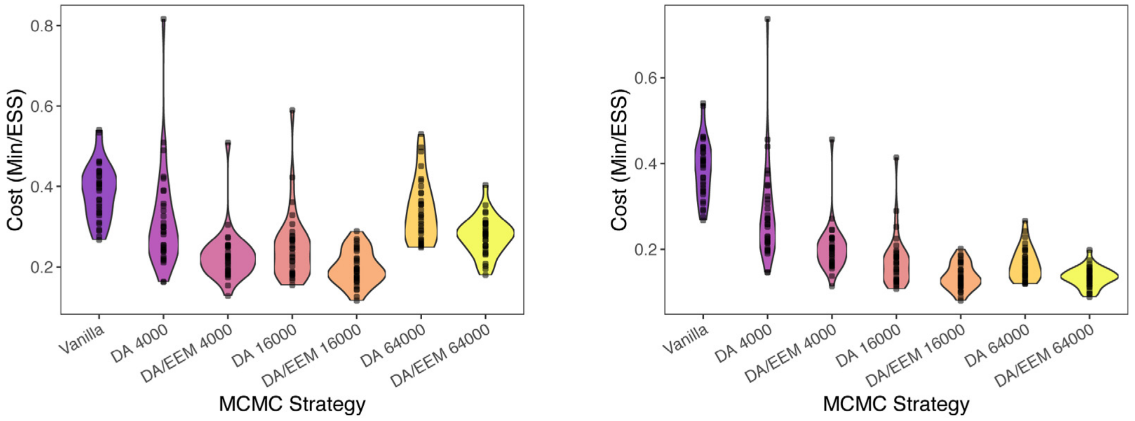
4.1.4. Total cost

Since the DA chains required computation of a significant number of fine model solutions and training of a DNN in advance of running the chain, the total cost C_{total} of each strategy was computed as

$$\text{C}_{\text{total}} = \frac{t_{\text{fine}} + t_{\text{train}} + t_{\text{run}}}{N_{eff}} \tag{25}$$

where t_{fine} was the time spent on precomputing fine model solution, t_{train} was the time spent on training the respective DNN, t_{run} was the time taken to run the chain and N_{eff} was the resulting effective sample size (Fig. 9).

The mean cost of every DA chain was lower than that of the Vanilla pCN chain, with the chains using the EEM consistently cheaper than their non-EEM counterparts. Moreover, using the EEM reduced the variance of the cost in repeated experiments, allowing each repetition to produce a consistently high N_{eff}. The overall cheapest inversion was completed using the DNN trained on 16,000 samples using the EEM, reducing the total cost, relative to the Vanilla pCN MCMC, with 50%. Notice that these results are extremely conservative in the sense that the entire cost of evaluating every DNN training sample and training the DNN in serial on a CPU was factored into the cost of every repetition, even though the same DNN was used for all the repetitions within each sampling strategy. The precomputation cost can be dramatically reduced by evaluating the DNN samples in parallel and utilising high-performance hardware, such as GPUs, for training the DNN.



(a) Total cost (conservative) with the full cost of constructing the DNN factored into all independent DA chains.

(b) Total cost (normalised) with the cost of constructing the DNN distributed between independent DA chains.

Fig. 9. Violinplots showing the total cost C_{total} of each MCMC strategy with $n = 32$. Points show independent Markov Chains.

4.2. Example 2: 3D rectangular cuboid

4.2.1. Model setup

This example was conducted on a rectangular cuboid domain $\Omega = [0, 2] \times [0, 1] \times [0, 0.5]$ meshed using an unstructured tetrahedral grid with 10,416 degrees of freedom (Fig. 10). Dirichlet boundary conditions of $h = 1$ and $h = 0$ were imposed at $x_1 = 0$ and $x_1 = 2$, respectively. No-flow Neumann conditions were imposed on all remaining boundaries.

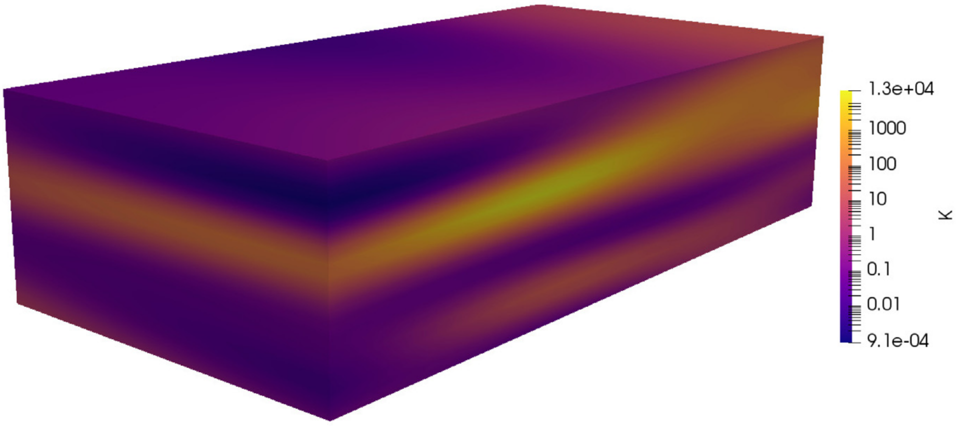
The covariance lengths scales for ARD squared exponential covariance kernel were set to $\boldsymbol{l} = (0.55, 0.95, 0.06)^{\intercal}$ for data generation and $\boldsymbol{l} = (0.5, 1.0, 0.05)^{\intercal}$ for the forward model used in sampling, resulting in a highly anisotropic random process with high variation in the x_3 direction to simulate geological stratification, some variation in the x_1 direction and little variation in the x_2 direction (Fig. 10(a)). Like in the first model, the random process was truncated at 64 KL eigenmodes for the fine model and 32 KL eigenmodes for the coarse model, embodying $> 97\%$ and $> 90\%$ of the total signal energy, respectively.

We drew $w = 50$ sampling well locations randomly using the Maximin Latin Hypercube Design [49], and samples of hydraulic head were extracted at each well at datums $x_3 = \{0.05, 0.15, 0.25, 0.35, 0.45\}$, measured from the bottom of the domain, resulting in $m = 250$ datapoints in total (Fig. 10(b)). These data were perturbated with white noise with covariance $\boldsymbol{\Sigma}_e = 0.001\, \mathbb{I}_m$.

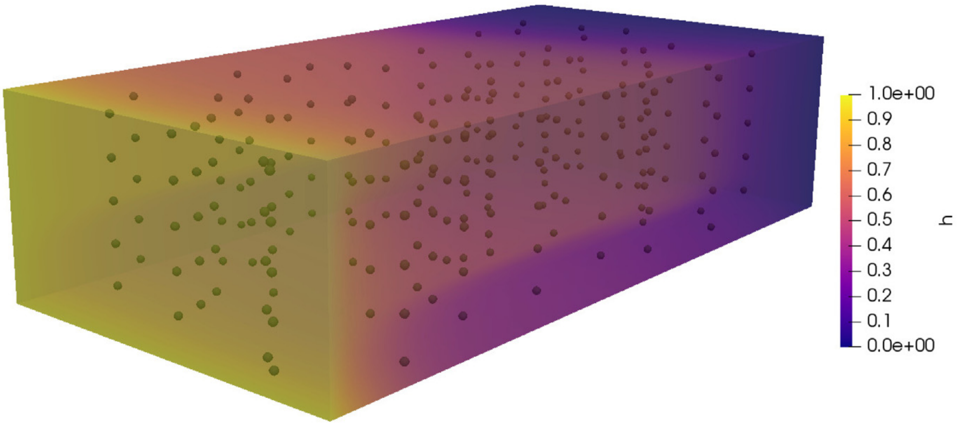
For this example, we first converged the conductivity parameters to the Maximum a posteriori (MAP) estimate $\boldsymbol{\theta}_{MAP} = \arg\max_{\theta} \pi_0(\boldsymbol{\theta})\mathcal{L}(\boldsymbol{d}_{\text{obs}}|\boldsymbol{\theta})$ using gradient descent, since initial MCMC experiments struggled to converge to the posterior distribution for random initial parameter sets.

4.2.2. Deep neural network design, training and evaluation

Training a DNN to accurately emulate the model response for this setup was challenging, and we found no single combination of neural network layers and activation functions that would predict the head at every datapoint with sufficient accuracy. We hypothesise that this limitation could be caused by a strong ill-posedness of the DNN — for a single neural network, the output dimension greatly exceeded the input dimension, i.e. $m \gg k$ where $m = 250$ was the number of datapoints, and $k = 32$ was the coarse model KL modes. When we instead predicted the heads at each datapoint datum using a separate DNN, we found that we could utilise largely the same DNN design as had been employed in the first example. Hence, to predict the head at all datapoints, we utilised five identically designed but independent DNNs (Fig. 11), each with four hidden layers and activation functions as indicated in Table 3. Each DNN was trained and tested on a dataset of $N_{DNN} = 16000$ samples with KL coefficients drawn from a Latin Hypercube [48] in the interval [0, 1] and transformed to a normal distribution centred on the MAP estimate of the parameters $\boldsymbol{\theta}_{MAP}$, i.e. $\boldsymbol{\theta}_{train} \sim \mathcal{N}(\boldsymbol{\theta}_{MAP}, \mathbb{I}_k)$. This was done to increase the density of samples and



(a) Log-Conductivity of ground truth.



(b) Hydraulic head of ground truth and location of sampling points.

Fig. 10. "True" conductivity field, its corresponding solution and sampling points.

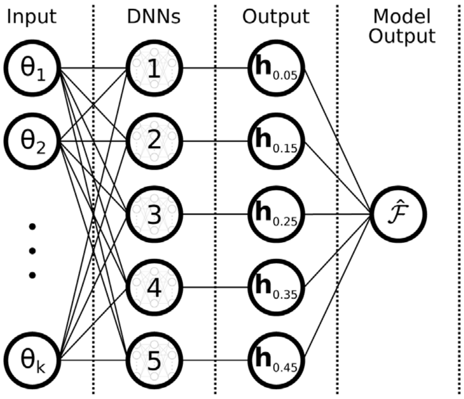


Fig. 11. Layout of the multi-DNN design. Each DNN outputs a vector $\boldsymbol{h}_{x_3}$ vector of w head predictions at datum x_3.

thus improve the DNN prediction at and around the MAP point, which ideally equals the mode of the posterior distribution. The DNNs were then trained for 200 epochs using a batch size of 50, the MSE loss function and the `rmsprop` optimiser [41]. Fig. 12 shows performance plots of each DNN for both the training (top) and the testing (bottom) datasets. While every DNN is clearly moderately biased by the training data, they all performed adequately with respect to the testing data.

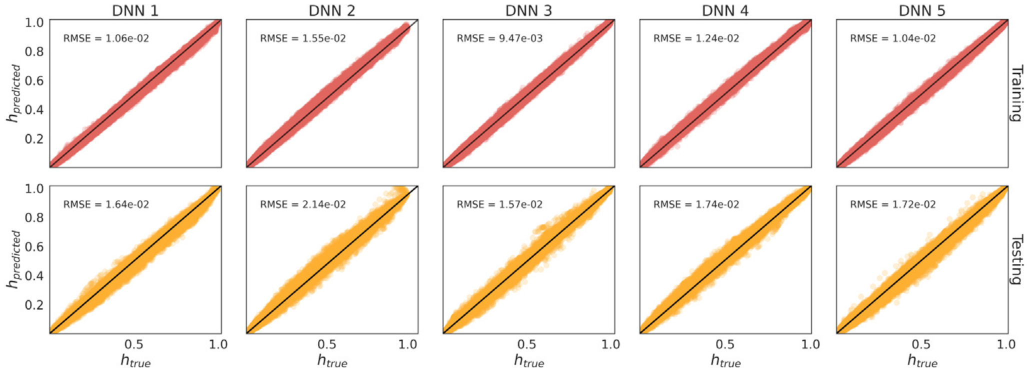


Fig. 12. Performance of the five DNNs used in the multi-DNN approach, as shown in Fig. 11, with respect to the training dataset (top) and the testing dataset (bottom).

Table 3
Layers and activation functions in the four DNNs. Each DNN takes all k KL coefficients as input and predicts the head $\boldsymbol{h}_{x_3}$ at w wells for a given datum.

Layer	# Nodes	Activation functions
Input	k KL coefficients	–
1	$4k$	Sigmoid
2	$8k$	ReLU
3	$8k$	ReLU
3	$4k$	ReLU
Output	w wells	Exponential

4.2.3. *Uncertainty quantification*

Similarly to the first example, we chose a multivariate standard normal distribution $\pi_0(\boldsymbol{\theta}) = \mathcal{N}(0, \mathbb{I}_k)$ as the prior distribution of parameters, and set the error covariance to $\boldsymbol{\Sigma}_e = 0.001\ \mathbb{I}_m$. Hence, the synthetic head data from the wells were perturbated with white noise with covariance $\boldsymbol{\Sigma}_e$. In this example, we utilised the Adaptive Metropolis (AM) transition kernel for generating proposals. We completed $n = 8$ independent simulations, each initialised from a random initial point close to the MAP point $\boldsymbol{\theta}_{MAP}$, with a burnin of 1000 and a final sample size of $N = 10{,}000$. The subchains were run with an acceptance delay of $t = 2$, since longer subchains tended to diverge, leading to sub-optimal acceptance rates on the fine level. The simulations had a mean acceptance rate of 0.26, a mean effective sample size (N_{eff}) of 55.2 and a mean autocorrelation length $\tau = N/N_{eff}$ of 181.0. The samples of each independent simulation were pruned according to the respective autocorrelation length, and the remaining samples were pooled together to yield 443 statistically independent samples that were then analysed further.

Fig. 13 shows the marginal distributions of the six coarsest KL coefficients along with a scatterplot matrix of all the samples remaining after pruning. All the marginal distributions are approximately Gaussian, and the two-parameter marginal distributions are mostly elliptical. It is evident that some of these parameters are correlated, namely parameters $(\boldsymbol{\theta}_0, \boldsymbol{\theta}_5)$, $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2)$, $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_3)$, $(\boldsymbol{\theta}_1, \boldsymbol{\theta}_4)$ and $(\boldsymbol{\theta}_2, \boldsymbol{\theta}_4)$. It is worth mentioning that in every independent simulation, the AM proposal kernel managed to capture these correlations.

Moreover, we analysed the hydraulic head as a function of datum $h(x_3)$ along a line in the centre of the domain $\boldsymbol{x} = (1.0, 0.5, x_3)^{\intercal}$. Fig. 14 shows $h(x_3)$ of the ground truth, MAP point $\boldsymbol{\theta}_{MAP}$, the mean of the $n = 8$ independent simulations, and all the samples remaining after pruning. We observe that both the MAP point and the sample mean are fairly close to the ground truth, albeit exhibiting higher smoothness, particularly between the observation depths, where the head is essentially allowed to vary freely. It is also clear that the individual samples encapsulate the ground truth, indicating that the ground truth is indeed contained by posterior distribution.

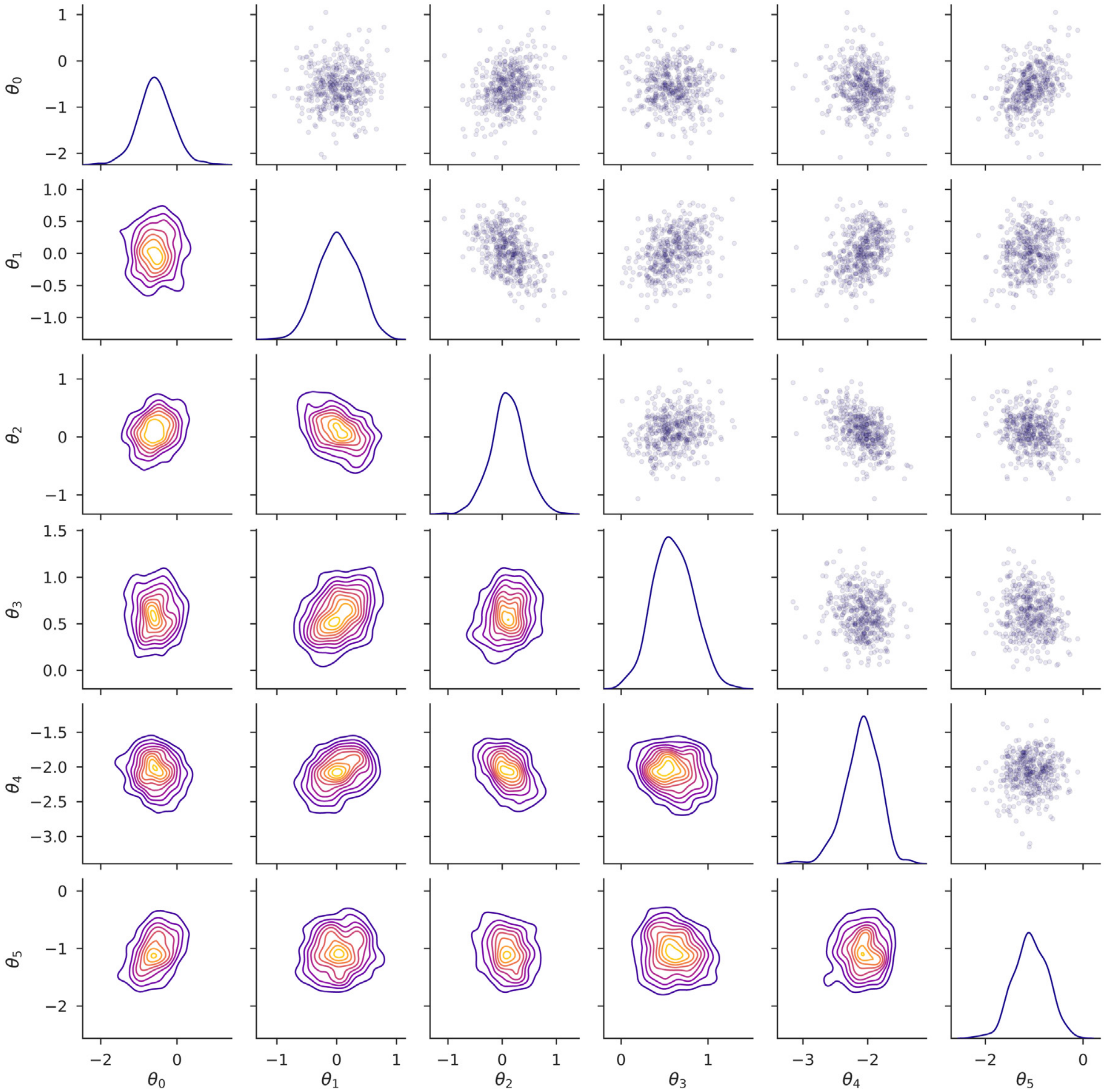


Fig. 13. One and two-dimensional posterior marginal distributions (diagonal and lower triangle) and scatterplots (upper triangle) of posterior samples pruned according to the autocorrelation length of each chain for the largest 5 KL eigenmodes. Please note that the axis scales of are not equal.

5. Discussion

In this paper, we have demonstrated the use of a novel Markov Chain Monte Carlo methodology which employs a delayed acceptance (DA) model hierarchy with a deep neural network (DNN) as an approximate model and a FEM solver as a fine model, and generates proposals using the pCN and AM transition kernels. Results from the first example clearly indicate that the use of a carefully designed DNN as a model approximation can significantly reduce the cost of uncertainty quantification, even for DNNs trained on relatively small sample sizes. We have established that offsetting fine model evaluations in the DA algorithm reduces the autocorrelation of the fine chain, resulting in a higher effective sample size which, in turn, improves the statistical validity of the results. In this context, the performance of the DNN is a critical driver when determining a feasible offset length to avoid divergence of the coarse chain. Hence, if a high effective sample size is required, it may be desirable to invest in

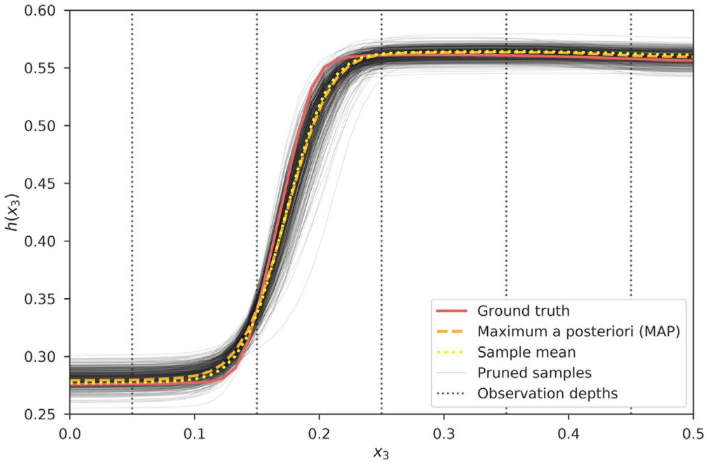


Fig. 14. Hydraulic head as a function of datum $h(x_3)$ at $\boldsymbol{x} = (1.0, 0.5, x_3)^\intercal$. The solid red line shows the hydraulic head of the ground truth, the dashed orange line shows the head of the Maximum a posteriori (MAP) point $\boldsymbol{\theta}_{MAP}$, the dotted yellow line shows the mean head of the independent simulations ($n = 8$) and the thin black lines show the head of 538 statistically independent samples, remaining after pruning according to the autocorrelation length of each chain, $n = 443$. The vertical dotted lines show the observation depths.

a well-performing DNN. Moreover, we have shown that an enhanced error model, which introduces an iteratively-constructed bias distribution in the coarse chain likelihood function, further increases the effective sample size and decreases the variance of the cost in repeated experiments. Finally, we observed that for the second example, even when employing a relatively well-performing model approximation, we had to constrain the offset length of the subchains rather strongly to achieve optimal acceptance rates. This can be attributed in part to an apparent non-spherical and correlated posterior distribution, causing the employed proposal kernels to struggle to discover areas of high posterior probability.

We have demonstrated that relatively simple inverse hydrogeological problems can be solved in reasonable time on a commonly available personal computer with no GPU-acceleration. This opens the opportunity to apply robust uncertainty quantification during fieldwork and as a decision-support tool for groundwater surveying campaigns. We have also demonstrated the applicability of our approach on a larger scale three-dimensional problem, utilising a GPU-accelerated high-performance computer (HPC). Aside from the benefit of using a HPC computer for accelerating the fine model evaluations, utilising the GPU allowed for rapidly training and testing multiple different DNN designs to efficiently establish a well performing model approximation. There are other obvious ways to further increase the efficiency of the proposed methodology. For example, construction of the DNNs used as coarse models comes with the cost of evaluating multiple models from the prior distribution, and, unlike the MCMC sampler, the prior models are independent and these fine model evaluations can thus be massively parallelised.

Our methodology was demonstrated in the context of two relatively simple groundwater flow problems with log-Gaussian transmissivity fields parametrised by Karhunen–Loève decompositions. While this model provides a convenient computational structure for our purposes, it may not reflect the full scale transmissivity of real-world aquifers, particularly in the presence of geological faults and other heterogeneities, as discussed in [24]. Future research could address this problem through geological layer stratification using the universal cokriging interpolation method suggested in [50], potentially utilising the open-source geological modelling tool GemPy [51], which allows for simple parametric representation of geological strata. Spatially heterogeneous parameters within each strata could then be modelled hierarchically using a low order log-Gaussian random field to account for within-stratum variation, as demonstrated in [12].

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was funded as part of the Water Informatics Science and Engineering Centre for Doctoral Training (WISE CDT) under a grant from the Engineering and Physical Sciences Research Council (EPSRC), UK, grant number EP/L016214/1. TD was funded by a Turing AI Fellowship, UK (2TAFFP\100007). DM acknowledges support from the EPSRC Platform Grant PRISM, UK (EP/R029423/1). The authors have no competing interests. Data supporting the findings in this study are available in the Open Research Exeter (ORE, https://ore.exeter.ac.uk/repository/) data repository.

Appendix A. Preconditioned Crank–Nicolson

The preconditioned Crank–Nicolson (pCN) proposal was developed in [16] and is based on the following Stochastic Partial Differential Equation (SPDE):

$$\frac{du}{ds} = -\mathcal{K}\mathcal{L}u + \sqrt{2\mathcal{K}}\frac{db}{ds}$$

where $\mathcal{L} = \mathcal{C}^{-1}$ is the precision operator for the prior distribution μ_0, b is brownian motion with covariance operator I, and $\mathcal{K}$ is a positive operator. This equation can be discretised using the Crank–Nicolson approach to yield

$$v = u - \frac{1}{2}\delta\mathcal{K}\mathcal{L}(u+v) + \sqrt{2\mathcal{K}\delta}\xi_0$$

for white noise ξ_0 and a weight $\delta \in [0, 2]$. If we choose $\mathcal{K} = I$, we get the plain Crank–Nicolson (CN) proposal:

$$(2\mathcal{C} + \delta I)v = (2\mathcal{C} - \delta I)u + \sqrt{8\delta\mathcal{C}}\xi$$

where $\xi \sim \mathcal{N}(0, \mathcal{C})$. If we instead choose $\mathcal{K} = \mathcal{C}$, we get the pCN proposal:

$$v = \sqrt{1-\beta^2}u + \beta\xi, \quad \beta = \frac{\sqrt{8\delta}}{2+\delta}, \quad \beta \in [0, 1]$$

This is rewritten, conforming to our previous notation:

$$\boldsymbol{\theta}' = \sqrt{1-\beta^2}\boldsymbol{\theta}_i + \beta\boldsymbol{\xi}$$

Appendix B. Recovered conductivity fields

See Figs. B.15–B.21.

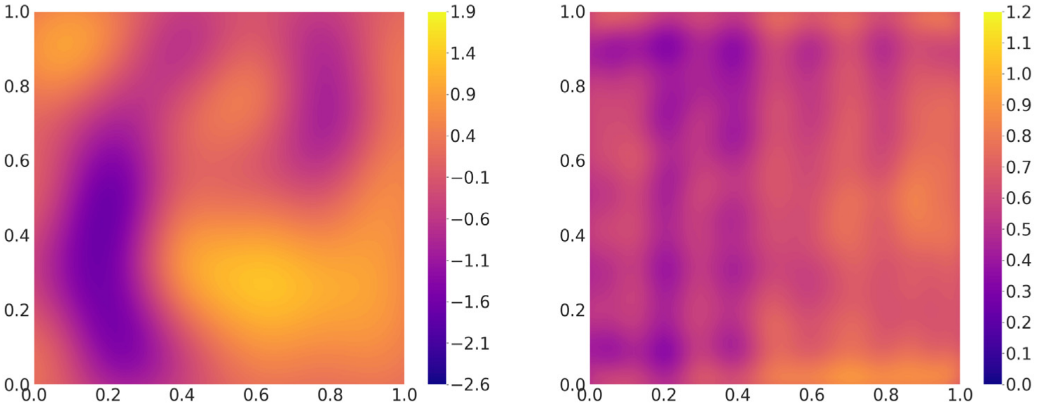


Fig. B.15. Mean (left) and variance (right) of recovered log-transmissivity for Vanilla pCN.

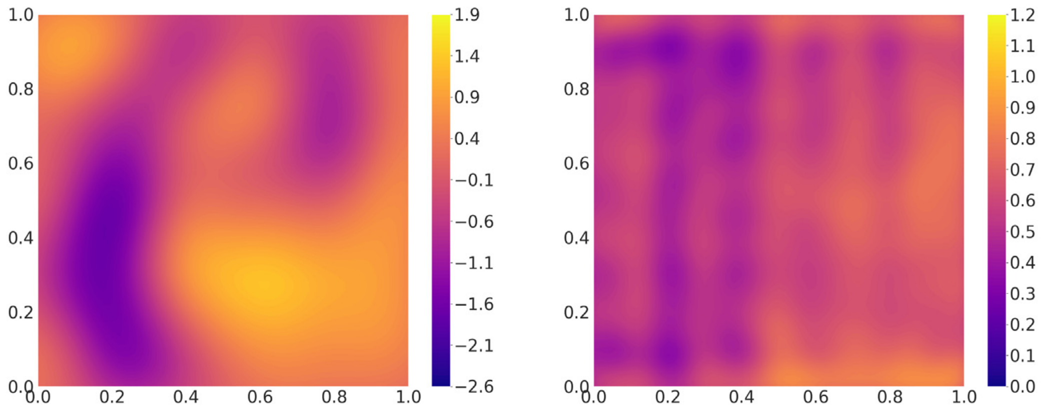


Fig. B.16. Mean (left) and variance (right) of recovered log-transmissivity for DA, $N_{DNN} = 4000$.

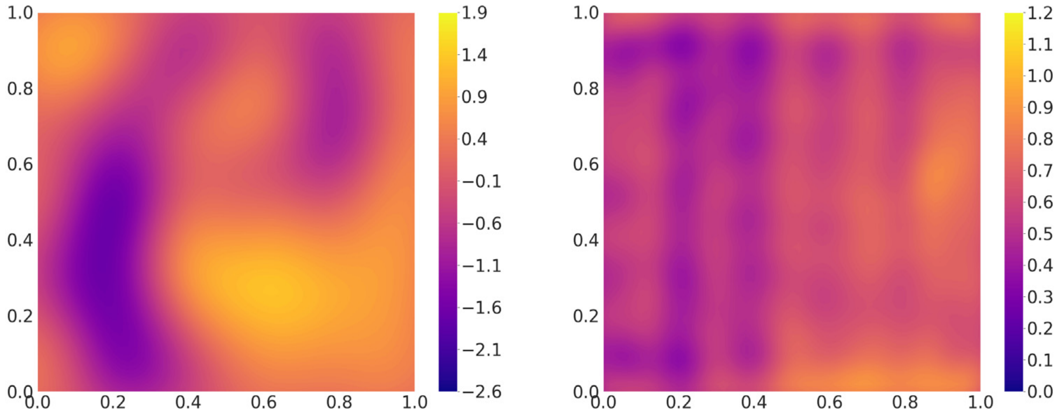


Fig. B.17. Mean (left) and variance (right) of recovered log-transmissivity for DA/EEM, $N_{DNN} = 4000$.

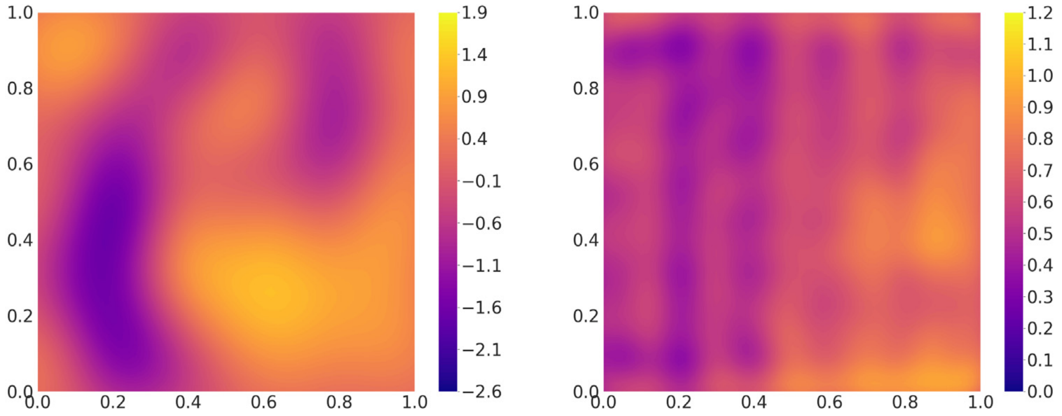


Fig. B.18. Mean (left) and variance (right) of recovered log-transmissivity for DA, $N_{DNN} = 16000$.

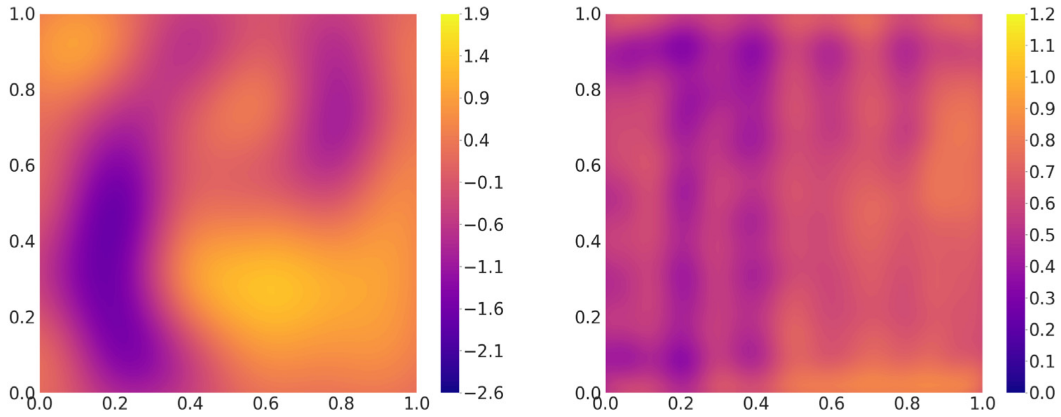


Fig. B.19. Mean (left) and variance (right) of recovered log-transmissivity for DA/EEM, $N_{DNN} = 16000$.

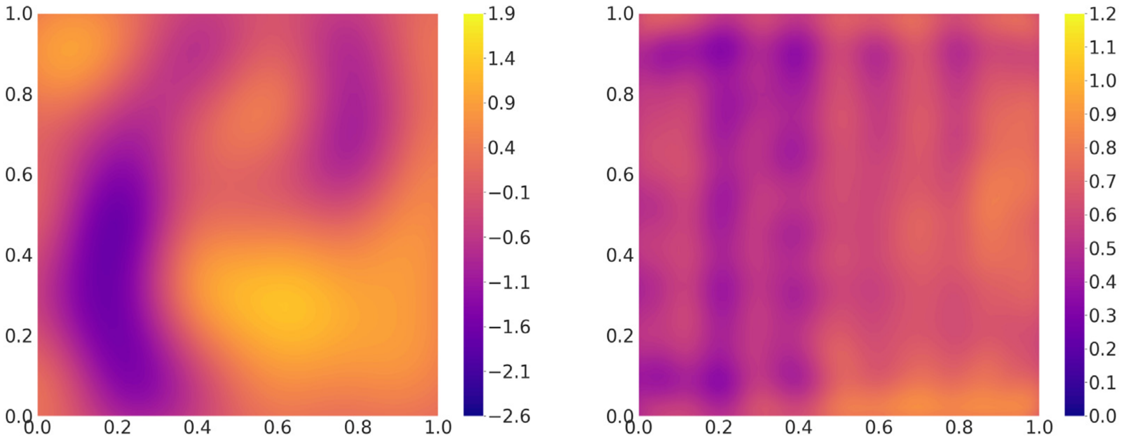


Fig. B.20. Mean (left) and variance (right) of recovered log-transmissivity for DA, $N_{DNN} = 64000$.

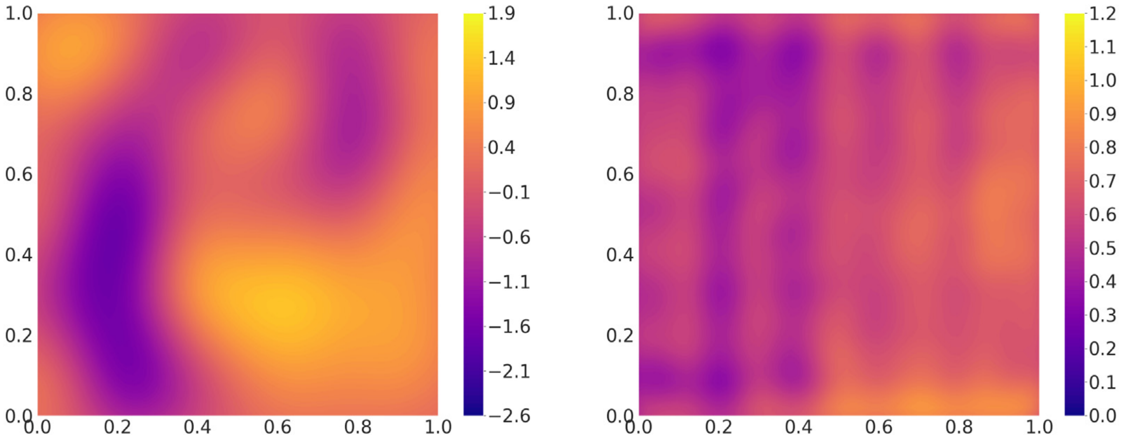


Fig. B.21. Mean (left) and variance (right) of recovered log-transmissivity for DA/EEM, $N_{DNN} = 64000$.

References

[1] M.P. Anderson, W.W. Woessner, R.J. Hunt, Applied Groundwater Modeling: simulation of Flow and Advective Transport, second ed., Academic Press, London; San Diego, CA, 2015, oCLC: ocn921253555.

[2] A.D. Woodbury, T.J. Ulrych, A full-Bayesian approach to the groundwater inverse problem for steady state flow, Water Resour. Res. 36 (8) (2000) 2081–2093, http://dx.doi.org/10.1029/2000WR900086, URL http://doi.wiley.com/10.1029/2000WR900086.

[3] G. Mariethoz, P. Renard, J. Caers, Bayesian inverse problem and optimization with iterative spatial resampling: ITERATIVE SPATIAL RESAMPLING, Water Resour. Res. 46 (11) (2010) http://dx.doi.org/10.1029/2010WR009274, URL http://doi.wiley.com/10.1029/2010WR009274.

[4] M. de la Varga, J.F. Wellmann, Structural geologic modeling as an inference problem: A Bayesian perspective, Interpretation 4 (3) (2016) SM1–SM16, http://dx.doi.org/10.1190/INT-2015-0188.1, URL http://library.seg.org/doi/10.1190/INT-2015-0188.1.

[5] C.P. Robert, G. Casella, Monte Carlo Statistical Methods, second ed., in: Springer Texts in Statistics, Springer, New York, NY, 2010, oCLC: 837651914.

[6] D. Higdon, H. Lee, C. Holloman, Markov chain Monte Carlo-based approaches for inference in computationally intensive inverse problems, in: Bayesian Statistics, Vol. 7, Oxford University Press., 2003, pp. 181–197.

[7] T.J. Dodwell, C. Ketelsen, R. Scheichl, A.L. Teckentrup, A hierarchical multilevel Markov chain Monte Carlo algorithm with applications to uncertainty quantification in subsurface flow, SIAM/ASA J. Uncertain. Quantif. 3 (1) (2015) 1075–1108, http://dx.doi.org/10.1137/130915005, URL http://epubs.siam.org/doi/10.1137/130915005.

[8] G. Detommaso, T. Dodwell, R. Scheichl, Continuous level Monte Carlo and sample-adaptive model hierarchies, 2018, arXiv:1802.07539 [math]. URL http://arxiv.org/abs/1802.07539.

[9] J. Doherty, Calibration and Uncertainty Analysis for Complex Environmental Models, 2015, oCLC: 991568728.

[10] B. Peherstorfer, K. Wilcox, M. Gunzburger, Survey of multifidelity methods in uncertainty propagation, inference, and optimization, SIAM Rev. 60 (3) (2018) 550–591.

[11] Y. Efendiev, A. Datta-Gupta, V. Ginting, X. Ma, B. Mallick, An efficient two-stage Markov chain Monte Carlo method for dynamic data integration, Water Resour. Res. 41 (12) (2005) http://dx.doi.org/10.1029/2004WR003764, URL https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2004WR003764.

[12] A. Mondal, Y. Efendiev, B. Mallick, A. Datta-Gupta, Bayesian uncertainty quantification for flows in heterogeneous porous media using reversible jump Markov chain Monte Carlo methods, Adv. Water Resour. 33 (3) (2010) 241–256, http://dx.doi.org/10.1016/j.advwatres.2009.10.010, URL https://linkinghub.elsevier.com/retrieve/pii/S0309170809001729.

[13] P. Dostert, Y. Efendiev, B. Mohanty, Efficient uncertainty quantification techniques in inverse problems for Richards' equation using coarse-scale simulation models, Adv. Water Resour. 32 (3) (2009) 329–339, http://dx.doi.org/10.1016/j.advwatres.2008.11.009, URL https://linkinghub.elsevier.com/retrieve/pii/S0309170808002121.

[14] E. Laloy, B. Rogiers, J.A. Vrugt, D. Mallants, D. Jacques, Efficient posterior exploration of a high-dimensional groundwater model from two-stage Markov chain Monte Carlo simulation and polynomial chaos expansion: Speeding up MCMC simulation of a groundwater model, Water Resour. Res. 49 (5) (2013) 2664–2682, http://dx.doi.org/10.1002/wrcr.20226, URL http://doi.wiley.com/10.1002/wrcr.20226.

[15] J.A. Christen, C. Fox, Markov chain Monte Carlo using an approximation, J. Comput. Graph. Statist. 14 (4) (2005) 795–810, http://dx.doi.org/10.1198/106186005X76983, URL http://www.tandfonline.com/doi/abs/10.1198/106186005X76983.

[16] S.L. Cotter, G.O. Roberts, A.M. Stuart, D. White, MCMC methods For functions: Modifying old algorithms to make them faster, Statist. Sci. 28 (3) (2013) 424–446, http://dx.doi.org/10.1214/13-STS421, URL http://arxiv.org/abs/1202.0709.

[17] H. Haario, E. Saksman, J. Tamminen, An adaptive metropolis algorithm, Bernoulli 7 (2) (2001) 223, http://dx.doi.org/10.2307/3318737, URL https://www.jstor.org/stable/3318737?origin=crossref.

[18] T. Cui, C. Fox, M.J. O'Sullivan, A eriori stochastic correction of reduced models in delayed acceptance MCMC, with application to multiphase subsurface inverse problems, 2018, arXiv:1809.03176 [stat]. URL http://arxiv.org/abs/1809.03176.

[19] H.-J.G. Diersch, FEFLOW: Finite Element Modeling of Flow, Mass and Heat Transport in Porous and Fractured Media, Springer Berlin Heidelberg, Berlin, Heidelberg, 2014, http://dx.doi.org/10.1007/978-3-642-38739-5, URL http://link.springer.com/10.1007/978-3-642-38739-5.

[20] H.P. Langtangen, A. Logg, Solving PDEs in Python – The FEniCS Tutorial Volume I, 2017.

[21] A.W. Harbaugh, MODFLOW-2005: The U.S. Geological Survey Modular Ground-Water Model–the Ground-Water Flow Process, Report, 2005, http://dx.doi.org/10.3133/tm6A16, URL http://pubs.er.usgs.gov/publication/tm6A16.

[22] R.A. Freeze, A stochastic-conceptual analysis of one-dimensional groundwater flow in nonuniform homogeneous media, Water Resour. Res. 11 (5) (1975) 725–741, http://dx.doi.org/10.1029/WR011i005p00725, URL http://doi.wiley.com/10.1029/WR011i005p00725.

[23] N.-O. Kitterrød, L. Gottschalk, Simulation of normal distributed smooth fields by Karhunen-Loéve expansion in combination with kriging, Stoch. Hydrol. Hydraul. 11 (6) (1997) 459–482, http://dx.doi.org/10.1007/BF02428429, URL http://link.springer.com/10.1007/BF02428429.

[24] J. Gómez-Hernández, X.-H. Wen, To be or not to be multi-Gaussian? A reflection on stochastic hydrogeology, Adv. Water Resour. 21 (1) (1998) 47–61, http://dx.doi.org/10.1016/S0309-1708(96)00031-0, URL http://linkinghub.elsevier.com/retrieve/pii/S0309170896000310.

[25] J. Kerrou, P. Renard, H.-J. Hendricks Franssen, I. Lunati, Issues in characterizing heterogeneity and connectivity in non-multiGaussian media, Adv. Water Resour. 31 (1) (2008) 147–159, http://dx.doi.org/10.1016/j.advwatres.2007.07.002, URL https://linkinghub.elsevier.com/retrieve/pii/S0309170807001236.

[26] D. Russo, M. Bouton, Statistical analysis of spatial variability in unsaturated flow parameters, Water Resour. Res. 28 (7) (1992) 1911–1925, http://dx.doi.org/10.1029/92WR00669, URL http://doi.wiley.com/10.1029/92WR00669.

[27] R.J. Hoeksema, P.K. Kitanidis, Analysis of the spatial structure of properties of selected aquifers, Water Resour. Res. 21 (4) (1985) 563–572, http://dx.doi.org/10.1029/WR021i004p00563, URL http://doi.wiley.com/10.1029/WR021i004p00563.

[28] P. Dostert, Y. Efendiev, T. Hou, W. Luo, Coarse-gradient Langevin algorithms for dynamic data integration and uncertainty quantification, J. Comput. Phys. 217 (1) (2006) 123–142, http://dx.doi.org/10.1016/j.jcp.2006.03.012, URL https://linkinghub.elsevier.com/retrieve/pii/S0021999106001380.

[29] Y.M. Marzouk, H.N. Najm, Dimensionality reduction and polynomial chaos acceleration of Bayesian inference in inverse problems, J. Comput. Phys. 228 (6) (2009) 1862–1902, http://dx.doi.org/10.1016/j.jcp.2008.11.024, URL https://linkinghub.elsevier.com/retrieve/pii/S0021999108006062.

[30] C. Scarth, S. Adhikari, P.H. Cabral, G.H. Silva, A.P.d. Prado, Random field simulation over curved surfaces: Applications to computational structural mechanics, Comput. Methods Appl. Mech. Engrg. 345 (2019) 283–301, http://dx.doi.org/10.1016/j.cma.2018.10.026, URL https://linkinghub.elsevier.com/retrieve/pii/S0045782518305309.

[31] A. Gelman (Ed.), Bayesian Data Analysis, second ed., in: Texts in Statistical Science, Chapman & Hall/CRC, Boca Raton, Fla, 2004.

[32] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller, Equation of state calculations by fast computing machines, J. Chem. Phys. 21 (6) (1953) 1087–1092, http://dx.doi.org/10.1063/1.1699114, URL http://aip.scitation.org/doi/10.1063/1.1699114.

[33] W.K. Hastings, Monte Carlo sampling methods using Markov chains and their applications, Biometrika (1970) 13.

[34] L. Katafygiotis, K. Zuev, Geometric insight into the challenges of solving high-dimensional reliability problems, Probab. Eng. Mech. 23 (2–3) (2008) 208–218, http://dx.doi.org/10.1016/j.probengmech.2007.12.026, URL https://linkinghub.elsevier.com/retrieve/pii/S0266892007000707.

[35] G.O. Roberts, J.S. Rosenthal, Examples of adaptive MCMC, J. Comput. Graph. Statist. 18 (2) (2009) 349–367, http://dx.doi.org/10.1198/jcgs.2009.06134, URL http://www.tandfonline.com/doi/abs/10.1198/jcgs.2009.06134.

[36] F. Chollet, et al., Keras, 2015, https://github.com/fchollet/keras.

[37] Theano Development Team, Theano: A python framework for fast computation of mathematical expressions, 2016, arXiv e-prints abs/1605.02688. URL http://arxiv.org/abs/1605.02688, 2016.

[38] T.M. Hansen, K.S. Cordua, Efficient Monte Carlo sampling of inverse problems using a neural network-based forward—applied to GPR crosshole traveltime inversion, Geophys. J. Int. 211 (2017) 10.

[39] D. Moghadas, A.A. Behroozmand, A.V. Christiansen, Soil electrical conductivity imaging using a neural network-based forward solver: Applied to large-scale Bayesian electromagnetic inversion, J. Appl. Geophys. 176 (2020) 104012, http://dx.doi.org/10.1016/j.jappgeo.2020.104012, URL https://linkinghub.elsevier.com/retrieve/pii/S0926985120300033.

[40] T. Hastie, R. Tibshirani, J.H. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, second ed., in: Springer Series in Statistics, Springer, New York, NY, 2009.

[41] G. Hinton, N. Srivastava, K. Swersky, Neural Networks for Machine Learning, Lecture 6a: Overview of Mini-Batch Gradient Descent, Coursera, University of Toronto, 2012.

[42] U. Wolff, Monte Carlo errors with less errors, Comput. Phys. Comm. 176 (5) (2007) 383, http://dx.doi.org/10.1016/j.cpc.2006.12.001, URL http://arxiv.org/abs/hep-lat/0306017.

[43] J. Kaipio, E. Somersalo, Statistical inverse problems: Discretization, model reduction and inverse crimes, J. Comput. Appl. Math. 198 (2) (2007) 493–504, http://dx.doi.org/10.1016/j.cam.2005.09.027, URL https://linkinghub.elsevier.com/retrieve/pii/S0377042705007296.

[44] T.M. Hansen, K.S. Cordua, B.H. Jacobsen, K. Mosegaard, Accounting for imperfect forward modeling in geophysical inverse problems — Exemplified for crosshole tomography, Geophysics 79 (3) (2014) 22.

[45] T. Cui, C. Fox, M.J. O'Sullivan, Bayesian calibration of a large-scale geothermal reservoir model by a new adaptive delayed acceptance Metropolis Hastings algorithm: ADAPTIVE DELAYED ACCEPTANCE METROPOLIS-HASTINGS ALGORITHM, Water Resour. Res. 47 (10) (2011) http://dx.doi.org/10.1029/2010WR010352, URL http://doi.wiley.com/10.1029/2010WR010352.

[46] C. Köpke, J. Irving, A.H. Elsheikh, Accounting for model error in Bayesian solutions to hydrogeophysical inverse problems using a local basis approach, Adv. Water Resour. 116 (2018) 195–207, http://dx.doi.org/10.1016/j.advwatres.2017.11.013, URL https://linkinghub.elsevier.com/retrieve/pii/S0309170817308308.

[47] J. Brynjarsdóttir, A. O'Hagan, Learning about physical parameters: The importance of model discrepancy, Inverse Problems 30 (11) (2014) 114007, http://dx.doi.org/10.1088/0266-5611/30/11/114007, URL http://stacks.iop.org/0266-5611/30/i=11/a=114007?key=crossref.7b886360dda7b385609c577ad82450aa.

[48] M.D. McKay, R.J. Beckman, W.J. Conover, A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, Technometrics 21 (2) (1979) 239–245, URL http://www.jstor.org/stable/1268522.

[49] M.D. Morris, T.J. Mitchell, Exploratory designs for computational experiments, J. Statist. Plann. Inference 43 (3) (1995) 381–402, http://dx.doi.org/10.1016/0378-3758(94)00035-T, URL https://linkinghub.elsevier.com/retrieve/pii/037837589400035T.

[50] C. Lajaunie, G. Courrioux, L. Manuel, Foliation fields and 3D cartography in geology: Principles of a method based on potential interpolation, Math. Geol. 29 (4) (1997) 571–584, http://dx.doi.org/10.1007/BF02775087, URL http://link.springer.com/10.1007/BF02775087.

[51] M. de la Varga, A. Schaaf, F. Wellmann, GemPy 1.0: Open-source stochastic geological modeling and inversion, Geosci. Model Dev. 12 (1) (2019) 1–32, http://dx.doi.org/10.5194/gmd-12-1-2019, URL https://www.geosci-model-dev.net/12/1/2019/.

6. Where to Drill Next? A Dual-Weighted Approach to Adaptive Optimal Design of Groundwater Surveys

This journal paper, currently undergoing its second round of reviews in Advances in Water Resources, explores a novel application of Bayesian uncertainty quantification to inform engineering decision support systems for groundwater surveying. The method exploits an uncertainty function in conjunction with the expected solution to an adjoint state equation to determine the location of the next monitoring well, which is optimal with respect to some quantity of interest. The computational experiments employ the MLDA algorithm developed in Chapters 3 and 4.

The idea was conceived by Tim Dodwell and me. I developed the computer code, conducted the experiments and wrote the paper. Tim Dodwell provided feedback during the research process. Both authors contributed to the editing.

Where to Drill Next? A Dual-Weighted Approach to Adaptive Optimal Design of Groundwater Surveys

A Preprint

Mikkel B. Lykkegaard[*]
Centre for Water Systems and
Institute for Data Science and AI
University of Exeter
Exeter, EX44QF
m.lykkegaard@exeter.ac.uk

Tim J. Dodwell
The Alan Turing Institute and
Institute for Data Science and AI
University of Exeter
Exeter, EX44QF
tdodwell@turing.ac.uk

February 24, 2022

Abstract

We present a novel approach to adaptive optimal design of groundwater surveys – a methodology for choosing the location of the next monitoring well. Our dual-weighted approach borrows ideas from Bayesian Optimisation and goal-oriented error estimation to propose the next monitoring well, given that some data is already available from existing wells. Our method is distinct from other optimal design strategies in that it does not rely on Fisher Information and it instead directly exploits the posterior uncertainty and the expected solution to a dual (or *adjoint*) problem to construct an acquisition function that optimally reduces the uncertainty in the model as a whole and some engineering quantity of interest in particular. We demonstrate our approach in the context of 2D groundwater flow example and show that the dual-weighted approach outperforms the baseline approach with respect to reducing the error in the posterior estimate of the quantity of interest.

***K*eywords** Adaptive Optimal Design · Groundwater Surveying · Uncertainty Quantification · Bayesian Inverse Problems · Adjoint State Equations

[*] Corresponding author.

1 Introduction

In this paper, we present a novel approach to optimally choosing the location of the next monitoring well when conducting a groundwater survey. Establishing a monitoring well is generally costly, and depends on the specific geological context and the required penetration depth, and choosing the most informative location for each well is a critical task when designing a groundwater survey. Groundwater surveying and modelling are intrinsically imbued with uncertainty and solutions and predictions are without exception non-unique [1]. Hence, in this paper we assume the perspective that a useful sampling location is one that most significantly reduces the uncertainty in the solution, while simultaneously having a substantial influence on some quantity of interest (QoI). While multiple non-invasive and relatively inexpensive methods for groundwater surveying exist [2, 3, 4, 5], these methods all involve solving an inverse problem to reconstruct the hydraulic head, which introduces an additional layer of uncertainty. Hence, in this work, we focus on the problem of determining aquifer characteristics from direct point measurements of hydraulic head and flux from monitoring wells, and how to optimally choose the locations of such wells, given existing data. While the method is here contextualised within this particular problem, it can easily be generalised to any setting where a continuous function and a derived QoI are approximated with point measurements.

In the "classic" theory of optimal design, we often distinguish between optimality criteria that minimise the estimated parameter variances (e.g. $A-$, $D-$ and $E-$optimality) and those that minimise the prediction variance (e.g. $G-$, $V-$ and $I-$optimality) [6, 7]. Since in this study we are primarily concerned with the prediction variance, the method presented here belongs in the latter category. In this context, our method can broadly be considered $G-$optimal, since our vanilla acquisition function targets the location of the highest posterior dispersion [see e.g. 7]. However, rather than iteratively searching for a design that maximises an optimality criterion, we directly utilise a posterior dispersion estimate to construct an acquisition function. We remark that while there are some abstract parallels between the method presented here and classic optimal design, our method is probably better understood in the context of Bayesian Optimisation, as discussed later. Additionally, the classic optimal design approach is typically centered around the problem of choosing an experimental design that is optimal with respect to an optimality criterion, *before taking any measurements*. In this paper, we take an adaptive approach and assume that some measurements are already available, and we want to propose optimal *new* sampling locations, given the data we already have. How the initial measurement locations are optimally chosen is beyond the scope of this paper, but we refer to e.g. Cox and Reid [8], Pukelsheim [6], Myers et al. [7] for an extensive overview of optimal design of experiments. We remark that our dual-weighted method could in theory be employed to choose initial measurement locations, but in that case the dispersion of the solution would be constrained only by the prior distribution of parameters and the constraints imposed by the constitutive equations. In this case, the method presented herein may be used in conjunction with some space-filling design strategy or using local penalisation functions as described in Section 2.3.2. However, either of these workarounds would require an informed prior to work well.

We recycle the notion from classic optimal design that the information gain is driven by minimising the dispersion of a target distribution [9]. However, rather than integrating out all possible measurements and model parameters to find the utility of a given design, we take a simpler approach. Namely, we use a Monte Carlo estimate of the (current) posterior dispersion of the solution to a Partial Differential Equation (PDE) (or some appropriate function thereof) as an acquisition function. The underlying rationale being that if we wish to know more about the distribution of our solution, the most useful place to take a new sample is at the point of the highest posterior uncertainty.

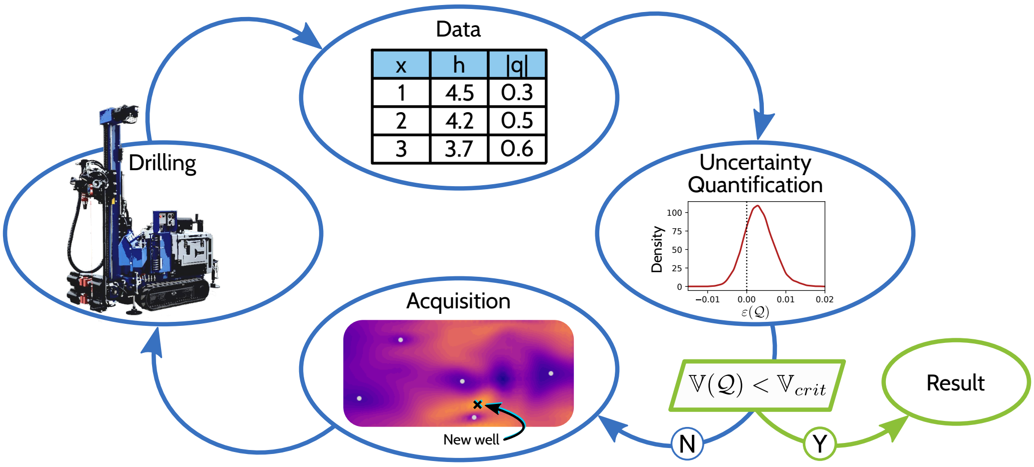
In this context, our Vanilla approach (see Section 2.3.1) is not dissimilar to the maximum entropy approach to the optimal sensor placement problem [10], where sensors are added at the point of the highest uncertainty of some probabilistic function that is fitted to current sensor measurements, for example a Gaussian Process (GP) emulator. While this strategy will typically place many sensors at the boundaries of the sampling space in the context of adaptive GP fitting [11], this is not necessarily the case when targeting the uncertainty of the solution to a PDE, since that will be constrained by boundary conditions. The sensor placement problem has been studied extensively in the context of GP emulators, and multiple improvements to the maximum entropy approach have been made (see e.g. Krause et al. [12], Beck and Guillas [13], Mohammadi et al. [11]). However, since our objective is to minimise the uncertainty of a PDE-derived QoI, and not a GP emulator, many of the recent developments are not immediately applicable, since they are tailored for use with a GP emulator. Hence, the Vanilla approach presented herein can be considered a reformulation of the original maximum entropy approach, particularly tailored for the (probabilistic) solution of a PDE.

Our method (see Section 2.3) borrows ideas from other fields, not obviously related to classic optimal design. First, our adaptive optimal design approach is formulated in terms of an acquisition function, a term typically associated with Bayesian Optimisation (BO, Močkus [14], Frazier [15]). Moreover, our approach uses ideas from both prior-guided BO [16] and batch BO [17], the similarities with which are discussed in Section 2.3.3. While in the context of BO, the aim is to find the maximum or minimum of some function that is expensive to evaluate, our objective is to simply reduce the uncertainty of our model predictions. Hence, our vanilla acquisition function addresses solely the uncertainty of some target function, and not the function value itself. Second, our approach is inspired by the goal-oriented error-estimation used in mesh-adaptation for PDEs [18, 19], where the intention is to refine a mesh locally and parsimoniously to reduce the simulation error with respect to some QoI using an influence function that is the solution to an adjoint PDE. This approach, however, is most useful for forward problems, where the domain and coefficients are well-known, and the groundwater flow problem is typically not of this kind. Instead, we use the same approach of computing an influence function with respect to the QoI to determine, not where the mesh should be refined, but from where we need more data.

The idea of exploiting the adjoint or *dual* problem to minimise the posterior uncertainty with respect to a QoI was first explored by Attia et al. [20] in a similar context as our model problem. However, there are several crucial differences between their approach and the one presented in this paper. First, their method is set in the “classic” optimal design context, where a number of sampling locations are determined before taking any measurements, based on the maximising the expected information gain according to some criterion derived from the Fisher Information matrix. Second, since only a finite number of designs can be explored this way, the prospective sampling locations are fixed to a relatively coarse grid. Finally, the approach described in Attia et al. [20] requires the adjoint operator to be linear – an assumption which is suitable for only a subset of QoIs.

We employ Markov Chain Monte Carlo (MCMC) techniques (see Section 2.1) to generate samples from the posterior distribution of the model parameters given the data $\pi(\theta|\mathbf{d})$, where the model parameters (θ) in this case describe hydraulic conductivity and the data $(\mathbf{d})$ are point measurements of hydraulic head and flux (see Section 2.2). Even if the model parameters themselves are of secondary interest to a given problem, we can use the MCMC samples to construct Monte Carlo estimates of any parameter-derived quantity or function, such as the hydraulic flux across a boundary, or the peak concentration of a contaminant at a well. Additionally, unlike traditional inversion techniques, MCMC allows for rigorously quantifying the uncertainty of the inverse problem, which is useful in the context of engineering decision support systems, in particular risk assessment studies. We believe that there are many unexploited application opportunities tangential to the study of Bayesian posteriors and demonstrate, in this paper, one such application.

Figure 1 illustrates the proposed workflow at a high level, where new wells are sequentially established at locations of high uncertainty and influence on a QoI, as dictated by the acquisition function. This paper is mainly concerned with the construction of optimal acquisition functions based on the posterior information which would be immediately available from quantifying the uncertainty of the Bayesian inverse problem.


Data
Drilling
Uncertainty Quantification
Density
100
75
50
25
0
−0.01
0.00
0.01
0.02
$\varepsilon(\mathcal{Q})$
Acquisition
New well
$\mathbb{V}(\mathcal{Q}) < \mathbb{V}_{crit}$
N
Y
Result

x	h	|q|
1	4.5	0.3
2	4.2	0.5
3	3.7	0.6

Figure 1: Conceptual diagram of the proposed adaptive optimal design workflow. Here, $\mathbb{V}(\mathcal{Q})$ denotes the variance of the quantity of interest $\mathcal{Q}$ and $\mathbb{V}_{crit}$ the desired critical variance.

In the following sections, we briefly summarise the theory of Bayesian inverse problems, MCMC and groundwater flow modelling. We then outline the proposed methodology and demonstrate the effectiveness of methodology on a synthetic example. We show that efficient acquisition functions can easily be constructed from information that would already be available from solving the Bayesian inverse problem using MCMC. The method avoids many of the complex calculations that are associated with classic optimal design and exploits information about the Bayesian posterior in a direct and straightforward way.

2 Theory

In this section, we first briefly outline the framework of Bayesian inverse problems and Markov Chain Monte Carlo (MCMC), a popular technique employed to draw samples from the Bayesian posterior. We then summarise the fundamentals of groundwater flow modelling for steady-state groundwater flow in a confined aquifer using the Finite Element Method (FEM). Finally, we describe our novel approach to adaptive optimal design of groundwater surveys.

2.1 Bayesian Inversion

A Bayesian inverse problem can be stated compactly as: Given some data $\mathbf{d}$, find the distribution $\pi(\theta|\mathbf{d})$ with model parameters $\theta \in \Theta$, where Θ is the parameter space, so that

$$\mathbf{d} = \mathcal{F}(\theta) + \epsilon \tag{1}$$

where $\mathcal{F}(\theta)$ is the model output and ϵ is the measurement error, which is typically assumed to be Gaussian. Bayes theorem then states that

$$\pi(\theta|\mathbf{d}) = \frac{\pi_{\mathrm{p}}(\theta)\mathcal{L}(\mathbf{d}|\theta)}{\pi(\mathbf{d})} \tag{2}$$

where $\pi(\theta|\mathbf{d})$ is referred to as the *posterior* distribution, $\pi_{\mathrm{p}}(\theta)$ is *prior* distribution, encapsulating what we already know about our model parameters and $\mathcal{L}(\mathbf{d}|\theta)$ is called the *likelihood*, essentially a measure of misfit between the model output $\mathcal{F}(\theta)$ and the data $\mathbf{d}$. While the the so-called *evidence* $\pi(\mathbf{d}) = \int_{\Theta} \pi_{\mathrm{p}}(\theta)\ \mathcal{L}(\mathbf{d}|\theta)\ d\theta$ is generally infeasible or impossible to determine in most real-world scenarios, various sampling techniques allows us to make statistical inferences from $\pi(\theta|\mathbf{d})$ anyway. Examples include Importance Sampling (IS) and Markov Chain Monte Carlo (MCMC) methods. While these methods are not the object of this study, a short summary of the main ideas of MCMC, which is the specific method employed for inversion in this study, is provided for completeness.

In MCMC we exploit that $\pi(\mathbf{d})$ is constant and does not depend on the parameters θ. We can therefore write

$$\pi(\theta|\mathbf{d}) \propto \pi_{\mathrm{p}}(\theta)\mathcal{L}(\mathbf{d}|\theta) \tag{3}$$

or equivalently, for $x, y \in \Theta$

$$\frac{\pi(y|\mathbf{d})}{\pi(x|\mathbf{d})} = \frac{\pi_{\mathrm{p}}(y)\mathcal{L}(\mathbf{d}|y)}{\pi_{\mathrm{p}}(x)\mathcal{L}(\mathbf{d}|x)} \tag{4}$$

We then introduce a *transition kernel* or *proposal distribution* $q(y|x)$, allowing us to transition from one state x to another y. Repeatedly applying the transition kernel $q(y|x)$ followed by an accept/reject step prescribed by equation (5) we construct a Markov chain where the samples, after an initial *burn-in*, are precisely from the required distribution $\pi(\theta|\mathbf{d})$. Here, burn-in refers to the initial MCMC samples which are discarded, since they may not be representative of the equilibrium distribution of the Markov chain. This procedure is described in the box below [21, 22, 23].

The Metropolis-Hastings Algorithm, $\theta^{(0)} \sim \pi_{\mathrm{p}}(\theta)$, for $i = 0, \ldots, N$:

1. Given a parameter realisation $\theta^{(i)}$ and a transition kernel $q(\theta'|\theta^{(i)})$, generate a proposal θ'.
2. Compute the acceptance probability of the proposal given the previous realisation:
$$\alpha(\theta'|\theta^{(i)}) = \min\left\{1, \frac{\pi_{\mathrm{p}}(\theta')\mathcal{L}(\mathbf{d}|\theta')}{\pi_{\mathrm{p}}(\theta^{(i)})\mathcal{L}(\mathbf{d}|\theta^{(i)})} \frac{q(\theta^{(i)}|\theta')}{q(\theta'|\theta^{(i)})}\right\} \tag{5}$$
3. If $u \sim U(0,1) > \alpha$ then set $\theta^{(i+1)} = \theta^{(i)}$, otherwise, set $\theta^{(i+1)} = \theta'$.

The acceptance probability (Eq. 5) ensures that the algorithm is in detailed balance with the target (posterior) distribution $\pi(\theta|\mathbf{d})$. See e.g. Liu [24, Sec. 5.3] for more details. Note that when the measurement error ϵ is Gaussian, $\epsilon \sim \mathcal{N}(0, \Sigma_\epsilon)$, which we assume in the experiment in Section 3, then the (unnormalised) likelihood functional takes the following form:

$$\mathcal{L}(\mathbf{d}|\theta) \propto \exp\left(-\frac{1}{2}(\mathcal{F}(\theta) - \mathbf{d})^T \Sigma_\epsilon^{-1} (\mathcal{F}(\theta) - \mathbf{d})\right). \tag{6}$$

In this study we employ a number of extensions to the Metropolis-Hastings algorithm to speed up inference, namely the Delayed Acceptance (DA, [25]) algorithm with finite subchains [26, 27], also referred to as the *surrogate transition method* by Liu [24]. The DA algorithm exploits an approximate forward model (or Reduced Order Model, ROM) $\hat{\mathcal{F}}$ to filter MCMC proposals before evaluating them with the fully resolved forward model $\mathcal{F}$, resulting in a reduction in computational cost. Moreover, we employ a state-independent Approximation Error Model (AEM) to probabilistically correct for model reduction errors introduced by the approximate model, as described by Cui et al. [28]. Finally, we use the Adaptive Metropolis (AM) algorithm as the transition kernel [29]. In this work, we used the open-source DA MCMC framework `tinyDA`[†] to perform the MCMC sampling.

2.2 Groundwater Flow

The groundwater flow equation for steady flow in a confined, inhomogeneous aquifer occupying the domain Ω with boundary Γ can be written as the scalar elliptic partial differential equation

$$-\nabla \cdot k(\mathbf{x}) \nabla u(\mathbf{x}) = g(\mathbf{x}), \quad \text{for all} \quad \mathbf{x} \in \Omega, \tag{7}$$

[†] https://github.com/mikkelbue/tinyDA

subject to boundary conditions on $\Gamma = \Gamma_D \cup \Gamma_N$ with the constraints

$$u(\mathbf{x}) = u_D(\mathbf{x}) \quad \text{on} \quad \Gamma_D \quad \text{and} \quad -\left(k(\mathbf{x})\nabla u(\mathbf{x})\right)\cdot \mathbf{n} = q_N(\mathbf{x}) \quad \text{on} \quad \Gamma_N. \tag{8}$$

Here, $k(\mathbf{x})$ is the hydraulic conductivity, $u(\mathbf{x})$ is the hydraulic head, $g(\mathbf{x})$ are sources and sinks, and Γ_D and Γ_N are boundaries with Dirichlet and Neumann conditions, respectively (see e.g. Diersch [30]). If θ somehow parameterises the conductivity, then we have $k(\mathbf{x}) = k(\mathbf{x}, \theta)$. This equation can be converted into the weak form by multiplying with a test function $v \in H^1(\Omega)$ and integrating by parts:

$$\int_\Omega \nabla v \cdot (k(\mathbf{x}, \theta) \cdot \nabla u)\, d\mathbf{x} + \int_{\Gamma_N} v\, q_N(\mathbf{x})\, ds = \int_\Omega v\, g(\mathbf{x})\, d\mathbf{x}, \quad \forall v \in H^1(\Omega) \tag{9}$$

subject to the boundary condition $u(\mathbf{x}) = u_D(\mathbf{x})$ on Γ_D, where $H^1(\Omega)$ is the Hilbert space of weakly differentiable functions on Ω. We approximate the solution $u(\mathbf{x})$ in a finite element space $V_\tau \subset H^1(\Omega)$ on a finite element mesh $\mathcal{Q}_\tau(\Omega)$, defined by piecewise linear Lagrange polynomials $\{\phi_i(\mathbf{x})\}_{i=1}^M$ associated with the M finite element nodes. This can be rewritten as a sparse system of equations

$$\mathbf{A}(\theta)\mathbf{u} = \mathbf{b} \quad \text{where} \quad A_{ij} = \int_\Omega \nabla\phi_i(\mathbf{x}) \cdot k(\mathbf{x}, \theta)\nabla\phi_j(\mathbf{x}) d\mathbf{x} \quad \text{and} \tag{10}$$

$$b_i = -\int_{\Gamma_N} \phi_i(\mathbf{x})\, q_N(\mathbf{x})\, ds + \int_\Omega \phi_i(\mathbf{x})\, g(\mathbf{x})\, d\mathbf{x} \tag{11}$$

where $\mathbf{A}(\theta) \in \mathbb{R}^{M\times M}$ is the global stiffness matrix and $\mathbf{b} \in \mathbb{R}^M$ is the load vector. The solution to this system $\mathbf{u} := [u_1, u_2, \ldots, u_M] \in \mathbb{R}^M$ represents the hydraulic head at each node, which can be interpolated to the entire domain using the finite element shape functions: $u(\mathbf{x}) = \sum_{i=1}^M u_i\phi_i(\mathbf{x})$. In our numerical experiments, we used the open-source high-performance finite elements package `FEniCS` [31] to solve these equations.

2.3 Adaptive Optimal Design

The overarching research question of this paper is this: if we want to collect more data to reduce the variance in our posterior Monte Carlo estimates, where in the modelling domain Ω should we do it, to maximise the benefit of the new borehole? More formally, if we let t denote the current design of the survey, so that $\mathbf{d}_t$ and $\pi_t(\theta|\mathbf{d}_t)$ denote, respectively, the data and posterior distribution corresponding to that design, we want to find the next sampling point $\mathbf{x}^\star$ that constrains $\pi_{t+1}(\theta|\mathbf{d}_{t+1})$ in an optimal way, after setting $\mathbf{d}_{t+1} = (\mathbf{d}_t, d^\star)^T$, where $d^\star$ is the newly collected data at $\mathbf{x}^\star$.

2.3.1 "Vanilla" Approach

As outlined in section 2.1, Bayesian inversion allows us to construct the posterior distribution of parameters given the data $\pi_t(\theta|\mathbf{d}_t)$. If the inversion was completed using MCMC, and obtaining the model output $\mathcal{F}(\theta)$ involved solving some partial differential equation with solution $u(\mathbf{x})$, we can cache these solutions during sampling, and would after sampling possess a set of pairs $\{(\theta^{(i)}, u^{(i)}(\mathbf{x}))\}_{i=0}^{N^\dagger}$. Since $\{\theta^{(i)}\}_{i=0}^{N^\dagger}$ are distributed exactly according to $\pi_t(\theta|\mathbf{d}_t)$, so are any functions of θ, such as $u(\mathbf{x})$. Here, $N^\dagger$ is the number of MCMC samples after discarding the burn-in. Hence, we can easily obtain Monte Carlo estimates for

$$\mathbb{E}_{\pi_t(\theta|\mathbf{d}_t)}[u(\mathbf{x},\theta)] \quad \text{and} \quad \mathbb{D}_{\pi_t(\theta|\mathbf{d}_t)}[u(\mathbf{x},\theta)]$$

Here, $\mathbb{D}$ signifies some measure of statistical dispersion, for example variance, standard deviation, or entropy. We could, in accordance with the maximum entropy approach [10], postulate that the accuracy of our inversion is driven by the dispersion in $u(\mathbf{x})$ and hence we could solve the following optimisation problem

$$\mathbf{x}^\star = \operatorname*{arg\,max}_{\mathbf{x}\in\Omega} \mathbb{D}_{\pi_t(\theta|\mathbf{d}_t)}[u(\mathbf{x},\theta)] \tag{12}$$

2.3.2 Dual-Weighted Approach

The simple approach outlined above will improve the general quality of $u(\mathbf{x})$, but it is limited by the fact that it is not tailored for a particular quantity of interest $\mathcal{Q}$ and this is where the *dual weighted* approach comes into play. In this context, rather than simply sampling from places with high uncertainty, we aim to pick sampling points that also have a high expected influence on our quantity of interest $\mathcal{Q}$. This is exactly the problem, that *adjoint* or *dual* state methods aim to solve [32].

Suppose in a particular application, we are interested in estimating a particular quantity of interest $\mathcal{Q}(u)$, which we can write as a functional of the solution. For example, if our quantity of interest is the hydraulic head around a point $\mathbf{x}' \in \Omega$, we could choose

$$\mathcal{Q}_{\mathbf{x}'}(u) = \int_\Omega u(\mathbf{x}) \exp\left(-\frac{(\mathbf{x}-\mathbf{x}')^2}{\lambda}\right)\, d\mathbf{x} \tag{13}$$

for some sufficiently small length scale λ. This, however, is a trivial problem, since if the quantity of interest is the hydraulic head at some point, we can just place our monitoring well at that point and measure it. It would be much more useful to target a quantity of interest that we cannot measure directly. Hence, in this study we consider flux over a boundary Γ' with the following functional:

$$\mathcal{Q}_{\Gamma'}(u) = \int_{\Gamma'} [-k(\mathbf{x},\theta)\cdot\nabla u(\mathbf{x})]\cdot\mathbf{n}\, ds \tag{14}$$

The adjoint state equation associated with Eq. (14) is

$$\nabla\cdot k\nabla\omega = 0 \tag{15}$$

subject to the boundary conditions

$$
\begin{aligned}
\omega_D(\mathbf{x}) &= 0 && \text{on} \quad \Gamma_D \setminus \Gamma' \\
\omega_{\Gamma'}(\mathbf{x}) &= 1 && \text{on} \quad \Gamma' \\
q_N^\omega(\mathbf{x}) &= (k(\mathbf{x})\nabla\omega(\mathbf{x})) \cdot \mathbf{n} = 0 && \text{on} \quad \Gamma_N.
\end{aligned}
$$

The solution $\omega(\mathbf{x})$ is called the adjoint state or *influence* function. Please refer to Sykes et al. [33] and A for details on the derivation of the adjoint state equation and its associated boundary conditions. Integrating by parts and multiplying with a test function $v \in H^1(\Omega)$, we arrive at the weak form of the adjoint state equation:

$$\int_\Omega \nabla v \cdot (k(\mathbf{x},\theta) \cdot \nabla\omega)\, d\mathbf{x} + \int_{\Gamma_N} v\, q_N^\omega(\mathbf{x})\, ds \; = 0, \; \forall v \in H^1(\Omega) \tag{16}$$

subject to boundary conditions $\omega_D(\mathbf{x}) = 0 \, \text{on} \, \Gamma_D \setminus \Gamma' \, \text{and} \, \omega_{\Gamma'}(\mathbf{x}) = 1 \, \text{on} \, \Gamma'$. Given some conductivity parameters θ, (16) can be discretised using the same finite element grid as (10), leading to the following sparse system of equations:

$$\mathbf{A}(\theta)\omega = \mathbf{b}_\omega \quad \text{where} \quad A_{ij} = \int_\Omega \nabla\phi_i(\mathbf{x}) \cdot k(\mathbf{x},\theta)\nabla\phi_j(\mathbf{x}) d\mathbf{x} \quad \text{and} \tag{17}$$

$$b_{\omega,i} = -\int_{\Gamma_N} \phi_i(\mathbf{x})\, q_N^\omega(\mathbf{x})\, ds. \tag{18}$$

It is important to note here, that the stiffness matrix $\mathbf{A}(\theta)$, since the steady-state groundwater flow equation is *self-adjoint*, is exactly the same as in equation (10), and the assembled system can hence be partially recycled when solving both equations. However, since the boundary conditions for the adjoint state equation are different than for the primal problem, care must be taken when assembling the adjoint system of equations. After solving this system of equations, the influence function can be interpolated to the entire domain using our finite element shape functions:

$$\omega(\mathbf{x}) = \sum_{i=1}^{M} \omega_i \phi_i(\mathbf{x}) \quad \text{where} \quad \omega = [\omega_1, \omega_2, \ldots, \omega_M]^T.$$

The influence function is commonly interpreted as the sensitivity of the quantity of interest to a unit point source anywhere on the domain [33, 34], or in this particular case as the sensitivity of flow anywhere on the domain to the boundary condition. Broadly speaking, the influence function directs us towards areas of the modelling domain with a potentially high influence on our quantity of interest, which is what we required for our dual-weighted approach.

We note that $\omega(\mathbf{x})$ is now a random function which depends on model parameters θ, and we can obtain estimates for $\mathbb{E}_{\pi_t(\theta|\mathbf{d}_t)}[\omega(\mathbf{x},\theta)]$. Hence, we propose the following acquisition function

$$\mathbf{x}^\star = \arg\max_{\mathbf{x}\in\Omega} \mathbb{D}_{\pi_t(\theta|\mathbf{d}_t)}[u(\mathbf{x},\theta)] \cdot |\mathbb{E}_{\pi_t(\theta|\mathbf{d}_t)}[\omega(\mathbf{x},\theta)]|. \tag{19}$$

where $|\cdot|$ denotes the absolute value. We use the absolute value of the expectation of the influence function to make sure that the weighting is always positive, since $\omega(\mathbf{x},\theta)$ is not always positive for other adjoint equations. We call this approach dual-weighted, since we are essential re-weighting the dispersion $\mathbb{D}_{\pi_t(\theta|\mathbf{d}_t)}[u(\mathbf{x},\theta)]$, by the expected solution of the dual problem. Figure 2 illustrates the different steps in the proposed adaptive optimal design procedure.

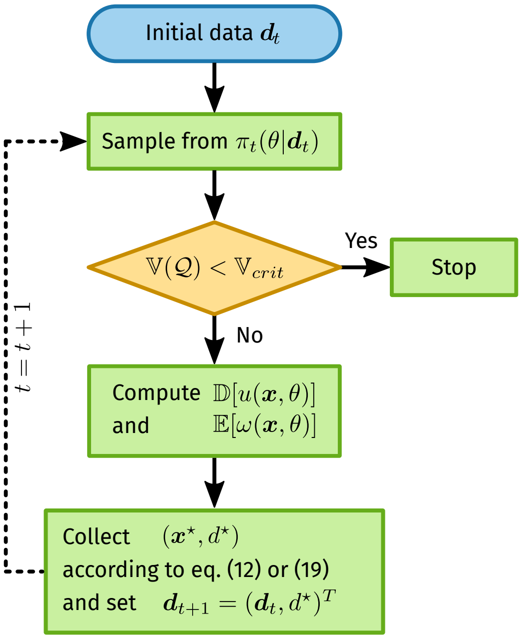

Initial data $\boldsymbol{d}_t$
Sample from $\pi_t(\theta|\boldsymbol{d}_t)$
$\mathbb{V}(\mathcal{Q}) < \mathbb{V}_{crit}$
Yes
Stop
No
$t = t + 1$
Compute $\mathbb{D}[u(\boldsymbol{x},\theta)]$
and $\mathbb{E}[\omega(\boldsymbol{x},\theta)]$
Collect $(\boldsymbol{x}^\star, d^\star)$
according to eq. (12) or (19)
and set $\boldsymbol{d}_{t+1} = (\boldsymbol{d}_t, d^\star)^T$

Figure 2: Proposed adaptive optimal design procedure. As in Figure 1, $\mathbb{V}(\mathcal{Q})$ denotes the variance of the quantity of interest $\mathcal{Q}$ and $\mathbb{V}_{crit}$ the desired critical variance.

2.3.3 Remarks

(1) The dual-weighted approach can be considered a hybrid between the goal-oriented error estimation employed for mesh-adaptation in the context of various expensive and mesh-sensitive PDE problems (see e.g. Prudhomme and Oden [18], Oden and Prudhomme [19]), and Bayesian Optimisation (BO), typically used to optimise some unknown function approximated with sparse and/or noisy data (see e.g. Močkus [14], Frazier [15]). In this context, our dual-weighted approach could be framed as a form of prior-guided BO [16], where $\omega(\mathbf{x})$ broadly represents our prior belief that any point $\mathbf{x}$ constitutes a "good" sampling location. However, we remark that in our formulation $\omega(\mathbf{x})$ is not a probability distribution but a random weighting function.

(2) In the above formulations, we have chosen the dispersion of the hydraulic head $\mathbb{D}_{\pi_t(\theta|\mathbf{d}_t)}[u(\mathbf{x},\theta)]$ as the function representing uncertainty in the model. Other sensible choices of uncertainty metrics would be the dispersion of the hydraulic conductivity $\mathbb{D}_{\pi_t(\theta|\mathbf{d}_t)}[k(\mathbf{x},\theta)]$, or of some norm of the flux $\mathbb{D}_{\pi_t(\theta|\mathbf{d}_t)}[\|\mathbf{q}(\mathbf{x},\theta)\|_p]$.

(3) Since sampling from $\pi_t(\theta|\mathbf{d}_t)$ can be computationally expensive, it may be desirable to pick multiple new sampling locations at each step of the algorithm. Denote the number of new sampling locations in each such batch acquisition as $N^\star$. Then this can be achieved by penalising the acquisition function by some local penalisation functions $\{\psi_{\mathbf{x}_i^\star}(\mathbf{x})\}_{i=1}^{N^\star-1}$, centered on the previous sampling points $\{\mathbf{x}_i^\star\}_{i=1}^{N^\star-1}$ of the current batch, as described in Gonzalez et al. [17]. This approach would yield the following dual-weighted batch acquisition function for $\{\mathbf{x}_i^\star\}_{i=2}^{N}$:

$$\mathbf{x}_i^\star = \underset{\mathbf{x}\in\Omega}{\arg\max}\ \mathbb{D}_{\pi_t(\theta|\mathbf{d}_t)}[u(\mathbf{x},\theta)] \cdot |\mathbb{E}_{\pi_t(\theta|\mathbf{d}_t)}[\omega(\mathbf{x},\theta)]| \cdot \prod_{j=1}^{i-1} \psi_{\mathbf{x}_j^\star}(\mathbf{x}). \tag{20}$$

Similarly, the batch acquisition function for the vanilla approach takes the form

$$\mathbf{x}_i^\star = \underset{\mathbf{x}\in\Omega}{\arg\max}\ \mathbb{D}_{\pi_t(\theta|\mathbf{d}_t)}[u(\mathbf{x},\theta)] \cdot \prod_{j=1}^{i-1} \psi_{\mathbf{x}_j^\star}(\mathbf{x}). \tag{21}$$

A reasonable choice of penalisation functions would be the Gaussian

$$\psi_{\mathbf{x}'}(\mathbf{x}) = 1 - \exp\left(-\frac{1}{2}\frac{\|\mathbf{x}-\mathbf{x}'\|_2^2}{l_\psi}\right) \tag{22}$$

where l_ψ controls the dispersion of the function and $\|\cdot\|_2$ is the L^2-norm. Using such a penalisation function, the acquisition function would be exactly zero at previous sampling points from the current batch, and smoothly rebound to Eq. (19) or Eq. (12) as the distance to previous sampling points increases.

(4) As mentioned earlier, we formulate our method in the context of steady state groundwater flow in a confined aquifer. While this is the most common approach to groundwater flow modelling, it is, naturally, not exhaustive. For a detailed analysis of the adjoint state equations for transient groundwater flow, we refer the to e.g. Sun [35] and Lu and Vesselinov [36]. The unconfined case is considerably more complex, since the constitutive equations are nonlinear. While unconfined groundwater flow can, under some assumptions, be reasonably approximated by the constitutive equations for confined flow [37], this is not always the case. For a derivation and analysis of the adjoint equations pertaining to unconfined and coupled aquifers, we refer to e.g. Sun [35] and Neupauer and Griebling [38].

(5) Note that the constitutive and adjoint equations are discretised using FEM in the above section. We restrict ourselves to this method for brevity, but remark that the proposed acquisition functions (Eqs. (12), (19), (20) and (21)) are valid for any discretisation scheme. Also note that if piecewise

linear shape functions are employed to approximate $u(\mathbf{x})$, the maxima of the acquisition functions will occur at finite element nodes.

3 Example

In this section, we demonstrate the vanilla and dual-weighted approach in the context of a synthetic groundwater flow example. We first outline the model setup, including the geological model and finite element representation. We then explain the particular methodology for this example in detail. Finally, we present the results.

3.1 Model Setup

We model the hydraulic conductivity as a log-Gaussian Random Field with a Matern 3/2 covariance kernel:

$$C(\mathbf{x}, \mathbf{y}) = \left(1 + \sqrt{3}\frac{\|\mathbf{x} - \mathbf{y}\|_2}{l}\right) \exp\left(-\sqrt{3}\frac{\|\mathbf{x} - \mathbf{y}\|_2}{l}\right) \tag{23}$$

where l is the length scale [39] and $\|\cdot\|_2$ is the L^2-norm. The resulting random field is expanded in an orthogonal eigenbasis with N_{KL} Karhunen–Loève (KL) eigenmodes. To this end, we construct a matrix of covariances between each pair of finite element nodes $\mathbf{C} \in \mathbb{R}^{M \times M}$ according to Eq. (23), so that $C_{ij} = C(\mathbf{x}_i, \mathbf{x}_j)$. This covariance matrix $\mathbf{C}$ is decomposed into the N_{KL} largest eigenvalues $\{\lambda_i\}_{i=1}^{N_{\text{KL}}}$ and eigenvectors $\{\psi_i\}_{i=1}^{N_{\text{KL}}}$. The nodal conductivities $\mathbf{k} := [k_1, k_2, \dots, k_M]$ are then given by

$$\log \mathbf{k} = \mu + \sigma \mathbf{\Psi} \mathbf{\Lambda}^{\frac{1}{2}} \theta \tag{24}$$

with $\mathbf{\Lambda} = \text{diag}([\lambda_1, \lambda_2, \dots, \lambda_{N_{\text{KL}}}])$ and $\mathbf{\Psi} = [\psi_1, \psi_2, \dots, \psi_{N_{\text{KL}}}]$. The vector $\mu = \mu\mathbf{1}$ is the mean of the log-conductivity, σ is the standard deviation of the log-conductivity, and $\theta \sim \mathcal{N}(0, \mathbb{I}_{N_{\text{KL}}})$ [40]. When defined in this way, the associated Bayesian inverse problem involves exploring $\pi(\theta|\mathbf{d})$, i.e. the posterior distribution of hydraulic conductivity parameters θ given measurements $\mathbf{d}$, where the aforementioned normal distribution constitutes the prior distribution of parameters: $\pi_{\text{p}}(\theta) = \mathcal{N}(0, \mathbb{I}_{N_{\text{KL}}})$.

We used three different models for the experiments (Fig. 3), one *data-generating* model representing the ground truth, a *fine* forward model representing the fully resolved forward model $\mathcal{F}$ in the Bayesian inverse problem (see Eq. (1)), and a *coarse* forward model, corresponding to the reduced order forward model in the Delayed Acceptance MCMC sampler $\hat{\mathcal{F}}$, as described in e.g. Christen and Fox [25], Liu [24], Cui et al. [28], Lykkegaard et al. [26, 27]. Note that using the dual-weighted approach described herein does not require a Delayed Acceptance MCMC sampler. Any method capable of producing Monte Carlo samples from the posterior will do.

The experiments were performed on a rectangular domain $\Omega = [0, 2] \times [0, 1]$ meshed using a structured triangular grid with $M_{fine} = 1326$ degrees of freedom for the data-generating model and the fine forward model, and $M_{coarse} = 703$ degrees of freedom for the coarse forward model. For the data-generating model, the log-Gaussian random conductivity was truncated at $N_{\text{KL}} = 256$

KL eigenmodes, while for the fine and coarse models it was truncated at $N_{\text{KL}} = 128$. Hence the dimensionality of the inverse problem in these experiments was 128, which is very high and a challenging problem for any MCMC algorithm. Moreover, we set $l = 0.1$, $\mu = -2$ and $\sigma = 1.0$ for every model. This resulted in strongly anisotropic conductivity fields with log-conductivities broadly between -5 and 1 (Fig. 3a).

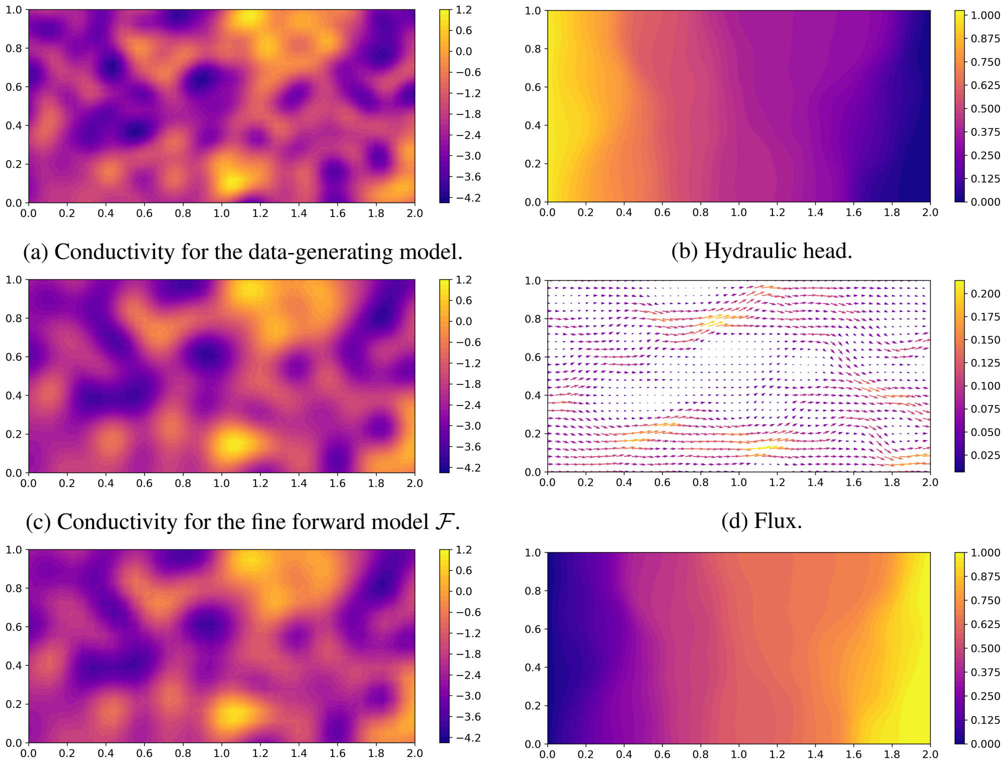
We imposed fixed head Dirichlet boundary conditions of 1 and 0 on the left and right boundaries, respectively, and no-flow Neumann conditions on the remaining top and bottom boundaries. We set the right hand side of Eq. (7) to $g(\mathbf{x}) = 0$. We chose flux across the right boundary Γ_r as our quantity of interest $\mathcal{Q}$, corresponding to the following functional (as in equation (14)):

$$\mathcal{Q}(u) = \int_{\Gamma_r} [-k(\mathbf{x}, \theta) \cdot \nabla u(\mathbf{x})] \cdot \mathbf{n} \, ds \tag{25}$$

and the associated adjoint state equation shown in (15) with $\Gamma' = \Gamma_r$. Figure 3f shows an example of the influence function generated by this adjoint state equation. The left column of Fig. 3 shows the conductivity associated with a random draw from the prior $\pi_{\text{p}}(\theta)$, for the data-generating model, the fine model, and the coarse model, respectively. The right column of Fig. 3 shows the corresponding hydraulic head, flux and influence function for the data-generating model.

3.1.1 Methodology

Using the above setup, we completed a total of $n = 30$ independent numerical experiments to demonstrate the feasibility of the dual-weighted approach. We chose the standard deviation of the L^2-norm of the flux $S(\|\mathbf{q}(\mathbf{x})\|_2)$ as the general measure of uncertainty in the model. For each independent experiment, the following experimental procedure was observed: (1) The hydraulic conductivity for the data-generating model was initialised with a random draw from the prior, and the primary problem was solved. (2) Eight observation wells were placed randomly on the domain by Latin Hypercube sampling [41] (see Fig. 4). (3) For each observation well $\mathbf{x}_i$, the hydraulic head $u(\mathbf{x}_i)$ and the norm of the flux $\|\mathbf{q}(\mathbf{x}_i)\|_2$ were computed. These head and flux observations were contaminated with white noise from $\epsilon_u \sim \mathcal{N}(0, 0.01^2)$ and $\epsilon_{\|q\|_2} \sim \mathcal{N}(0, 0.001^2)$, respectively. (4) Delayed Acceptance MCMC sampling was completed with 2 independent samplers each drawing $N = 25000$ fine samples with a subsampling length of 5 (see e.g. Lykkegaard et al. [26, 27]), and a burn-in of $N_{burn} = 5000$ was discarded. This resulted in a total number of MCMC samples of $N^\dagger = 40000$ for each experiment. (5) The standard deviation of the L^2-norm of the flux $S(\|\mathbf{q}(\mathbf{x})\|_2)$ and the mean of the influence function $\bar{\omega}(\mathbf{x})$ were computed at the finite element nodes and interpolated to the entire domain using the finite element shape functions, and eight new observation wells were placed according to the batch vanilla and dual-weighted acquisition functions, see Eq. (21) and Eq. (20). Figure 4 shows the vanilla and dual weighted acquisition functions for one sample of the $n = 30$ models. As expected, the weighting function $\bar{\omega}(\mathbf{x})$ prioritised observation wells closer to the boundary of the quantity of interest. (6) Data were extracted from the four new observation wells as in step (3) and appended to the data vector. (7) Delayed Acceptance MCMC sampling was repeated, using the new data vectors for both the vanilla and dual-weighted approaches.



(a) Conductivity for the data-generating model.

(b) Hydraulic head.

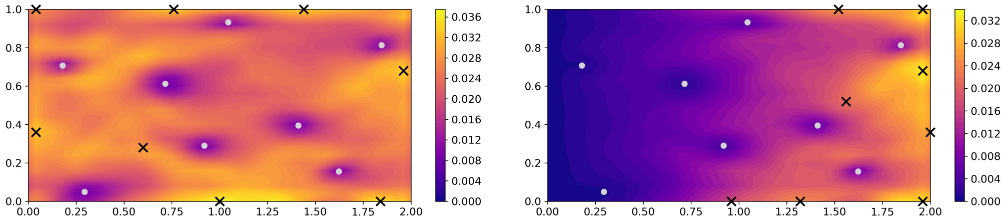
(c) Conductivity for the fine forward model $\mathcal{F}$.

(d) Flux.

(e) Conductivity for the coarse forward model $\hat{\mathcal{F}}$.

(f) Influence function.

Figure 3: A random realisation from the prior $\pi_{\mathrm{p}}(\theta)$, with the corresponding primary and adjoint solutions. The left column shows the conductivity for the data-generating model (a), the fine forward model (c) and the coarse forward model (e) respectively. The right column shows the hydraulic head (b), the flux (d), and the influence function (f), respectively.



(a) Vanilla acquisition $S(\|\mathbf{q}(\mathbf{x})\|_2)$.

(b) Dual-weighted acquisition $S(\|\mathbf{q}(\mathbf{x})\|_2) \cdot \bar{\omega}(\mathbf{x})$.

Figure 4: Acquisition functions of the vanilla and dual-weighted approaches for one sample of the $n = 30$ models. The white dots show the initial datapoints, while the black crosses show the new datapoints suggested by each acquisition function.

For each experiment and each posterior distribution (initial, vanilla, and dual-weighted) with each $N^\dagger = 40000$ posterior samples, we computed the mean squared error (MSE) and variance of the predicted quantity of interest $\{\mathcal{Q}^{(i)}\}_{i=1}^{N^\dagger}$ compared to the true value $\mathcal{Q}_{true}$. The MSE of the predicted value of the quantity of interest $\mathcal{Q}^{(i)}$ with respect to the true value $\mathcal{Q}_{true}$ was computed as

$$\text{MSE} = \frac{1}{N^\dagger}\sum_{i=1}^{N^\dagger}(\mathcal{Q}_{true} - \mathcal{Q}^{(i)})^2 \tag{26}$$

Similarly, the sample variance of $\mathcal{Q}$ for each experiment was computed as:

$$s^2 = \frac{1}{N^\dagger - 1}\sum_{i=1}^{N^\dagger}(\mathcal{Q}^{(i)} - \bar{\mathcal{Q}})^2 \tag{27}$$

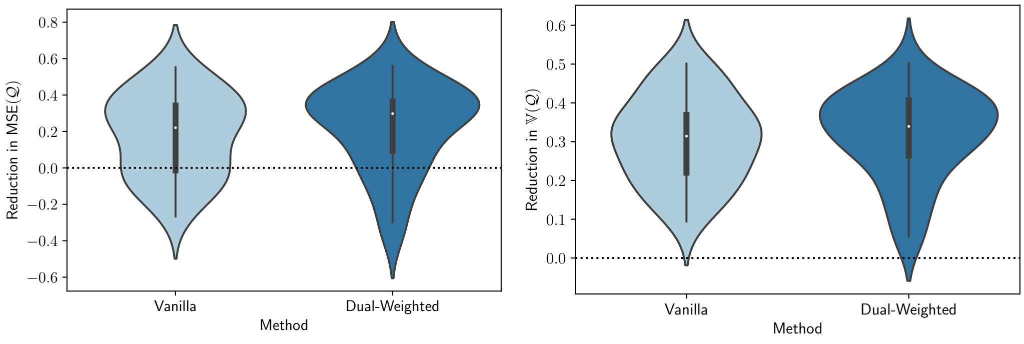
Finally, we constructed Gaussian kernel posterior density estimates $\hat{f}_{\pi(\theta|\mathbf{d})}(\mathcal{Q})$ from the posterior samples from each experiment $\{\mathcal{Q}^{(i)}\}_{i=1}^{N^\dagger}$, and computed the kernel density of the true value $\mathcal{Q}_{true}$ with respect to this density estimate. Kernel density estimates were computed using `SciPy` [42] with automatic bandwidth determination [43].

3.1.2 Results

We compared the MSE, variance, and kernel density of both the vanilla and dual-weighted posterior samples with the corresponding values for the initial posterior samples for all $n = 30$ experiments.

With respect to the MSE, the vanilla approach yielded a median reduction of 22%, while the dual–weighted approach yielded a median reduction of 30% (Fig. 5a). This demonstrates that both acquisition strategies approach the true value when we add more datapoints, but that the dual-weighted approach is more efficient. With respect to the variance of the quantity of interest, the vanilla approach yielded a median reduction of 31%, while the dual–weighted approach yielded a median reduction of 34% (Fig. 5b). This shows that for both acquisition strategies the posterior distribution contracts as more data is added, and that the two approaches differ less with respect to this feature. However, this metric shows only that the posterior contracts, and not if it moves closer to the true value. Finally, we computed the posterior densities of the true quantity of interest with respect to kernel posterior density estimates $\hat{f}_{\pi(\theta|\mathbf{d})}(\mathcal{Q})$ for each experiment. Here, the vanilla approach yielded a median improvement of 12%, while the dual–weighted approach yielded a median improvement of 17%. Since the prediction variance of the quantity of interest reduced in every experiment (Fig. 5b), this again shows that the posterior distribution moves closer to the true value as more data is added, but that the dual-weighted approach is better.

We note that in neither method was capable of improving the posterior estimate of the quantity of interest for every experiment. Hence, in $8/30$ vanilla experiments and $5/30$ dual-weighted experiments, adding additional wells resulted in a worse posterior MSE than the initial one. This is not surprising since we are dealing with a very ill-posed inverse problem, and any new datapoint may



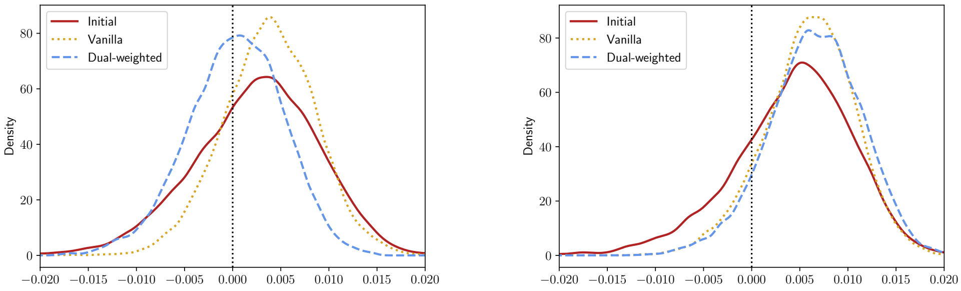
(a) Reduction in MSE($\mathcal{Q}$)

(b) Reduction in $\mathbb{V}(\mathcal{Q})$

Figure 5: Kernel densities of the sample error of the quantity of interest $\varepsilon^{(i)} = Q_{true} - Q^{(i)}$ for the initial, vanilla and dual-weighted posteriors for two samples of the $n = 30$ experiments.

reinforce the initial bias rather than reduce it. While both approaches occasionally failed to improve the posterior estimate, the dual-weighted approach performed better than the vanilla approach.

We computed the Gaussian kernel density estimates of the error $\varepsilon^{(i)} = \mathcal{Q}_{true} - \mathcal{Q}^{(i)}$ for two samples of the $n = 30$ experiments. The left panel shows a typical example, where the vanilla approach resulted in a moderate improvement while the dual-weighted approach yielded a more dramatic improvement. The right panel shows an example where both the dual-weighted and vanilla approaches failed to produce any improvement.



(a)

(b)

Figure 6: Kernel densities of the sample error of the quantity of interest $\varepsilon^{(i)} = Q_{true} - Q^{(i)}$ for the initial, vanilla and dual-weighted posteriors for two samples of the $n = 30$ experiments.

4 Discussion

In this paper, we have proposed a novel approach to the problem of optimally choosing the next location for a monitoring well, given existing data and some quantity of interest (QoI). The proposed methodology exploits the solution of an adjoint problem to weigh such an acquisition function according to the expected influence on the QoI. Numerical experiments have demonstrated that the approach works for our model problem. We emphasize that the problem is intrinsically probabilistic, and hence subject to uncertainty. We have demonstrated that the approach works *on average* for our model problem, but there were certain experiments, where the dual-weighted acquisition strategy did not approach the true QoI (see e.g. Fig. 6b). As the number of wells approach infinity, the posterior distribution will certainly approach the true value, but for any one new observation well, there are no such guarantees. In a sense, the dual-weighted approach merely increases the chance of improving the posterior distribution of the QoI.

While we formulated and demonstrated the approach in the context of a groundwater surveying problem, the method could be applicable to other areas of science and engineering, where measurements are expensive. The most obvious parallel application is petroleum engineering, where there are similarities both in terms of the constituent equations and the mode of sampling, but the method could be adapted with little effort to any inverse problem where establishing sensors is expensive. We note, however, that the dual problem in our case was unusually simple, since the groundwater flow equation is self-adjoint. Clearly, the dual-weighted approach can only be used as-written for QoIs, where an adjoint problem can be formulated and solved directly. For more complicated QoIs, an alternative approach would be to perturb the posterior mean or mode to approximate the influence function. Using such an approach would yield $\omega(\mathbf{x}, \mathbb{E}[\theta])$ rather than $\mathbb{E}[\omega(\mathbf{x}, \theta)]$ as a weighting function.

A bottleneck of our approach is that the MCMC sampler is rerun after each (batch) data acquisition. Running MCMC for expensive forward models is notoriously computationally demanding, and while we employ various tricks to reduce the cost (such as Delayed Acceptance and proposal adaptivity), this is not the most elegant approach. One way to significantly alleviate the cost of subsequent posterior distributions would be to employ a particle filter to sequentially reweigh MCMC samples according to the new data [44]. This sequential approach was investigated in this study but it did not work well, mainly because of very high sample degeneracy. When the variance of the solution, as in our case, is relatively high at unobserved locations, only few posterior samples fit the new observations well, with the mentioned sample degeneracy as a result. Moreover, we found that the dispersion measures in Eqs. (12), (19), (20) and (21) where highly sensitive to this sample degeneracy. This challenge could be alleviated by drawing more posterior samples for the initial MCMC, but that would only offset the cost. We remark that this approach might work better for lower-dimensional problems than the one investigated in this study. We highlight this problem as a potential target for future research.

The methodology was demonstrated empirically in the context of a synthetic groundwater flow example. This gives rise to at least three additional interesting directions of future research. First, showing theoretically that the distribution of the quantity of interest does indeed converge faster to the true value when using the dual-weighted approach, and examining the mechanisms that govern this process in detail. Second, testing the method in practice in the context of an actual groundwater survey. While testing the method in practice would certainly expose limitations and complications that were not identified in this study, it would be difficult to validate the method further in this fashion, since the true value of the QoI is rarely known in reality. This may be overcome by testing the method under controlled (laboratory) conditions. Third, generalising the dual-weighted approach to a wider range of PDE problems with different constituent equations and QoIs.

Acknowledgements

The MCMC code used for Delayed Acceptance sampling can be found at https://github.com/mikkelbue/tinyDA, and additional code will be made available in the Open Research Exeter data repository upon publication at https://ore.exeter.ac.uk/repository/. ML was funded as part of the Water Informatics Science and Engineering Centre for Doctoral Training (WISE CDT) under a grant from the Engineering and Physical Sciences Research Council (EPSRC), grant number EP/L016214/1. TD was funded by a Turing AI Fellowship (2TAFFP\100007). The authors would like to thank Robert Scheichl and Karina Koval for advice with regards to the formulation of the adjoint state equation. The authors have no competing interests.

Appendix A Adjoint State Equations

A.1 Domain Integral as Objective Function

Given an objective function defined as an integral over the entire domain

$$\mathcal{Q} = \int_\Omega f \, dx \tag{28}$$

Sykes et al. [33, Eq. (15)] write the derivative of $\mathcal{Q}$ with respect to some parameter α as

$$\begin{aligned}\frac{d\mathcal{Q}}{d\alpha} = &\int_\Omega \left[\frac{\partial f}{\partial \alpha} + \psi\left(\frac{\partial f}{\partial u} + \nabla \cdot k\nabla\omega\right) + \omega\frac{\partial g}{\partial \alpha} - \nabla\omega \cdot \frac{\partial k}{\partial \alpha}\nabla u\right] dx \\ &+ \int_\Gamma \left[\psi(k\nabla\omega)\cdot \mathbf{n} + \omega\frac{\partial q_N}{\partial \alpha}\right] ds\end{aligned} \tag{29}$$

To eliminate the unknown state sensitivities $\psi = \frac{\partial u}{\partial \alpha}$ they solve

$$\nabla \cdot k\nabla\omega + \frac{\partial f}{\partial u} = 0 \tag{30}$$

with boundary conditions $\omega_D = 0$ on Γ_D and $q_N^\omega = k\nabla\omega \cdot \mathbf{n} = 0$ on Γ_N.

A.2 Boundary Integral as Objective Function

The problem addressed in this paper involves an objective function defined on a fixed-head boundary Γ':

$$\mathcal{Q} = \int_{\Gamma'} f \, ds \quad \text{with} \quad f = q = -k\nabla u \cdot \mathbf{n}^+ \tag{31}$$

Where $\mathbf{n}^+$ is the outward normal. Hence, the derivative of the objective function instead takes the form

$$\begin{aligned}\frac{dQ}{d\alpha} &= \int_\Omega \left[\psi\left(\nabla \cdot k\nabla\omega\right) + \omega\frac{\partial g}{\partial \alpha} - \nabla\omega \cdot \frac{\partial k}{\partial \alpha}\nabla u\right] dx \\ &+ \int_\Gamma \left[\psi(k\nabla\omega)\cdot \mathbf{n}^- + \omega\left(\frac{\partial \mathbf{q}}{\partial \alpha}\cdot \mathbf{n}^- + \frac{\partial \mathbf{q}}{\partial u}\psi \cdot \mathbf{n}^-\right)\right] ds \\ &+ \int_{\Gamma'} \left[\frac{\partial f}{\partial \alpha} + \frac{\partial f}{\partial u}\psi\right] ds\end{aligned} \tag{32}$$

where $\mathbf{n}^-$ is the inward normal [33] and

$$\frac{\partial \mathbf{q}}{\partial \alpha}\cdot \mathbf{n}^- + \frac{\partial \mathbf{q}}{\partial u}\psi \cdot \mathbf{n}^- = \frac{\partial q_N}{\partial \alpha} \quad \text{on} \quad \Gamma_N. \tag{33}$$

To eliminate the unknown state sensitivities ψ, we now solve

$$\nabla \cdot k\nabla\omega = 0 \tag{34}$$

with boundary conditions $\omega_D = 0$ on $\Gamma_D \setminus \Gamma'$ and $q_N^\omega = k\nabla\omega \cdot \mathbf{n}^- = 0$ on Γ_N. For the remaining boundary Γ', we impose

$$\frac{\partial f}{\partial u} + \omega\frac{\partial \mathbf{q}}{\partial u}\cdot \mathbf{n}^- = 0. \tag{35}$$

Since on Γ' we have

$$-\frac{\partial \mathbf{q}}{\partial u}\cdot \mathbf{n}^- = \frac{\partial f}{\partial u} \tag{36}$$

we can substitute (36) into (35) to get

$$\frac{\partial f}{\partial u} - \omega\frac{\partial f}{\partial u} = 0 \quad \text{on} \quad \Gamma' \tag{37}$$

and so the operative boundary condition on Γ' is $\omega_{\Gamma'} = 1$.

References

[1] Mary P. Anderson, William W. Woessner, and R. J. Hunt. *Applied groundwater modeling: simulation of flow and advective transport*. Academic Press, London ; San Diego, CA, second edition edition, 2015. ISBN 978-0-12-058103-0. OCLC: ocn921253555.

[2] M.H. Loke, J.E. Chambers, D.F. Rucker, O. Kuras, and P.B. Wilkinson. Recent developments in the direct-current geoelectrical imaging method. *Journal of Applied Geophysics*, 95:135–156, August 2013. ISSN 09269851. doi: 10.1016/j.jappgeo.2013.02.017. URL https://linkinghub.elsevier.com/retrieve/pii/S0926985113000499.

[3] T. Saey, M. Van Meirvenne, P. De Smedt, B. Stichelbaut, S. Delefortrie, E. Baldwin, and V. Gaffney. Combining EMI and GPR for non-invasive soil sensing at the Stonehenge World Heritage Site: the reconstruction of a WW1 practice trench: Reconstructing a practice trench using GPR and EMI. *European Journal of Soil Science*, 66(1):166–178, January 2015. ISSN

13510754. doi: 10.1111/ejss.12177. URL https://onlinelibrary.wiley.com/doi/10.1111/ejss.12177.

[4] Esben Auken, Tue Boesen, and Anders V. Christiansen. A Review of Airborne Electromagnetic Methods With Focus on Geotechnical and Hydrological Applications From 2007 to 2017. volume 58 of *Advances in Geophysics*, pages 47–93. Elsevier, 2017. doi: https://doi.org/10.1016/bs.agph.2017.10.002. URL https://www.sciencedirect.com/science/article/pii/S006526871730002X. ISSN: 0065-2687.

[5] Esben Auken, Nikolaj Foged, Jakob Juul Larsen, Knud Valdemar Trøllund Lassen, Pradip Kumar Maurya, Søren Møller Dath, and Tore Tolstrup Eiskjær. tTEM — A towed transient electromagnetic system for detailed 3D imaging of the top 70 m of the subsurface. *GEOPHYSICS*, 84(1):E13–E22, January 2019. ISSN 0016-8033, 1942-2156. doi: 10.1190/geo2018-0355.1. URL https://library.seg.org/doi/10.1190/geo2018-0355.1.

[6] Friedrich Pukelsheim. *Optimal design of experiments*. Number 50 in Classics in applied mathematics. SIAM/Society for Industrial and Applied Mathematics, Philadelphia, classic ed edition, 2006. ISBN 978-0-89871-604-7. OCLC: ocm62742628.

[7] Raymond H. Myers, Douglas C. Montgomery, and Christine M. Anderson-Cook. *Response surface methodology: process and product optimization using designed experiments*. Wiley series in probability and statistics. Wiley, Hoboken, New Jersey, fourth edition edition, 2016. ISBN 978-1-118-91601-8.

[8] D. R. Cox and N. Reid. *The theory of the design of experiments*. Number 86 in Monographs on statistics and applied probability. Chapman & Hall/CRC, Boca Raton, 2000. ISBN 978-1-58488-195-7.

[9] D. V. Lindley. On a Measure of the Information Provided by an Experiment. *The Annals of Mathematical Statistics*, 27(4):986–1005, December 1956. ISSN 0003-4851. doi: 10.1214/aoms/1177728069. URL http://projecteuclid.org/euclid.aoms/1177728069.

[10] M. C. Shewry and H. P. Wynn. Maximum entropy sampling. *Journal of Applied Statistics*, 14 (2):165–170, January 1987. ISSN 0266-4763, 1360-0532. doi: 10.1080/02664768700000020. URL https://www.tandfonline.com/doi/full/10.1080/02664768700000020.

[11] Hossein Mohammadi, Peter Challenor, Daniel Williamson, and Marc Goodfellow. Cross-validation based adaptive sampling for gaussian process models. *arXiv:2005.01814 [stat]*, 2021. URL https://arxiv.org/abs/2005.01814. arXiv: 2005.01814.

[12] Andreas Krause, Ajit Singh, and Carlos Guestrin. Near-Optimal Sensor Placements in Gaussian Processes: Theory, Efficient Algorithms and Empirical Studies. *Journal of Machine Learning Research*, 9(8):235–284, 2008. URL http://jmlr.org/papers/v9/krause08a.html.

[13] Joakim Beck and Serge Guillas. Sequential Design with Mutual Information for Computer Experiments (MICE): Emulation of a Tsunami Model. *SIAM/ASA Journal on Uncertainty Quantification*, 4(1):739–766, January 2016. ISSN 2166-2525. doi: 10.1137/140989613. URL http://epubs.siam.org/doi/10.1137/140989613.

[14] Jonas Močkus. *Bayesian Approach to Global Optimization: Theory and Applications*. Springer Netherlands, Dordrecht, 1989. ISBN 978-94-009-0909-0. URL https://doi.org/10.1007/978-94-009-0909-0. OCLC: 851374758.

[15] Peter I. Frazier. A Tutorial on Bayesian Optimization. *arXiv:1807.02811 [cs, math, stat]*, July 2018. URL http://arxiv.org/abs/1807.02811. arXiv: 1807.02811.

[16] Artur Souza, Luigi Nardi, Leonardo B. Oliveira, Kunle Olukotun, Marius Lindauer, and Frank Hutter. Bayesian Optimization with a Prior for the Optimum. In Nuria Oliver, Fernando Pérez-Cruz, Stefan Kramer, Jesse Read, and Jose A. Lozano, editors, *Machine Learning and Knowledge Discovery in Databases. Research Track*, pages 265–296, Cham, 2021. Springer International Publishing. ISBN 978-3-030-86523-8.

[17] Javier Gonzalez, Zhenwen Dai, Philipp Hennig, and Neil Lawrence. Batch bayesian optimization via local penalization. In Arthur Gretton and Christian C. Robert, editors, *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, volume 51 of *Proceedings of Machine Learning Research*, pages 648–657, Cadiz, Spain, 09–11 May 2016. PMLR. URL https://proceedings.mlr.press/v51/gonzalez16a.html.

[18] S. Prudhomme and J.T. Oden. On goal-oriented error estimation for elliptic problems: application to the control of pointwise errors. *Computer Methods in Applied Mechanics and Engineering*, 176(1-4):313–331, July 1999. ISSN 00457825. doi: 10.1016/S0045-7825(98)00343-0. URL https://linkinghub.elsevier.com/retrieve/pii/S0045782598003430.

[19] J.T. Oden and S. Prudhomme. Goal-oriented error estimation and adaptivity for the finite element method. *Computers & Mathematics with Applications*, 41(5-6):735–756, March 2001. ISSN 08981221. doi: 10.1016/S0898-1221(00)00317-5. URL https://linkinghub.elsevier.com/retrieve/pii/S0898122100003175.

[20] Ahmed Attia, Alen Alexanderian, and Arvind K Saibaba. Goal-oriented optimal design of experiments for large-scale Bayesian linear inverse problems. *Inverse Problems*, 34(9):095009, September 2018. ISSN 0266-5611, 1361-6420. doi: 10.1088/1361-6420/aad210. URL https://iopscience.iop.org/article/10.1088/1361-6420/aad210.

[21] 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(6):1087–1092, June 1953. ISSN 0021-9606, 1089-7690. doi: 10.1063/1.1699114. URL http://aip.scitation.org/doi/10.1063/1.1699114.

[22] W K Hastings. Monte Carlo sampling methods using Markov chains and their applications. *Biometrika*, page 13, 1970.

[23] Andrew Gelman, editor. *Bayesian data analysis*. Texts in statistical science. Chapman & Hall/CRC, Boca Raton, Fla, 2nd ed edition, 2004. ISBN 978-1-58488-388-3.

[24] Jun S. Liu. *Monte Carlo Strategies in Scientific Computing*. Springer Series in Statistics. Springer New York, New York, NY, 2004. ISBN 978-0-387-76369-9 978-0-387-

76371-2. doi: 10.1007/978-0-387-76371-2. URL http://link.springer.com/10.1007/978-0-387-76371-2.

[25] J. Andrés Christen and Colin Fox. Markov chain Monte Carlo Using an Approximation. *Journal of Computational and Graphical Statistics*, 14(4):795–810, December 2005. ISSN 1061-8600, 1537-2715. doi: 10.1198/106186005X76983. URL http://www.tandfonline.com/doi/abs/10.1198/106186005X76983.

[26] Mikkel B Lykkegaard, Grigorios Mingas, Robert Scheichl, Colin Fox, and Tim J Dodwell. Multilevel Delayed Acceptance MCMC with an Adaptive Error Model in PyMC3. In *Machine Learning for Engineering Modeling, Simulation, and Design Workshop at Neural Information Processing Systems (NeurIPS) 2020*, 2020. https://ml4eng.github.io/camera_readys/04.pdf.

[27] Mikkel B. Lykkegaard, Tim J. Dodwell, Colin Fox, Grigorios Mingas, and Robert Scheichl. Multilevel Delayed Acceptance MCMC, 2022.

[28] Tiangang Cui, Colin Fox, and Michael J. O'Sullivan. A posteriori stochastic correction of reduced models in delayed acceptance MCMC, with application to multiphase subsurface inverse problems. *arXiv:1809.03176 [stat]*, September 2018. URL http://arxiv.org/abs/1809.03176. arXiv: 1809.03176.

[29] Heikki Haario, Eero Saksman, and Johanna Tamminen. An Adaptive Metropolis Algorithm. *Bernoulli*, 7(2):223, April 2001. ISSN 13507265. doi: 10.2307/3318737. URL https://www.jstor.org/stable/3318737?origin=crossref.

[30] Hans-Jörg G. Diersch. *FEFLOW: Finite Element Modeling of Flow, Mass and Heat Transport in Porous and Fractured Media*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014. ISBN 978-3-642-38738-8 978-3-642-38739-5. doi: 10.1007/978-3-642-38739-5. URL http://link.springer.com/10.1007/978-3-642-38739-5.

[31] Hans Petter Langtangen and Anders Logg. *Solving PDEs in Python: The FEniCS Tutorial I*. Number 3 in Simula SpringerBriefs on Computing. Springer International Publishing : Imprint: Springer, Cham, 1st ed. 2016 edition, 2016. ISBN 978-3-319-52462-7. doi: 10.1007/978-3-319-52462-7.

[32] R.-E. Plessix. A review of the adjoint-state method for computing the gradient of a functional with geophysical applications. *Geophysical Journal International*, 167(2):495–503, November 2006. ISSN 0956540X, 1365246X. doi: 10.1111/j.1365-246X.2006.02978.x. URL https://academic.oup.com/gji/article-lookup/doi/10.1111/j.1365-246X.2006.02978.x.

[33] J. F. Sykes, J. L. Wilson, and R. W. Andrews. Sensitivity Analysis for Steady State Groundwater Flow Using Adjoint Operators. *Water Resources Research*, 21(3):359–371, March 1985. ISSN 00431397. doi: 10.1029/WR021i003p00359. URL http://doi.wiley.com/10.1029/WR021i003p00359.

[34] John L. Wilson and Douglas E. Metcalfe. Illustration and Verification of Adjoint Sensitivity Theory for Steady State Groundwater Flow. *Water Resources Research*, 21(11):1602–1610,

November 1985. ISSN 00431397. doi: 10.1029/WR021i011p01602. URL http://doi.wiley.com/10.1029/WR021i011p01602.

[35] Ne-Zheng Sun. *Inverse Problems in Groundwater Modeling*, volume 6 of *Theory and Applications of Transport in Porous Media*. Springer Netherlands, Dordrecht, 1999. ISBN 978-90-481-4435-8 978-94-017-1970-4. doi: 10.1007/978-94-017-1970-4. URL http://link.springer.com/10.1007/978-94-017-1970-4.

[36] Zhiming Lu and Velimir V. Vesselinov. Analytical sensitivity analysis of transient groundwater flow in a bounded model domain using the adjoint method. *Water Resources Research*, 51(7): 5060–5080, July 2015. ISSN 0043-1397, 1944-7973. doi: 10.1002/2014WR016819. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/2014WR016819.

[37] Herbert Wang and Mary P. Anderson. *Introduction to groundwater modeling: finite difference and finite element methods*. Series of books in geology. W.H. Freeman, San Francisco, 1982. ISBN 978-0-7167-1303-6.

[38] Roseanna M. Neupauer and Scott A. Griebling. Adjoint Simulation of Stream Depletion Due to Aquifer Pumping. *Ground Water*, 50(5):746–753, September 2012. ISSN 0017467X. doi: 10.1111/j.1745-6584.2011.00901.x. URL https://onlinelibrary.wiley.com/doi/10.1111/j.1745-6584.2011.00901.x.

[39] Carl Edward Rasmussen and Christopher K. I. Williams. *Gaussian processes for machine learning*. Adaptive computation and machine learning. MIT Press, Cambridge, Mass, 2006. ISBN 978-0-262-18253-9. OCLC: ocm61285753.

[40] T. J. Dodwell, C. Ketelsen, R. Scheichl, and A. L. Teckentrup. A Hierarchical Multilevel Markov Chain Monte Carlo Algorithm with Applications to Uncertainty Quantification in Subsurface Flow. *SIAM/ASA Journal on Uncertainty Quantification*, 3(1):1075–1108, January 2015. ISSN 2166-2525. doi: 10.1137/130915005. URL http://epubs.siam.org/doi/10.1137/130915005.

[41] M. D. McKay, R. J. Beckman, and W. J. Conover. A comparison of three methods for selecting values of input variables in the analysis of output from a computer code. *Technometrics*, 21(2): 239–245, 1979. ISSN 00401706. URL http://www.jstor.org/stable/1268522.

[42] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020. doi: 10.1038/s41592-019-0686-2.

[43] David W. Scott. *Multivariate density estimation: theory, practice, and visualization*. Wiley series in probability and mathematical statistics. Wiley, New York, 1992. ISBN 978-0-471-54770-9.

[44] N. Chopin. A sequential particle filter method for static models. *Biometrika*, 89(3):539–552, August 2002. ISSN 0006-3444, 1464-3510. doi: 10.1093/biomet/89.3.539. URL `https://academic.oup.com/biomet/article-lookup/doi/10.1093/biomet/89.3.539`.

7. Conclusions

In this thesis, I have presented a novel MCMC method, namely the Multilevel Delayed Acceptance (MLDA) algorithm, which can be considered as a biaxial extension and generalisation of the Delayed Acceptance algorithm (Christen and Fox, 2005). MLDA is particularly suitable for large-scale, high-dimensional Bayesian inverse problems, where the cost of the fully resolved forward model may be detrimental to the sampling efficiency of standard MCMC methods, but it can be employed to sample from any probability distribution with a (hierarchy of) coarse approximation(s). Additionally, the MLDA algorithm can be adaptively improved using an approximation error model and it can be exploited for variance reduction in a fashion similar to Multilevel Monte Carlo (Giles, 2008a). The algorithm was employed in the context of two novel developments in the field of hydrogeological inverse problems of a more practical nature. First, the development of Deep Neural Networks (DNNs) that can be utilised as fast coarse models when quantifying the geological uncertainty of groundwater flow problems. Second, the development of a novel approach to adaptive optimal design of groundwater surveying, which directly utilises Monte Carlo estimates from MLDA to suggest the next monitoring well.

7.1 Multilevel Delayed Acceptance MCMC

7.1.1 Software Implementation

In Chapter 3 and 4 we presented the MLDA algorithm, which has been implemented in the popular open-source probabilistic programming framework PyMC. While the theoretical framework underpinning MLDA is relatively simple, nestling the algorithm within PyMC was a less than straightforward task. Being designed mainly to provide a high-level platform for the NUTS sampler in `Python`, the existing implementation of the elementary Metropolis sampler remains fairly rudimentary and is not optimised for expensive statistical models. Moreover, since `PyMC` makes heavy use of the high-performance tensor library `Aesara` (previously `Theano`) to compute gradients, defining custom probability distributions appropriate to the Bayesian inverse problems mainly targeted by MLDA is not straightforward. If the built-in distributions are sufficient for a given problem, our implementation of MLDA in `PyMC` provides practitioners with a simple interface that can serve as an introduction to multilevel methods. While it might not be adequate for high-performance computing, it has value as an educational tool, allowing laymen to apply the method to their problems, without having to engage with the subtle details of the underlying theory.

The challenges associated with integrating MLDA within an existing software package ultimately prompted me to launch a new `Python` library, titled `tinyDA`[1]. Still under active development, `tinyDA` provides a high-level interface for the MLDA sampler, supporting a variety of gradient-free MCMC proposals along with both state-dependent and state-independent error modelling. Given that most of the computational cost of quantifying the uncertainty of Bayesian inverse problems is concentrated on the likelihood functional, the software design emphasises using a high-performance black-box forward model, while the actual MCMC iterations are performed using idiomatic `Python` (Rossum, 1995), `NumPy` (Harris et al., 2020) and `SciPy` (Virtanen et al., 2020) code. This makes `tinyDA` easy to customise and extend while maintaining high performance for computationally intensive tasks. Parallelisation is provided through the modern distributed execution framework `Ray` (Moritz et al., 2018), allowing easy deployment on computing platforms of any scale, including clusters. While the actual software of `tinyDA` is ready to use and can be acquired through either GitHub or PyPI, the software documentation is still under development. A journal paper detailing the software design and providing a tutorial and examples of Bayesian inverse problems is under preparation.

[1] https://github.com/mikkelbue/tinyDA

7.1.2 Parallelisation

As discussed in Chapter 4 one possible future research direction for MLDA is the issue of parallelising the (sequential) algorithm. There are two obvious ways to exploit parallel processing in the context of MCMC, namely running multiple parallel chains and parallelising the likelihood functional, i.e. the forward model. While both of these approaches are completely valid and in combination offer some flexibility with respect to exploiting high-performance computing clusters, it would be desirable to have some method for evaluating e.g. multiple model levels for a single chain in parallel. Since the MCMC samplers on each level of MLDA are mutually interdependent, this is not a trivial problem. However, as we propose in Chapter 4, coarse samplers could be allowed to "sample ahead" while the next-finer sampler evaluates the current proposal or an entire chain of proposals (depending on the available resources), in a fashion similar to the pre-fetching approach described in Brockwell, 2006. Since there are only two options for each proposal, i.e. accept or reject, the coarse samplers could construct trees of proposals, so that no matter at what point a proposal is rejected on the next finer level, an alternative would immediately be available to evaluate. This strategy is particularly promising in the context of MLDA, where the finer samplers, given that the next-coarser sampler is a sufficiently good approximation, will have an acceptance rate close to 1. Clearly, this approach would be significantly more wasteful than running MLDA strictly sequentially, since entire chains of pre-fetched proposals may be rejected all at once. As outlined in Chapter 4, the pre-fetching length could be controlled by a reinforcement learning agent that would be trained to minimise waste, while maximising the use of the available resources. It is also possible that the problem could be described in a strictly deterministic way to avoid the additional problem of constructing an adequate reinforcement learning agent. I am highlighting this as a potential avenue for future research.

7.1.3 Tuning

Another possible future research direction is the question of optimal coarse subchain lengths for MLDA. When using the variance reduction techniques described in

Chapter 4, the subchain lengths can be optimised with respect to the acceptable Monte Carlo sampling error. This principle is described in detail in Dodwell et al. (2015), and can also be applied to the MLDA algorithm. Here, care must be taken when deciding the subchain lengths, since the total number of MCMC samples on a given coarse level will depend on the number of samples on the finest level and the subsampling lengths on every level above. However, when variance reduction is not the primary goal of running MLDA, there may be other considerations when deciding the subchain lengths, mainly associated with the cost and quality of the approximation. With a low-cost, high-quality approximation, it would be favourable to run very long subchains to decorrelate proposals to the next-finer level, but for lower quality approximations, the subchains may diverge from the target distribution, with lower acceptance rates on the next-finer level as a result. This effect was demonstrated in Chapter 5, where neural networks trained on fewer data from the prior required shorter subchains to achieve sufficient acceptance rates. When dealing with a high-cost, low-quality approximation, it may be more reasonable to revert to the standard Delayed Acceptance algorithm, where the coarse sampler is simply used as a filter, or to avoid multilevel methods altogether and simply run some standard MCMC sampler.

7.1.4 Comparison with Gradient-based MCMC

Finally, an interesting research question that remains unanswered is how do the multilevel methods, i.e. DA, MLMCMC and MLDA, compare to state of the art gradient-based MCMC such as MALA (Roberts and Tweedie, 1996), HMC (Duane et al., 1987), RMHMC (Girolami and Calderhead, 2011) and the NUTS sampler (Hoffman and Gelman, 2014). In Chapter 4, we demonstrated the superiority of MLDA compared to standard Random Walk Metropolis–Hastings (RWMH) in terms of the number of effective samples produced per second of running each algorithm. While the RWMH sampler was allowed to adapt the global step-size during burn-in, it is well-known that it does not typically perform well, particularly when dealing with high-dimensional target distributions. There are several aspects to this question.

First, if the problem can be described in terms of closed-form probability distributions, for example in the context of Bayesian generalised linear models or mixed-effect models, using multi-level methods may not be the best choice since any coarse approximation would only provide a marginal computational speed-up (except when dealing with excessive amounts of data). In this case, it is almost certainly better to use state-of-the-art gradient-based MCMC, such as the NUTS sampler. Second, if the gradient of the likelihood functional associated with some complex model is available, by way of e.g. an adjoint equation, any gradient-based sampler can be employed on the coarse level. As shown by the proofs presented in Chapter 4, the MLDA algorithm is agnostic to the proposal distribution of the coarsest level, so long as it is in detailed balance with the distribution rendered by its respective forward model. Since gradient-based MCMC algorithms are capable of taking longer steps at each iteration, successive samples generally exhibit less autocorrelation than those produced by RWMH or other gradient-free MCMC methods. This could be exploited in the context of MLDA, either by using shorter subchains to get a similar effective sample size at a lower cost or to use similar subchain lengths to get a higher effective sample size at a similar cost. Third, if the gradient of the likelihood functional is not available in closed form, a sufficiently cheap and accurate coarse model could be exploited to produce a finite difference approximation to the gradient. This approach would be rather sensitive to the dimension of the target distribution but could be viable under the specified conditions. Finally, the multi-level approach could be directly integrated to HMC and its derivatives, since these methods all require taking multiple leapfrog steps when generating a proposal. While the leapfrog integrator is volume-preserving, some error is introduced by using discrete time integration, and so each new proposal must still be subject to a Metropolis accept/reject adjustment to correct for this error. Given this Metropolis adjustment, nothing is preventing us from using a cheaper approximation when performing the leapfrog integration. In conclusion, multilevel methods and gradient-based proposals are not in opposition but could be combined in various ways to harness the strengths of both approaches.

7.2 MCMC for Bayesian Inverse Problems

7.2.1 Barriers to Uptake

While there has been some uptake of various MCMC methods for uncertainty quantification of complex models in applied research environments (see Chapter 2), there is little evidence of use outside academia, i.e. for real-world engineering problems. As mentioned in Chapter 1, this may be explained by the computational burden of running MCMC along with limited awareness of its potential amongst engineering practitioners. Additionally, theoretical studies of MCMC algorithms (including this study) rarely have realistic, worked examples of applications and are commonly restricted to toy problems that broadly illustrate the approach but may fail to demonstrate value in real-world decision making. To enable the transition from traditional inversion techniques and deterministic model calibration to a more rigorous, Bayesian approach using e.g. MCMC, more research into applications to real-world scenarios is required. Not only to provide transparent tutorials for practitioners but also to uncover the complications that will undoubtedly arise from using e.g. non-standard prior distributions and real measurement data. In addition to using real measurement data and modelling domains reflecting the actual physical constraints, such studies should be accompanied by computer code allowing for easy reproduction of the results and adaptation of the methodology to other settings.

This requirement is closely linked to a more technical barrier, namely the absence of an easy-to-use software framework that admits black-box forward models in the likelihood functional. Popular frameworks such as `PyMC` and `Stan` are almost exclusively targeted at data-science related tasks, and are not easily persuaded to tackle other, more complex, problems. The open-source software package `MUQ` (Parno et al., 2014) provides a modular, graph-based interface for MLMCMC and other MCMC algorithms, allowing the specification of a wide range of forward and inverse uncertainty quantification problems. However, while it does provide a high-level `Python` interface, the basic software is written in `C++`, making the learning curve rather steep, in terms of contribution and customisation. There exists a range of implementations of the DREAM family of MCMC samplers (Vrugt, 2016), including a `MATLAB` package, a `Python` package and a standalone commercial application with a graphical interface[2], but the latter two do not appear to be under active development. The current state of the matter means that the application of MCMC to complex (engineering) models requires a high degree of, not only statistical understanding but also programmatical expertise. It is my hope that the MLDA implementation in `PyMC` may attract some attention to the yet unrealised potential of multilevel methods, and provide an entry for practitioners interested in uncertainty quantification. It is my intention to further develop my own MLDA implementation, `tinyDA`[3] to fill this gap. This will be achieved by providing a suite of tools, explicitly designed to perform MCMC in black-box models using pure `Python`, allowing for easy customisation when used for uncertainty quantification of real-world engineering problems, and easy interpretation when used for education.

Another possible obstacle to adaptation may be the difficulty of linking MCMC code with the forward model. In the context of engineering, the forward model is often a partial differential equation (PDE), which requires either intricate knowledge of some discretisation scheme such as the finite difference/element/volume method (FXM), or access to a (typically commercial) third-party software application which handles this discretisation in a quasi black-box fashion. Such third-party applications are mostly not designed to be employed as a plugin for other software, i.e. MCMC code, which complicates using it in the context of uncertainty quantification. In this context, I would like to highlight the excellent geophysical inversion library `pyGIMLi` (Rücker, Günther and Wagner, 2017), which provides, not only a suite of regularisation tools for inverse problems but also implementations of many relevant geophysical models, including cross-hole traveltime tomography, electrical resistivity tomography, gravimetry and subsurface flow modelling. While `pyGIMLi` itself does not include any MCMC code, the software design allows for easily linking the forward models with external code.

[2] https://faculty.sites.uci.edu/jasper/software/

[3] https://github.com/mikkelbue/tinyDA

7.2.2 Separation of Concerns

While running MCMC for a given problem involves solving the forward model many times for the algorithm to converge to the target distribution, one of the greatest attractions of MCMC for uncertainty quantification of inverse problems is that it does not require direct inversion of the model. This means that any inverse problem that can be formulated in the compact form of a Bayesian inverse problem, i.e. $\mathbf{d}_{\text{obs}} = \mathcal{F}(\theta) + \boldsymbol{\epsilon}$, can be inverted using MCMC and there is no requirement for the solver to support sophisticated regularisation techniques since that is all controlled by the Bayesian prior. This important fact underpins the principle of *separation of concerns* with respect to MCMC for Bayesian inverse problems. The actual MCMC code is generally not computationally expensive, and while evaluating the forward model may be very expensive, it does not have to be completely exposed to the MCMC algorithm and can be deployed using any software capable of accepting model input and producing model output. With this in mind, a good MCMC software package for uncertainty quantification of inverse problems in engineering would provide a pipeline for easily specifying $\mathcal{F}(\theta)$ (possibly involving a call to third-party application), which could then be plugged into the appropriate probability distribution. Ideally, it would also implement a library of predefined wrappers, allowing for interfacing with popular FXM software. This has been one of the guiding principles with respect to the development of `tinyDA`. Conversely, the popular MCMC software packages `PyMC` and `Stan` were not designed with this separation of concern in mind, and this is one of the reasons for using them for uncertainty quantification of Bayesian inverse problems can be a frustrating experience.

7.3 Hydrogeological Inverse Problems

7.3.1 Surrogate Coarse Models

In Chapter 5, we demonstrated the use of deep neural networks (DNNs) as an approximation to the forward model PDE in the context of MLDA for uncertainty quantification of a groundwater flow problem. We described a methodology for the design of such DNNs, for both 2D and 3D groundwater flow problems. We showed that using this approach, the effective sample size of the resulting MCMC samples could be significantly increased, compared to the baseline of single-level MCMC. There are some clear advantages to using a DNN as a model approximation, including the ease of defining the model using modern deep-learning frameworks such as `PyTorch` (Paszke et al., 2019) and `TensorFlow` (Abadi et al., 2015), and the flexibility of DNNs with respect to approximating any function (Hornik, Stinchcombe and White, 1989). However, there are also some disadvantages. While we did take the (not insignificant) upfront cost of generating a dataset and training a DNN into consideration, there may be more parsimonious ways to construct a data-driven surrogate model. In this context, we must take into consideration (1) the "appetite" of the chosen surrogate, i.e. how much data does it need to produce a reasonable approximation, (2) the cost of training the surrogate and (3) the cost of evaluating the surrogate. For example, while a Gaussian Process (GP) regression has a low appetite and additionally provides the uncertainty of the approximation, the cost of training a GP for a high-dimensional forward model limits the use to relatively simple problems. However, this could be combined with the active subspace approach of Constantine, Kent and Bui-Thanh (2016) (see Section 2.3.2) with the GP approximation targeting only the active subspace. Ideally, the precomputation cost could be avoided completely by training the surrogate model on-the-fly, using incoming samples from the true forward model as sampling progresses. The sampler would be initialised as single-level sampler, and when enough samples have been collected in this fashion, a surrogate model would be introduced and trained iteratively as more samples from the true model arrive. The MLDA subsampling length could then be increased, as the accuracy of the surrogate improves. The surrogate model could even be trained on rejected MCMC samples, as this would not affect the convergence properties of MLDA. This would require a surrogate model that is particularly fast to train or fit or one that can be improved sequentially. Using linear or quadratic local approximations fitted to neighbouring samples as in Conrad et al. (2016) and Conrad et al. (2017) could be a fruitful approach. While the original algorithm suffers from a bias introduced by using the approximation *in place of* the true forward model, this problem would be avoided in the context of DA and MLDA, where the second Metropolis adjustment step corrects for any bias introduced by the approximation. However, one should take care when applying this approximation method in the context of MLDA, since it is a *state-dependent* approximation, requiring a different form of the second Metropolis adjustment step for subsampling lengths longer than 1, which we have not developed yet. Since this method would fulfil all the above-mentioned considerations, I consider this a potentially highly impactful idea and highlight it as a potential future research direction.

7.3.2 Optimal Design with MLDA

In Chapter 6 we presented a novel method to propose the optimal location of the next monitoring well when conducting a groundwater survey. Broadly, the suggested acquisition function is the product of a measure of uncertainty and the expectation of the solution to some adjoint state equation, that describes the sensitivity of the model with respect to a quantity of interest. We demonstrated numerically that the method could improve the prediction of the quantity of interest, compared to the baseline method, where only the uncertainty was considered. While termed a method for optimal design, it is not nestled in existing "classic" optimal design methods, where the driving principle is usually Fisher information. Instead, the method directly utilises uncertainty estimates discovered by e.g. MCMC and uses the adjoint state equation to link the model to a quantity of interest. The method borrows ideas from previous studies of both adjoint state equations and Bayesian Optimisation, in particular the sensor placement problem. The primary motivation for this study was the actual practical problem at hand, i.e. how to place a monitoring well in a way that would somehow maximise the expected information gain. A secondary motivation was to demonstrate that measures of uncertainty are not only of academic and economic interest but can be utilised to solve practical problems in engineering. I will here briefly remark that although the uncertainty estimates were discovered using MLDA in our work, any method for uncertainty quantification, such as Importance

Sampling or Variational Inference, can be employed for this task. As presented, the method requires an expression for the adjoint state equation, which is not obvious for all quantities of interest and sometimes does not exist. For example, if the quantity of interest is the concentration of a groundwater contaminant at some critical location, the model is typically time-dependent and solving the adjoint state equation is much more computationally demanding than our self-adjoint problem. As suggested in Chapter 6, this could be circumvented by computing the adjoint state for only the mode of the posterior for simple distributions or to use e.g. ensemble techniques for more complicated posteriors. Another way to tackle harder problems would be to solve a simpler but related problem to get some auxiliary information about the primary problem. For example, the hydraulic boundary flux as presented in Chapter 6 over some (potentially interior) boundary could be used as a proxy for predicting the migration of a contaminant. Chapter 6 outlines the method and provides a numerical proof-of-concept, and there are many opportunities to develop this idea further. These include testing the method with different quantities of interest and more elaborate models, such as time-dependent problems and groundwater flow in unconfined aquifers, and extending the idea to other fields of engineering. While the method is presented in the context of groundwater surveying, the underlying idea could easily be extended to any process governed by a PDE with random coefficients.

The litmus test of the optimal design algorithm presented in Chapter 6 would be to test the method in reality. However, as suggested in the relevant discussion, this would involve several complications. First, the method is probabilistic, and there are no guarantees that any one new monitoring well will improve the expectation of the quantity of interest. Second, when the true quantity of interest is unknown there is also no way to validate the outcome. However, the common-sense argument of why it *should* work is relatively straightforward. Using the uncertainty in the acquisition function corresponds broadly to the maximum entropy approach of Bayesian Optimisation for the sensor placement problem (Shewry and Wynn, 1987), but using the uncertainty of the solution to a PDE, rather than a GP. Weighing the uncertainty with the solution to an adjoint state equation simply allows the acquisition function to favour areas that have a high expected influence on our quantity of interest. However, the method is presented as a practical engineering decision support tool and its viability is proven with a numerical example, and there are some theoretical subtleties that have not been thoroughly studied. The best way to develop it further may be to generalise the idea to encompass any PDE with random coefficients and in that context study the subtle relationship between various measures of dispersion and the expected solution to different adjoint state equations.

7.4 Summary and Future Research

I have presented the novel MCMC algorithm Multilevel Delayed Acceptance, extending the Delayed Acceptance algorithm of Christen and Fox (2005) and integrating some of the lessons learned from the debate following the development of the Multilevel MCMC algorithm of Dodwell et al. (2015). Along with the simple yet flexible and powerful base algorithm, we have developed a multilevel error model capable of iteratively correcting every model level according to the finest model, and translated the variance reduction feature of Multilevel Monte Carlo and MLMCMC into the context of MLDA. In Chapter 5, we reimagined the coarse model of MLDA as a deep neural network and demonstrated that a carefully designed DNN could significantly speed up uncertainty quantification of groundwater flow problems. In Chapter 6, we presented a novel take on sequential optimal design using exact statistical expectations discovered by MLDA and demonstrated how such expectations could be utilised to solve a practical groundwater surveying problem. Of the potential future research avenues discussed above, I will now highlight what I consider the two most promising. First, investigating how various surrogate forward models and distributions may be rigorously and flexibly integrated with MLDA and potentially exploited for approximate gradient-based MCMC. While an inexpensive, exact gradient may not always be available for any forward model, an adequate surrogate distribution on the coarsest level would allow MLDA to harness the advantages of e.g. MALA, HMC and NUTS. Second, developing a unified software framework for uncertainty quantification of engineering models that would allow practitioners to easily and flexibly define probabilistic models for their engineering problems and sample from the posterior with minimal human effort. While most engineering businesses these days have the computational resources to perform rigorous uncertainty quantification on their problems, it is rarely done, owing to the lack of a simple and robust software framework.

7.5 Final Remarks

The Bayesian perspective on statistics in general and inverse problems, in particular, presents a simple and elegant framework for exploring uncertainty and a transparent way of constraining ill-posed problems. But it also embodies a much deeper meaning. By defining model parameters probabilistically, we embed ourselves in a universe that can only ever be described in terms of distributions and that can never be uniquely determined. In other words, a universe that is inherently random. This is a provocative sentiment to some, as evidenced by the famous debates between Albert Einstein and Niels Bohr (Skibba, 2018), and a transformative one. Even if it was not metaphysically “true”, it might serve us better to treat it as such. In a universe where all measurements are inherently noisy and all interpretations are subjectively biased, is it not arrogant to assume that there is only one legitimate perspective? Would it not be more honest to be transparent about our subjective biases and (at least attempt to) consider all possible perspectives?

Bibliography

M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro, G. S. Corrado, A. Davis, J. Dean, M. Devin et al. (2015). *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. URL: `https://www.tensorflow.org/` (Cited on page 134).

A. Alcolea, J. Carrera and A. Medina (Nov. 2006). 'Pilot points method incorporating prior information for solving the groundwater flow inverse problem'. en. In: *Advances in Water Resources* 29.11, pp. 1678–1689. ISSN: 03091708. DOI: `10.1016/j.advwatres.2005.12.009`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S0309170805002976` (visited on 07/11/2021) (Cited on page 12).

M. P. Anderson, W. W. Woessner and R. J. Hunt (2015). *Applied groundwater modeling: simulation of flow and advective transport*. Second edition. OCLC: ocn921253555. London ; San Diego, CA: Academic Press. ISBN: 978-0-12-058103-0 (Cited on page 10).

C. Andrieu and E. Moulines (Aug. 2006). 'On the ergodicity properties of some adaptive MCMC algorithms'. en. In: *The Annals of Applied Probability* 16.3. ISSN: 1050-5164. DOI: `10.1214/105051606000000286`. URL: `https://projecteuclid.org/journals/annals-of-applied-probability/volume-16/issue-3/On-the-ergodicity-properties-of-some-adaptive-MCMC-algorithms/10.1214/105051606000000286.full` (visited on 08/12/2021) (Cited on page 8).

C. Andrieu and J. Thoms (Dec. 2008). 'A tutorial on adaptive MCMC'. en. In: *Statistics and Computing* 18.4, pp. 343–373. ISSN: 0960-3174, 1573-1375. DOI: `10.1007/s11222-008-9110-y`. URL: `http://link.springer.com/10.1007/s11222-008-9110-y` (visited on 15/03/2021) (Cited on pages 8, 19 and 20).

T. Arbogast (Sept. 2002). 'Implementation of a Locally Conservative Numerical Subgrid Upscaling Scheme for Two-Phase Darcy Flow'. In: *Computational Geosciences* 6.3, pp. 453–481. ISSN: 1573-1499. DOI: 10.1023/A:1021295215383. URL: https://doi.org/10.1023/A:1021295215383 (Cited on page 28).

Y. F. Atchadé and J. S. Rosenthal (Oct. 2005). 'On adaptive Markov chain Monte Carlo algorithms'. en. In: *Bernoulli* 11.5, pp. 815–828. ISSN: 1350-7265. DOI: 10.3150/bj/1130077595. URL: http://projecteuclid.org/euclid.bj/1130077595 (visited on 15/03/2021) (Cited on page 8).

E. Auken, T. Boesen and A. V. Christiansen (2017). 'A Review of Airborne Electromagnetic Methods With Focus on Geotechnical and Hydrological Applications From 2007 to 2017'. In: ed. by L. Nielsen. Vol. 58. Advances in Geophysics. ISSN: 0065-2687. Elsevier, pp. 47–93. DOI: https://doi.org/10.1016/bs.agph.2017.10.002. URL: https://www.sciencedirect.com/science/article/pii/S006526871730002X (Cited on page 11).

E. Auken, N. Foged, J. J. Larsen, K. V. T. Lassen, P. K. Maurya, S. M. Dath and T. T. Eiskjær (Jan. 2019). 'tTEM — A towed transient electromagnetic system for detailed 3D imaging of the top 70 m of the subsurface'. en. In: *GEOPHYSICS* 84.1, E13–E22. ISSN: 0016-8033, 1942-2156. DOI: 10.1190/geo2018-0355.1. URL: https://library.seg.org/doi/10.1190/geo2018-0355.1 (visited on 04/10/2021) (Cited on page 11).

A. Beskos, M. Girolami, S. Lan, P. E. Farrell and A. M. Stuart (Apr. 2017). 'Geometric MCMC for infinite-dimensional inverse problems'. en. In: *Journal of Computational Physics* 335, pp. 327–351. ISSN: 00219991. DOI: 10.1016/j.jcp.2016.12.041. URL: https://linkinghub.elsevier.com/retrieve/pii/S0021999116307033 (visited on 16/08/2019) (Cited on page 37).

A. E Brockwell (Mar. 2006). 'Parallel Markov chain Monte Carlo Simulation by Pre-Fetching'. en. In: *Journal of Computational and Graphical Statistics* 15.1, pp. 246–261. ISSN: 1061-8600, 1537-2715. DOI: 10.1198/106186006X100579. URL: https://www.tandfonline.com/doi/full/10.1198/106186006X100579 (visited on 01/04/2021) (Cited on page 128).

S Brooks (2011). *Handbook of Markov Chain Monte Carlo.* English. OCLC: 952520183. Boca Raton: CRC netBASE. ISBN: 978-1-4200-7942-5 (Cited on pages 7, 9 and 16).

J. Brynjarsdóttir and A. O'Hagan (Nov. 2014). 'Learning about physical parameters: the importance of model discrepancy'. en. In: *Inverse Problems* 30.11, p. 114007. ISSN: 0266-5611, 1361-6420. DOI: 10.1088/0266-5611/30/11/114007. URL: http://stacks.iop.org/0266-5611/30/i=11/a=114007?key=crossref.7b886360dda7b385609c577ad82450aa (visited on 03/03/2020) (Cited on page 24).

J. E. Capilla, J. Jaime Gómez-Hernández and A. Sahuquillo (Dec. 1997). 'Stochastic simulation of transmissivity fields conditional to both transmissivity and piezometric data 2. Demonstration on a synthetic aquifer'. en. In: *Journal of Hydrology* 203.1-4, pp. 175–188. ISSN: 00221694. DOI: 10.1016/S0022-1694(97)00097-8. URL: https://linkinghub.elsevier.com/retrieve/pii/S0022169497000978 (visited on 07/11/2021) (Cited on page 14).

J. E. Capilla, J. Jaime Gómez-Hernández and A. Sahuquillo (June 1998). 'Stochastic simulation of transmissivity fields conditional to both transmissivity and piezometric head data—3. Application to the Culebra formation at the Waste Isolation Pilot Plan (WIPP), New Mexico, USA'. en. In: *Journal of Hydrology* 207.3-4, pp. 254–269. ISSN: 00221694. DOI: 10.1016/S0022-1694(98)00138-3. URL: https://linkinghub.elsevier.com/retrieve/pii/S0022169498001383 (visited on 07/11/2021) (Cited on page 14).

B. Carpenter, A. Gelman, M. D. Hoffman, D. Lee, B. Goodrich, M. Betancourt, M. Brubaker, J. Guo, P. Li and A. Riddell (2017). 'Stan: A probabilistic programming language'. In: *Journal of statistical software* 76.1 (Cited on page 17).

J. Carrera and S. P. Neuman (Feb. 1986). 'Estimation of Aquifer Parameters Under Transient and Steady State Conditions: 1. Maximum Likelihood Method Incorporating Prior Information'. en. In: *Water Resources Research* 22.2, pp. 199–210. ISSN: 00431397. DOI: 10.1029/WR022i002p00199. URL: http://doi.wiley.com/

10.1029/WR022i002p00199 (visited on 30/08/2019) (Cited on pages 11, 12 and 14).

C. Certes and G. de Marsily (Oct. 1991). ‘Application of the pilot point method to the identification of aquifer transmissivities’. en. In: *Advances in Water Resources* 14.5, pp. 284–300. ISSN: 03091708. DOI: 10.1016/0309-1708(91)90040-U. URL: https://linkinghub.elsevier.com/retrieve/pii/030917089190040U (visited on 06/03/2020) (Cited on page 12).

J. Charrier, R. Scheichl and A. L. Teckentrup (Jan. 2013). ‘Finite Element Error Analysis of Elliptic PDEs with Random Coefficients and Its Application to Multilevel Monte Carlo Methods’. en. In: *SIAM Journal on Numerical Analysis* 51.1, pp. 322–352. ISSN: 0036-1429, 1095-7170. DOI: 10.1137/110853054. URL: http://epubs.siam.org/doi/10.1137/110853054 (visited on 05/08/2021) (Cited on page 26).

J. A. Christen and C. Fox (Dec. 2005). ‘Markov chain Monte Carlo Using an Approximation’. en. In: *Journal of Computational and Graphical Statistics* 14.4, pp. 795–810. ISSN: 1061-8600, 1537-2715. DOI: 10.1198/106186005X76983. URL: http://www.tandfonline.com/doi/abs/10.1198/106186005X76983 (visited on 02/10/2019) (Cited on pages 3, 23, 24, 126 and 137).

S. Christensen and J. Doherty (Apr. 2008). ‘Predictive error dependencies when using pilot points and singular value decomposition in groundwater model calibration’. en. In: *Advances in Water Resources* 31.4, pp. 674–700. ISSN: 03091708. DOI: 10.1016/j.advwatres.2008.01.003. URL: http://linkinghub.elsevier.com/retrieve/pii/S0309170808000092 (visited on 28/12/2018) (Cited on page 13).

K. A. Cliffe, M. B. Giles, R. Scheichl and A. L. Teckentrup (Jan. 2011). ‘Multilevel Monte Carlo methods and applications to elliptic PDEs with random coefficients’. en. In: *Computing and Visualization in Science* 14.1, pp. 3–15. ISSN: 1432-9360, 1433-0369. DOI: 10.1007/s00791-011-0160-x. URL: http://link.springer.com/10.1007/s00791-011-0160-x (visited on 11/02/2019) (Cited on page 26).

P. Conrad, A. Davis, Y. Marzouk, N. Pillai and A. Smith (Dec. 2017). ‘Parallel local approximation MCMC for expensive models’. en. In: *arXiv:1607.02788 [stat]*. arXiv: 1607.02788. URL: `http://arxiv.org/abs/1607.02788` (visited on 26/03/2020) (Cited on pages 36, 37 and 134).

P. R. Conrad, Y. M. Marzouk, N. S. Pillai and A. Smith (Oct. 2016). ‘Accelerating Asymptotically Exact MCMC for Computationally Intensive Models via Local Approximations’. en. In: *Journal of the American Statistical Association* 111.516, pp. 1591–1607. ISSN: 0162-1459, 1537-274X. DOI: `10.1080/01621459.2015.1096787`. URL: `https://www.tandfonline.com/doi/full/10.1080/01621459.2015.1096787` (visited on 13/11/2021) (Cited on pages 36, 37 and 134).

P. G. Constantine, C. Kent and T. Bui-Thanh (Jan. 2016). ‘Accelerating Markov Chain Monte Carlo with Active Subspaces’. en. In: *SIAM Journal on Scientific Computing* 38.5, A2779–A2805. ISSN: 1064-8275, 1095-7197. DOI: `10.1137/15M1042127`. URL: `http://epubs.siam.org/doi/10.1137/15M1042127` (visited on 30/07/2019) (Cited on pages 35, 37 and 134).

S. L. Cotter, G. O. Roberts, A. M. Stuart and D. White (Aug. 2013). ‘MCMC Methods for Functions: Modifying Old Algorithms to Make Them Faster’. en. In: *Statistical Science* 28.3. arXiv: 1202.0709, pp. 424–446. ISSN: 0883-4237. DOI: `10.1214/13-STS421`. URL: `http://arxiv.org/abs/1202.0709` (visited on 12/05/2019) (Cited on pages 20, 21 and 37).

T. Cui, C. Fox and M. J. O’Sullivan (Oct. 2011). ‘Bayesian calibration of a large-scale geothermal reservoir model by a new adaptive delayed acceptance Metropolis Hastings algorithm: ADAPTIVE DELAYED ACCEPTANCE METROPOLIS-HASTINGS ALGORITHM’. en. In: *Water Resources Research* 47.10. ISSN: 00431397. DOI: `10.1029/2010WR010352`. URL: `http://doi.wiley.com/10.1029/2010WR010352` (visited on 25/02/2020) (Cited on page 25).

T Cui, J Martin, Y. M. Marzouk, A Solonen and A Spantini (Nov. 2014). ‘Likelihood-informed dimension reduction for nonlinear inverse problems’. en. In: *Inverse Problems* 30.11, p. 114015. ISSN: 0266-5611, 1361-6420. DOI: `10.1088/0266-5611/30/11/114015`. URL: `http://stacks.iop.org/0266-5611/30/i=`

11/a=114015?key=crossref.cf9640d6b4de5de065991958a8528564 (visited on 03/09/2019) (Cited on page 35).

T. Cui, C. Fox and M. O'Sullivan (Dec. 2012). *Adaptive Error Modelling in MCMC Sampling for Large Scale Inverse Problems.* Tech. rep. 687, ISSN 1178-360. Univeristy of Auckland, Faculty of Engineering (Cited on page 25).

T. Cui, C. Fox and M. J. O'Sullivan (June 2019). 'A posteriori stochastic correction of reduced models in delayed-acceptance MCMC, with application to multiphase subsurface inverse problems: Stochastic correction of reduced models in delayed-acceptance MCMC'. en. In: *International Journal for Numerical Methods in Engineering* 118.10, pp. 578–605. ISSN: 00295981. DOI: 10.1002/nme.6028. URL: https://onlinelibrary.wiley.com/doi/10.1002/nme.6028 (visited on 05/08/2021) (Cited on page 25).

T. Cui, K. J. H. Law and Y. M. Marzouk (Jan. 2016). 'Dimension-independent likelihood-informed MCMC'. en. In: *Journal of Computational Physics* 304. arXiv: 1411.3688, pp. 109–137. ISSN: 00219991. DOI: 10.1016/j.jcp.2015.10.008. URL: http://arxiv.org/abs/1411.3688 (visited on 31/07/2019) (Cited on pages 21 and 37).

C. R. Dietrich and G. N. Newsam (July 1997). 'Fast and Exact Simulation of Stationary Gaussian Processes through Circulant Embedding of the Covariance Matrix'. en. In: *SIAM Journal on Scientific Computing* 18.4, pp. 1088–1107. ISSN: 1064-8275, 1095-7197. DOI: 10.1137/S1064827592240555. URL: http://epubs.siam.org/doi/10.1137/S1064827592240555 (visited on 10/11/2021) (Cited on page 31).

T. J. Dodwell, C. Ketelsen, R. Scheichl and A. L. Teckentrup (Jan. 2015). 'A Hierarchical Multilevel Markov Chain Monte Carlo Algorithm with Applications to Uncertainty Quantification in Subsurface Flow'. en. In: *SIAM/ASA Journal on Uncertainty Quantification* 3.1, pp. 1075–1108. ISSN: 2166-2525. DOI: 10.1137/130915005. URL: http://epubs.siam.org/doi/10.1137/130915005 (visited on 28/12/2018) (Cited on pages 3, 26, 129 and 137).

J. E. Doherty and R. J. Hunt (2010). *Approaches to Highly Parameterized Inversion: A Guide to Using PEST for Groundwater-Model Calibration.* en. Tech. rep. Reston, Virginia: U.S. Geological Survey (Cited on pages 13 and 14).

J. E. Doherty, R. J. Hunt and M. J. Tonkin (2010). *Approaches to highly parameterized inversion: A guide to using PEST for model-parameter and predictive-uncertainty analysis.* English. Report 2010-5211. Reston, VA, pp. i–71. DOI: 10.3133/sir20105211. URL: http://pubs.er.usgs.gov/publication/sir20105211 (Cited on pages 13, 14 and 15).

P. Dostert, Y. Efendiev and B. Mohanty (Mar. 2009). 'Efficient uncertainty quantification techniques in inverse problems for Richards' equation using coarse-scale simulation models'. en. In: *Advances in Water Resources* 32.3, pp. 329–339. ISSN: 03091708. DOI: 10.1016/j.advwatres.2008.11.009. URL: https://linkinghub.elsevier.com/retrieve/pii/S0309170808002121 (visited on 07/02/2019) (Cited on page 28).

S. Duane, A. Kennedy, B. J. Pendleton and D. Roweth (Sept. 1987). 'Hybrid Monte Carlo'. en. In: *Physics Letters B* 195.2, pp. 216–222. ISSN: 03702693. DOI: 10.1016/0370-2693(87)91197-X. URL: https://linkinghub.elsevier.com/retrieve/pii/037026938791197X (visited on 06/08/2021) (Cited on pages 17 and 129).

L. J. Durlofsky (Mar. 1998). 'Coarse scale models of two phase flow in heterogeneous reservoirs: volume averaged equations and their relationship to existing upscaling techniques'. In: *Computational Geosciences* 2.2, pp. 73–92. ISSN: 1573-1499. DOI: 10.1023/A:1011593901771. URL: https://doi.org/10.1023/A:1011593901771 (Cited on page 28).

Y. Efendiev, V. Ginting and T. Y. Hou (2004). 'Multiscale Finite Element Methods for Nonlinear Problems and Their Applications'. en. In: *Communications in Mathematical Sciences* 2.4, pp. 553–589. ISSN: 15396746, 19450796. DOI: 10.4310/CMS.2004.v2.n4.a2. URL: http://www.intlpress.com/site/pub/pages/journals/items/cms/content/vols/0002/0004/a002/ (visited on 08/11/2021) (Cited on page 28).

Y. Efendiev, T. Hou and W. Luo (Jan. 2006). 'Preconditioning Markov Chain Monte Carlo Simulations Using Coarse-Scale Models'. en. In: *SIAM Journal on Scientific Computing* 28.2, pp. 776–803. ISSN: 1064-8275, 1095-7197. DOI: 10.1137/050628568. URL: http://epubs.siam.org/doi/10.1137/050628568 (visited on 02/11/2021) (Cited on pages 24 and 28).

T. A. El Moselhy and Y. M. Marzouk (Oct. 2012). 'Bayesian inference with optimal maps'. en. In: *Journal of Computational Physics* 231.23, pp. 7815–7850. ISSN: 00219991. DOI: 10.1016/j.jcp.2012.07.022. URL: https://linkinghub.elsevier.com/retrieve/pii/S0021999112003956 (visited on 04/11/2021) (Cited on page 20).

J. M. Fenelon (2005). *Analysis of Ground-Water Levels and Associated Trends in Yucca Flat, Nevada Test Site, Nye County, Nevada, 1951-2003.* en. Scientific Investigations Report 2055-5175. USGS, p. 97 (Cited on page 29).

C. Fox (Mar. 2021). 'Ergodicity of Multilevel MCMC as Adaptive Delayed Acceptance'. In: *Multiscale and Multilevel Methods for Uncertainty Quantification.* Virtual Conference: Society for Industrial and Applied Mathematics. URL: https://meetings.siam.org/sess/dsp_talk.cfm?p=107731 (Cited on page 27).

J. Fu and J. J. Gómez-Hernández (Feb. 2009). 'A Blocking Markov Chain Monte Carlo Method for Inverse Stochastic Hydrogeological Modeling'. en. In: *Mathematical Geosciences* 41.2, pp. 105–128. ISSN: 1874-8961, 1874-8953. DOI: 10.1007/s11004-008-9206-0. URL: http://link.springer.com/10.1007/s11004-008-9206-0 (visited on 07/02/2019) (Cited on page 28).

A. E. Gelfand and A. F. M. Smith (1990). 'Sampling-Based Approaches to Calculating Marginal Densities'. In: *Journal of the American Statistical Association* 85.410. Publisher: [American Statistical Association, Taylor & Francis, Ltd.], pp. 398–409. ISSN: 0162-1459. DOI: 10.2307/2289776. URL: https://www.jstor.org/stable/2289776 (visited on 30/10/2021) (Cited on page 17).

A. Gelman, ed. (2004). *Bayesian data analysis.* 2nd ed. Texts in statistical science. Boca Raton, Fla: Chapman & Hall/CRC. ISBN: 978-1-58488-388-3 (Cited on pages 7 and 31).

A. Gelman and D. B. Rubin (Nov. 1992). 'Inference from Iterative Simulation Using Multiple Sequences'. en. In: *Statistical Science* 7.4, pp. 457–472. ISSN: 0883-4237. DOI: 10.1214/ss/1177011136. URL: http://projecteuclid.org/euclid.ss/1177011136 (visited on 27/02/2020) (Cited on page 31).

S. Geman and D. Geman (Nov. 1984). 'Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images'. In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* PAMI-6.6. Conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence, pp. 721–741. ISSN: 1939-3539. DOI: 10.1109/TPAMI.1984.4767596 (Cited on page 16).

C. J. Geyer (1991). 'Markov Chain Monte Carlo Maximum Likelihood'. en. In: *Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface*. Ed. by E. M. Keramidas. Vol. 23. Seattle: Interface Foundation of North America, pp. 156–163 (Cited on page 35).

M. B. Giles (June 2008a). 'Multilevel Monte Carlo Path Simulation'. en. In: *Operations Research* 56.3, pp. 607–617. ISSN: 0030-364X, 1526-5463. DOI: 10.1287/opre.1070.0496. URL: http://pubsonline.informs.org/doi/abs/10.1287/opre.1070.0496 (visited on 11/02/2019) (Cited on pages 26 and 126).

M. Giles (2008b). 'Improved Multilevel Monte Carlo Convergence using the Milstein Scheme'. en. In: *Monte Carlo and Quasi-Monte Carlo Methods 2006*. Ed. by A. Keller, S. Heinrich and H. Niederreiter. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 343–358. ISBN: 978-3-540-74495-5 978-3-540-74496-2. DOI: 10.1007/978-3-540-74496-2_20. URL: http://link.springer.com/10.1007/978-3-540-74496-2_20 (visited on 03/11/2021) (Cited on page 26).

M. Girolami and B. Calderhead (Mar. 2011). 'Riemann manifold Langevin and Hamiltonian Monte Carlo methods: Riemann Manifold Langevin and Hamiltonian Monte Carlo Methods'. en. In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 73.2, pp. 123–214. ISSN: 13697412. DOI: 10.1111/j.1467-9868.2010.00765.x. URL: https://onlinelibrary.wiley.com/doi/10.1111/j.1467-9868.2010.00765.x (visited on 04/11/2021) (Cited on pages 20 and 129).

P. J. Green (1995). ‘Reversible jump Markov chain Monte Carlo computation and Bayesian model determination’. en. In: *Biometrika*, p. 22 (Cited on page 29).

C. M. Grinstead and J. L. Snell (1997). *Introduction to probability.* 2nd rev. ed. Providence, RI: American Mathematical Society. ISBN: 978-0-8218-0749-1 (Cited on page 75).

F. B. Guardiano and R. M. Srivastava (1993). ‘Multivariate Geostatistics: Beyond Bivariate Moments’. In: *Geostatistics Tróia ’92: Volume 1.* Ed. by A. Soares. Dordrecht: Springer Netherlands, pp. 133–144. ISBN: 978-94-011-1739-5. DOI: 10.1007/978-94-011-1739-5_12. URL: https://doi.org/10.1007/978-94-011-1739-5_12 (Cited on page 32).

J. J. Gómez-Hernánez, H.-J. W. M. H. Franssen and A. Sahuquillo (Nov. 2003). ‘Stochastic conditional inverse modeling of subsurface mass transport: A brief review and the self-calibrating method’. en. In: *Stochastic Environmental Research and Risk Assessment (SERRA)* 17.5, pp. 319–328. ISSN: 1436-3240, 1436-3259. DOI: 10.1007/s00477-003-0153-5. URL: http://link.springer.com/10.1007/s00477-003-0153-5 (visited on 06/11/2021) (Cited on page 14).

J. Gómez-Hernánez, A. Sahuquillo and J. E. Capilla (Dec. 1997). ‘Stochastic simulation of transmissivity fields conditional to both transmissivity and piezometric data—I. Theory’. en. In: *Journal of Hydrology* 203.1-4, pp. 162–174. ISSN: 00221694. DOI: 10.1016/S0022-1694(97)00098-X. URL: http://linkinghub.elsevier.com/retrieve/pii/S002216949700098X (visited on 28/12/2018) (Cited on page 14).

H. Haario, E. Saksman and J. Tamminen (Sept. 1999). ‘Adaptive proposal distribution for random walk Metropolis algorithm’. en. In: *Computational Statistics* 14.3, pp. 375–395. ISSN: 0943-4062, 1613-9658. DOI: 10.1007/s001800050022. URL: http://link.springer.com/10.1007/s001800050022 (visited on 31/10/2021) (Cited on page 19).

H. Haario, E. Saksman and J. Tamminen (Apr. 2001). ‘An Adaptive Metropolis Algorithm’. en. In: *Bernoulli* 7.2, p. 223. ISSN: 13507265. DOI: 10.2307/3318737.

URL: https://www.jstor.org/stable/3318737?origin=crossref (visited on 05/10/2019) (Cited on pages 18 and 19).

H. Haario, E. Saksman and J. Tamminen (June 2005). ‘Componentwise adaptation for high dimensional MCMC’. en. In: *Computational Statistics* 20.2, pp. 265–273. ISSN: 0943-4062, 1613-9658. DOI: 10.1007/BF02789703. URL: http://link.springer.com/10.1007/BF02789703 (visited on 17/03/2021) (Cited on page 19).

N. Hansen and A. Ostermeier (June 2001). ‘Completely Derandomized Self-Adaptation in Evolution Strategies’. en. In: *Evolutionary Computation* 9.2, pp. 159–195. ISSN: 1063-6560, 1530-9304. DOI: 10.1162/106365601750190398. URL: https://direct.mit.edu/evco/article/9/2/159-195/892 (visited on 10/11/2021) (Cited on page 29).

P. C. Hansen (Jan. 2010). *Discrete Inverse Problems: Insight and Algorithms*. en. Society for Industrial and Applied Mathematics. ISBN: 978-0-89871-696-2 978-0-89871-883-6. DOI: 10.1137/1.9780898718836. URL: http://epubs.siam.org/doi/book/10.1137/1.9780898718836 (visited on 06/03/2020) (Cited on page 15).

C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith et al. (Sept. 2020). ‘Array programming with NumPy’. In: *Nature* 585.7825, pp. 357–362. DOI: 10.1038/s41586-020-2649-2. URL: https://doi.org/10.1038/s41586-020-2649-2 (Cited on page 127).

W. K. Hastings (1970). ‘Monte Carlo sampling methods using Markov chains and their applications’. en. In: *Biometrika*, p. 13 (Cited on page 16).

S. Heinrich (2001). ‘Multilevel Monte Carlo Methods’. In: *Proceedings of the Third International Conference on Large-Scale Scientific Computing-Revised Papers*. LSSC ’01. London, UK, UK: Springer-Verlag, pp. 58–67. ISBN: 978-3-540-43043-8. URL: http://dl.acm.org/citation.cfm?id=645740.666755 (visited on 11/02/2019) (Cited on page 26).

F. Heße, A. Comunian and S. Attinger (June 2019). ‘What We Talk About When We Talk About Uncertainty. Toward a Unified, Data-Driven Framework for Uncertainty Characterization in Hydrogeology’. en. In: *Frontiers in Earth Science* 7, p. 118. ISSN: 2296-6463. DOI: 10.3389/feart.2019.00118. URL: https://www.frontiersin.org/article/10.3389/feart.2019.00118/full (visited on 10/11/2021) (Cited on page 32).

D. Higdon, H. Lee and C. Holloman (2003). ‘Markov chain Monte Carlo-based approaches for inference in computationally intensive inverse problems’. en. In: *Bayesian Statistics*. Vol. 7. Oxford University Press., pp. 181–197 (Cited on pages 33 and 34).

A. C. Hinnell, T. P. A. Ferré, J. A. Vrugt, J. A. Huisman, S. Moysey, J. Rings and M. B. Kowalsky (2010). ‘Improved extraction of hydrologic information from geophysical data through coupled hydrogeophysical inversion’. en. In: *Water Resources Research* 46.4. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2008WR007060. ISSN: 1944-7973. DOI: 10.1029/2008WR007060. URL: https://onlinelibrary.wiley.com/doi/abs/10.1029/2008WR007060 (visited on 01/11/2021) (Cited on page 22).

M. D. Hoffman and A. Gelman (2014). ‘The No-U-Turn Sampler: Adaptively Setting Path Lengths in Hamiltonian Monte Carlo’. In: *Journal of Machine Learning Research* 15.47, pp. 1593–1623. URL: http://jmlr.org/papers/v15/hoffman14a.html (Cited on pages 9, 17 and 129).

K. Hornik, M. Stinchcombe and H. White (Jan. 1989). ‘Multilayer feedforward networks are universal approximators’. en. In: *Neural Networks* 2.5, pp. 359–366. ISSN: 08936080. DOI: 10.1016/0893-6080(89)90020-8. URL: https://linkinghub.elsevier.com/retrieve/pii/0893608089900208 (visited on 15/12/2021) (Cited on page 134).

Z. Hu, Z. Yao and J. Li (Apr. 2016). ‘On an adaptive preconditioned Crank-Nicolson MCMC algorithm for infinite dimensional Bayesian inferences’. en. In: *arXiv:1511.05838 [math, stat]*. arXiv: 1511.05838. URL: http://arxiv.org/abs/1511.05838 (visited on 09/12/2019) (Cited on page 21).

J. Kaipio and E. Somersalo (Jan. 2007). 'Statistical inverse problems: Discretization, model reduction and inverse crimes'. en. In: *Journal of Computational and Applied Mathematics* 198.2, pp. 493–504. ISSN: 03770427. DOI: 10.1016/j.cam.2005.09.027. URL: https://linkinghub.elsevier.com/retrieve/pii/S0377042705007296 (visited on 11/10/2019) (Cited on page 24).

E. H. Keating, J. Doherty, J. A. Vrugt and Q. Kang (2010). 'Optimization and uncertainty assessment of strongly nonlinear groundwater models with high parameter dimensionality'. en. In: *Water Resources Research* 46.10. _eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2009WR008584. ISSN: 1944-7973. DOI: https://doi.org/10.1029/2009WR008584. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2009WR008584 (visited on 12/04/2021) (Cited on pages 22, 29 and 30).

P. K. Kitanidis and E. G. Vomvoris (1983). 'A geostatistical approach to the inverse problem in groundwater modeling (steady state) and one-dimensional simulations'. en. In: *Water Resources Research* 19.3, pp. 677–690. ISSN: 1944-7973. DOI: 10.1029/WR019i003p00677. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/WR019i003p00677 (visited on 04/02/2019) (Cited on page 12).

D. K. S. Y. Klaas and M. A. Imteaz (Sept. 2017). 'Investigating the impact of the properties of pilot points on calibration of groundwater models: case study of a karst catchment in Rote Island, Indonesia'. en. In: *Hydrogeology Journal* 25.6, pp. 1703–1719. ISSN: 1431-2174, 1435-0157. DOI: 10.1007/s10040-017-1590-4. URL: http://link.springer.com/10.1007/s10040-017-1590-4 (visited on 07/11/2021) (Cited on page 13).

T. S. Kuhn (1996). *The structure of scientific revolutions*. 3rd ed. Chicago, IL: University of Chicago Press. ISBN: 978-0-226-45807-6 978-0-226-45808-3 (Cited on page 2).

E. Laloy, R. Hérault, D. Jacques and N. Linde (Jan. 2018). 'Training-Image Based Geostatistical Inversion Using a Spatial Generative Adversarial Neural Network'. en. In: *Water Resources Research* 54.1, pp. 381–406. ISSN: 0043-1397, 1944-7973. DOI: 10.1002/2017WR022148. URL: https://onlinelibrary.wiley.com/doi/

abs/10.1002/2017WR022148 (visited on 15/04/2020) (Cited on pages 32 and 33).

E. Laloy, N. Linde, D. Jacques and J. A. Vrugt (2015). 'Probabilistic inference of multi-Gaussian fields from indirect hydrological data using circulant embedding and dimensionality reduction'. en. In: *Water Resources Research* 51.6. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/2014WR016395, pp. 4224–4243. ISSN: 1944-7973. DOI: 10.1002/2014WR016395. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/2014WR016395 (visited on 09/11/2021) (Cited on page 31).

E. Laloy, B. Rogiers, J. A. Vrugt, D. Mallants and D. Jacques (May 2013). 'Efficient posterior exploration of a high-dimensional groundwater model from two-stage Markov chain Monte Carlo simulation and polynomial chaos expansion: Speeding up MCMC Simulation of a Groundwater Model'. en. In: *Water Resources Research* 49.5, pp. 2664–2682. ISSN: 00431397. DOI: 10.1002/wrcr.20226. URL: http://doi.wiley.com/10.1002/wrcr.20226 (visited on 19/02/2020) (Cited on pages 22, 24 and 31).

E. Laloy and J. A. Vrugt (Jan. 2012). 'High-dimensional posterior exploration of hydrologic models using multiple-try DREAM $_{(ZS)}$ and high-performance computing: EFFICIENT MCMC FOR HIGH-DIMENSIONAL PROBLEMS'. en. In: *Water Resources Research* 48.1. ISSN: 00431397. DOI: 10.1029/2011WR010608. URL: http://doi.wiley.com/10.1029/2011WR010608 (visited on 22/03/2021) (Cited on page 23).

S. Lan (Sept. 2019). 'Adaptive dimension reduction to accelerate infinite-dimensional geometric Markov Chain Monte Carlo'. en. In: *Journal of Computational Physics* 392, pp. 71–95. ISSN: 00219991. DOI: 10.1016/j.jcp.2019.04.043. URL: https://linkinghub.elsevier.com/retrieve/pii/S002199911930289X (visited on 13/11/2021) (Cited on page 37).

K. J. H. Law (May 2014). 'Proposals which speed-up function-space MCMC'. en. In: *Journal of Computational and Applied Mathematics* 262. arXiv: 1212.4767, pp. 127–138. ISSN: 03770427. DOI: 10.1016/j.cam.2013.07.026. URL: http://arxiv.org/abs/1212.4767 (visited on 18/05/2020) (Cited on page 21).

J. S. Liu (2004). *Monte Carlo Strategies in Scientific Computing.* en. Springer Series in Statistics. New York, NY: Springer New York. ISBN: 978-0-387-76369-9 978-0-387-76371-2. DOI: 10.1007/978-0-387-76371-2. URL: http://link.springer.com/10.1007/978-0-387-76371-2 (visited on 13/11/2020) (Cited on pages 7 and 24).

J. S. Liu, F. Liang and W. H. Wong (Mar. 2000). 'The Multiple-Try Method and Local Optimization in Metropolis Sampling'. en. In: *Journal of the American Statistical Association* 95.449, pp. 121–134. ISSN: 0162-1459, 1537-274X. DOI: 10.1080/01621459.2000.10473908. URL: http://www.tandfonline.com/doi/abs/10.1080/01621459.2000.10473908 (visited on 30/01/2021) (Cited on pages 23 and 27).

M. Loke, J. Chambers, D. Rucker, O. Kuras and P. Wilkinson (Aug. 2013). 'Recent developments in the direct-current geoelectrical imaging method'. en. In: *Journal of Applied Geophysics* 95, pp. 135–156. ISSN: 09269851. DOI: 10.1016/j.jappgeo.2013.02.017. URL: https://linkinghub.elsevier.com/retrieve/pii/S0926985113000499 (visited on 04/10/2021) (Cited on page 11).

D. Lunn, D. Spiegelhalter, A. Thomas and N. Best (2009). 'The BUGS project: Evolution, critique and future directions'. en. In: *Statistics in Medicine* 28.25. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/sim.3680, pp. 3049–3067. ISSN: 1097-0258. DOI: 10.1002/sim.3680. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/sim.3680 (visited on 30/10/2021) (Cited on page 17).

M. B. Lykkegaard, T. J. Dodwell and D. Moxey (2021). 'Accelerating uncertainty quantification of groundwater flow modelling using a deep neural network proxy'. In: *Computer Methods in Applied Mechanics and Engineering* 383, p. 113895. ISSN: 0045-7825. DOI: https://doi.org/10.1016/j.cma.2021.113895. URL: https://www.sciencedirect.com/science/article/pii/S0045782521002322 (Cited on pages 24, 31 and 75).

M. B. Lykkegaard, G. Mingas, R. Scheichl, C. Fox and T. J. Dodwell (2020). *Multilevel Delayed Acceptance MCMC with an Adaptive Error Model in PyMC3*. arXiv: `2012.05668 [stat.CO]` (Cited on page 40).

B. Malama, K. L. Kuhlman and S. C. James (2013). 'Core-scale solute transport model selection using Monte Carlo analysis'. en. In: *Water Resources Research* 49.6. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/wrcr.20273, pp. 3133–3147. ISSN: 1944-7973. DOI: `10.1002/wrcr.20273`. URL: `https://onlinelibrary.wiley.com/doi/abs/10.1002/wrcr.20273` (visited on 01/11/2021) (Cited on pages 22 and 30).

G. Marsily (1984). 'Spatial Variability of Properties in Porous Media: A Stochastic Approach'. In: *Fundamentals of Transport Phenomena in Porous Media*. Ed. by J. Bear and M. Y. Corapcioglu. Dordrecht: Springer Netherlands, pp. 719–769. ISBN: 978-94-009-6177-7 978-94-009-6175-3. DOI: `10.1007/978-94-009-6175-3_15`. URL: `http://link.springer.com/10.1007/978-94-009-6175-3_15` (visited on 07/11/2021) (Cited on page 12).

Y. Marzouk, T. Moselhy, M. Parno and A. Spantini (2016). 'Sampling via Measure Transport: An Introduction'. In: *Handbook of Uncertainty Quantification*. Ed. by R. Ghanem, D. Higdon and H. Owhadi. Cham: Springer International Publishing, pp. 1–41. ISBN: 978-3-319-11259-6. DOI: `10.1007/978-3-319-11259-6_23-1`. URL: `https://doi.org/10.1007/978-3-319-11259-6_23-1` (Cited on page 20).

Y. M. Marzouk and H. N. Najm (Apr. 2009). 'Dimensionality reduction and polynomial chaos acceleration of Bayesian inference in inverse problems'. en. In: *Journal of Computational Physics* 228.6, pp. 1862–1902. ISSN: 00219991. DOI: `10.1016/j.jcp.2008.11.024`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S0021999108006062` (visited on 16/03/2020) (Cited on page 35).

Y. M. Marzouk, H. N. Najm and L. A. Rahn (June 2007). 'Stochastic spectral methods for efficient Bayesian solution of inverse problems'. en. In: *Journal of Computational Physics* 224.2, pp. 560–586. ISSN: 00219991. DOI: `10.1016/j.`

jcp.2006.10.010. URL: https://linkinghub.elsevier.com/retrieve/pii/S0021999106004839 (visited on 11/11/2021) (Cited on page 35).

N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller and E. Teller (June 1953). 'Equation of State Calculations by Fast Computing Machines'. en. In: *The Journal of Chemical Physics* 21.6, pp. 1087–1092. ISSN: 0021-9606, 1089-7690. DOI: 10.1063/1.1699114. URL: http://aip.scitation.org/doi/10.1063/1.1699114 (visited on 04/02/2019) (Cited on page 16).

A. Mondal, Y. Efendiev, B. Mallick and A. Datta-Gupta (Mar. 2010). 'Bayesian uncertainty quantification for flows in heterogeneous porous media using reversible jump Markov chain Monte Carlo methods'. en. In: *Advances in Water Resources* 33.3, pp. 241–256. ISSN: 03091708. DOI: 10.1016/j.advwatres.2009.10.010. URL: https://linkinghub.elsevier.com/retrieve/pii/S0309170809001729 (visited on 19/02/2020) (Cited on pages 28 and 29).

P. Moritz, R. Nishihara, S. Wang, A. Tumanov, R. Liaw, E. Liang, M. Elibol, Z. Yang, W. Paul, M. I. Jordan et al. (Oct. 2018). 'Ray: A Distributed Framework for Emerging AI Applications'. In: *13th USENIX Symposium on Operating Systems Design and Implementation (OSDI 18)*. Carlsbad, CA: USENIX Association, pp. 561–577. ISBN: 978-1-939133-08-3. URL: https://www.usenix.org/conference/osdi18/presentation/moritz (Cited on page 127).

R. W. Nelson (1960). 'In-place measurement of permeability in heterogeneous media: 1. Theory of a proposed method'. en. In: *Journal of Geophysical Research (1896-1977)* 65.6, pp. 1753–1758. ISSN: 2156-2202. DOI: 10.1029/JZ065i006p01753. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/JZ065i006p01753 (visited on 07/02/2019) (Cited on page 11).

S. P. Neuman (1973). 'Calibration of distributed parameter groundwater flow models viewed as a multiple-objective decision process under uncertainty'. en. In: *Water Resources Research* 9.4, pp. 1006–1021. ISSN: 1944-7973. DOI: 10.1029/WR009i004p01006. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/WR009i004p01006 (visited on 07/02/2019) (Cited on page 11).

D. S. Oliver, L. B. Cunha and A. C. Reynolds (Mar. 1997). ‘Markov chain Monte Carlo methods for conditioning a permeability field to pressure data’. en. In: *Mathematical Geology* 29.1, pp. 61–91. ISSN: 0882-8121, 1573-8868. DOI: 10.1007/BF02769620. URL: http://link.springer.com/10.1007/BF02769620 (visited on 04/02/2019) (Cited on page 28).

M. Parno, A. Davis, L. Seelinger and Y. Marzouk (2014). *MIT uncertainty quantification (MUQ) library* (Cited on page 131).

M. Parno and Y. Marzouk (Jan. 2018). ‘Transport map accelerated Markov chain Monte Carlo’. In: *SIAM/ASA Journal on Uncertainty Quantification* 6.2. arXiv: 1412.5492, pp. 645–682. ISSN: 2166-2525. DOI: 10.1137/17M1134640. URL: http://arxiv.org/abs/1412.5492 (visited on 19/05/2021) (Cited on page 20).

A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga et al. (2019). ‘PyTorch: An Imperative Style, High-Performance Deep Learning Library’. In: *Advances in Neural Information Processing Systems 32*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox and R. Garnett. Curran Associates, Inc., pp. 8024–8035. URL: http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf (Cited on page 134).

M. Plummer (2003). ‘JAGS: A program for analysis of Bayesian graphical models using Gibbs sampling’. en. In: *Working Papers*, p. 8 (Cited on page 17).

C. Rasmussen (2003). ‘Gaussian Processes to Speed up Hybrid Monte Carlo for Expensive Bayesian Integrals’. In: *Bayesian Statistics 7*. Backup Publisher: Max-Planck-Gesellschaft. Biologische Kybernetik, pp. 651–659 (Cited on page 20).

G. O. Roberts and J. S. Rosenthal (Feb. 1998). ‘Optimal scaling of discrete approximations to Langevin diffusions’. en. In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 60.1, pp. 255–268. ISSN: 1369-7412, 1467-9868. DOI: 10.1111/1467-9868.00123. URL: http://doi.wiley.com/10.1111/1467-9868.00123 (visited on 23/09/2019) (Cited on page 21).

G. O. Roberts and J. S. Rosenthal (2007). ‘Coupling and Ergodicity of Adaptive Markov Chain Monte Carlo Algorithms’. In: *Journal of Applied Probability*

44.2. Publisher: Applied Probability Trust, pp. 458–475. ISSN: 0021-9002. URL: https://www.jstor.org/stable/27595854 (visited on 10/04/2021) (Cited on page 19).

G. O. Roberts and J. S. Rosenthal (Jan. 2009). 'Examples of Adaptive MCMC'. en. In: *Journal of Computational and Graphical Statistics* 18.2, pp. 349–367. ISSN: 1061-8600, 1537-2715. DOI: 10.1198/jcgs.2009.06134. URL: http://www.tandfonline.com/doi/abs/10.1198/jcgs.2009.06134 (visited on 02/06/2020) (Cited on pages 8 and 19).

G. O. Roberts and R. L. Tweedie (Dec. 1996). 'Exponential Convergence of Langevin Distributions and Their Discrete Approximations'. en. In: *Bernoulli* 2.4, p. 341. ISSN: 13507265. DOI: 10.2307/3318418. URL: https://www.jstor.org/stable/3318418?origin=crossref (visited on 08/03/2020) (Cited on pages 21 and 129).

G. Rossum (1995). *Python Reference Manual.* Tech. rep. Amsterdam, The Netherlands, The Netherlands: CWI (Centre for Mathematics and Computer Science) (Cited on page 127).

Y. Rubin, X. Chen, H. Murakami and M. Hahn (2010). 'A Bayesian approach for inverse modeling, data assimilation, and conditional simulation of spatial random fields'. en. In: *Water Resources Research* 46.10. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1029/2009WR008799. ISSN: 1944-7973. DOI: 10.1029/2009WR008799. URL: https://onlinelibrary.wiley.com/doi/abs/10.1029/2009WR008799 (visited on 07/11/2021) (Cited on page 14).

C. Rücker, T. Günther and F. M. Wagner (2017). 'pyGIMLi: An open-source library for modelling and inversion in geophysics'. In: *Computers and Geosciences* 109, pp. 106–123. DOI: 10.1016/j.cageo.2017.07.011. URL: https://www.sciencedirect.com/science/article/pii/S0098300417300584 (Cited on page 132).

J. Salvatier, T. V. Wiecki and C. Fonnesbeck (Apr. 2016). 'Probabilistic programming in Python using PyMC3'. en. In: *PeerJ Computer Science* 2, e55. ISSN: 2376-5992. DOI: 10.7717/peerj-cs.55. URL: https://peerj.com/articles/cs-55 (visited on 30/10/2021) (Cited on page 17).

L. Seelinger, A. Reinarz, L. Rannabauer, M. Bader, P. Bastian and R. Scheichl (July 2021). ‘High Performance Uncertainty Quantification with Parallelized Multilevel Markov Chain Monte Carlo’. In: *arXiv:2107.14552 [cs, math]*. arXiv: 2107.14552. URL: `http://arxiv.org/abs/2107.14552` (visited on 03/11/2021) (Cited on page 26).

D. Sejdinovic, H. Strathmann, M. L. Garcia, C. Andrieu and A. Gretton (June 2014). ‘Kernel Adaptive Metropolis-Hastings’. In: *Proceedings of the 31st International Conference on Machine Learning*. Ed. by E. P. Xing and T. Jebara. Vol. 32. Proceedings of Machine Learning Research. Issue: 2. Bejing, China: PMLR, pp. 1665–1673. URL: `https://proceedings.mlr.press/v32/sejdinovic14.html` (Cited on page 19).

M. Shafii, B. Tolson and L. S. Matott (Aug. 2014). ‘Uncertainty-based multi-criteria calibration of rainfall-runoff models: a comparative study’. en. In: *Stochastic Environmental Research and Risk Assessment* 28.6, pp. 1493–1510. ISSN: 1436-3240, 1436-3259. DOI: `10.1007/s00477-014-0855-x`. URL: `http://link.springer.com/10.1007/s00477-014-0855-x` (visited on 01/11/2021) (Cited on page 22).

C. Sherlock, A. Golightly and D. A. Henderson (Sept. 2015). ‘Adaptive, delayed-acceptance MCMC for targets with expensive likelihoods’. en. In: *arXiv:1509.00172 [math, stat]*. arXiv: 1509.00172. URL: `http://arxiv.org/abs/1509.00172` (visited on 28/08/2019) (Cited on page 24).

M. C. Shewry and H. P. Wynn (Jan. 1987). ‘Maximum entropy sampling’. en. In: *Journal of Applied Statistics* 14.2, pp. 165–170. ISSN: 0266-4763, 1360-0532. DOI: `10.1080/02664768700000020`. URL: `https://www.tandfonline.com/doi/full/10.1080/02664768700000020` (visited on 06/10/2021) (Cited on page 136).

R. Skibba (Mar. 2018). ‘Einstein, Bohr and the war over quantum theory’. en. In: *Nature* 555.7698, pp. 582–584. ISSN: 0028-0836, 1476-4687. DOI: `10.1038/d41586-018-03793-2`. URL: `http://www.nature.com/articles/d41586-018-03793-2` (visited on 19/01/2022) (Cited on page 138).

R. W. Stallman (1956). 'Numerical analysis of regional water levels to define aquifer hydrology'. en. In: *Transactions, American Geophysical Union* 37.4, p. 451. ISSN: 0002-8606. DOI: 10.1029/TR037i004p00451. URL: http://doi.wiley.com/10.1029/TR037i004p00451 (visited on 07/02/2019) (Cited on page 11).

H. Strathmann, D. Sejdinovic, S. Livingstone, Z. Szabo and A. Gretton (2015). 'Gradient-free Hamiltonian Monte Carlo with Efficient Kernel Exponential Families'. In: *Advances in Neural Information Processing Systems*. Ed. by C. Cortes, N. Lawrence, D. Lee, M. Sugiyama and R. Garnett. Vol. 28. Curran Associates, Inc. URL: https://proceedings.neurips.cc/paper/2015/file/8ebda540cbcc4d7336496819a46a1b68-Paper.pdf (Cited on page 19).

S. Strebelle (2002). 'Conditional Simulation of Complex Geological Structures Using Multiple-Point Statistics'. en. In: *Mathematical Geology*, p. 21 (Cited on page 32).

C. J. F. ter Braak (Sept. 2006). 'A Markov Chain Monte Carlo version of the genetic algorithm Differential Evolution: easy Bayesian computing for real parameter spaces'. en. In: *Statistics and Computing* 16.3, pp. 239–249. ISSN: 0960-3174, 1573-1375. DOI: 10.1007/s11222-006-8769-1. URL: http://link.springer.com/10.1007/s11222-006-8769-1 (visited on 03/02/2021) (Cited on pages 22 and 35).

C. J. F. ter Braak and J. A. Vrugt (Dec. 2008). 'Differential Evolution Markov Chain with snooker updater and fewer chains'. en. In: *Statistics and Computing* 18.4, pp. 435–446. ISSN: 0960-3174, 1573-1375. DOI: 10.1007/s11222-008-9104-9. URL: http://link.springer.com/10.1007/s11222-008-9104-9 (visited on 03/02/2021) (Cited on pages 22 and 23).

M. Tonkin and J. Doherty (2009). 'Calibration-constrained Monte Carlo analysis of highly parameterized models using subspace techniques'. en. In: *Water Resources Research* 45.12. ISSN: 1944-7973. DOI: 10.1029/2007WR006678. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2007WR006678 (visited on 28/12/2018) (Cited on pages 15, 21 and 35).

M. J. Tonkin and J. Doherty (Oct. 2005). 'A hybrid regularized inversion methodology for highly parameterized environmental models: HYBRID REGULARIZATION

METHODOLOGY'. en. In: *Water Resources Research* 41.10. ISSN: 00431397. DOI: 10.1029/2005WR003995. URL: http://doi.wiley.com/10.1029/2005WR003995 (visited on 23/08/2019) (Cited on pages 14 and 31).

A. Vehtari, A. Gelman, D. Simpson, B. Carpenter and P.-C. Bürkner (May 2020). 'Rank-normalization, folding, and localization: An improved $\widehat{R}$ for assessing convergence of MCMC'. In: *arXiv:1903.08008 [stat]*. arXiv: 1903.08008. DOI: 10.1214/20-BA1221. URL: http://arxiv.org/abs/1903.08008 (visited on 20/11/2020) (Cited on page 31).

P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright et al. (2020). 'SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python'. In: *Nature Methods* 17, pp. 261–272. DOI: 10.1038/s41592-019-0686-2 (Cited on page 127).

C. R. Vogel (2002). *Computational methods for inverse problems*. Frontiers in applied mathematics. Philadelphia: Society for Industrial and Applied Mathematics. ISBN: 978-0-89871-507-1 (Cited on page 15).

J. A. Vrugt, C. ter Braak, C. Diks, B. A. Robinson, J. M. Hyman and D. Higdon (Jan. 2009). 'Accelerating Markov Chain Monte Carlo Simulation by Differential Evolution with Self-Adaptive Randomized Subspace Sampling'. en. In: *International Journal of Nonlinear Sciences and Numerical Simulation* 10.3. ISSN: 2191-0294, 1565-1339. DOI: 10.1515/IJNSNS.2009.10.3.273. URL: https://www.degruyter.com/document/doi/10.1515/IJNSNS.2009.10.3.273/html (visited on 14/03/2021) (Cited on pages 23 and 29).

J. A. Vrugt (Jan. 2016). 'Markov chain Monte Carlo simulation using the DREAM software package: Theory, concepts, and MATLAB implementation'. en. In: *Environmental Modelling & Software* 75, pp. 273–316. ISSN: 13648152. DOI: 10.1016/j.envsoft.2015.08.013. URL: https://linkinghub.elsevier.com/retrieve/pii/S1364815215300396 (visited on 14/03/2021) (Cited on pages 22 and 132).

M. J. Wainwright and M. I. Jordan (2007). 'Graphical Models, Exponential Families, and Variational Inference'. en. In: *Foundations and Trends® in Machine Learning*

1.1–2, pp. 1–305. ISSN: 1935-8237, 1935-8245. DOI: 10.1561/2200000001. URL: http://www.nowpublishers.com/article/Details/MAL-001 (visited on 07/12/2021) (Cited on page 9).

C. Zhang, B. Shahbaba and H. Zhao (Nov. 2017). 'Hamiltonian Monte Carlo acceleration using surrogate functions with random bases'. en. In: *Statistics and Computing* 27.6, pp. 1473–1490. ISSN: 0960-3174, 1573-1375. DOI: 10.1007/s11222-016-9699-1. URL: http://link.springer.com/10.1007/s11222-016-9699-1 (visited on 04/11/2021) (Cited on page 19).

X. Zhang and A. Curtis (2020). 'Seismic Tomography Using Variational Inference Methods'. In: *Journal of Geophysical Research: Solid Earth* 125.4. e2019JB018589 10.1029/2019JB018589, e2019JB018589. DOI: https://doi.org/10.1029/2019JB018589. eprint: https://agupubs.onlinelibrary.wiley.com/doi/pdf/10.1029/2019JB018589. URL: https://agupubs.onlinelibrary.wiley.com/doi/abs/10.1029/2019JB018589 (Cited on page 9).

H. Zhou, J. J. Gómez-Hernández and L. Li (Jan. 2014). 'Inverse methods in hydrogeology: Evolution and recent trends'. en. In: *Advances in Water Resources* 63, pp. 22–37. ISSN: 03091708. DOI: 10.1016/j.advwatres.2013.10.014. URL: https://linkinghub.elsevier.com/retrieve/pii/S0309170813002017 (visited on 28/12/2018) (Cited on page 11).

Q. Zhou, Z. Hu, Z. Yao and J. Li (Jan. 2017). 'A Hybrid Adaptive MCMC Algorithm in Function Spaces'. en. In: *SIAM/ASA Journal on Uncertainty Quantification* 5.1, pp. 621–639. ISSN: 2166-2525. DOI: 10.1137/16M1082950. URL: https://epubs.siam.org/doi/10.1137/16M1082950 (visited on 22/03/2021) (Cited on page 22).