Hamiltonian Monte Carlo solution of tomographic inverse problems

Fichtner, Andreas; Zunino, Andrea; Gebraad, Lars

Published in:
Geophysical Journal International

DOI:
10.1093/gji/ggy496

Publication date:
2019

Document version
Publisher's PDF, also known as Version of record

Citation for published version (APA):
Hamiltonian Monte Carlo solution of tomographic inverse problems

Andreas Fichtner,1 Andrea Zunino2 and Lars Gebraad1

1Department of Earth Sciences, ETH Zurich, 8092 Zurich, Switzerland. E-mail: andreas.fichtner@erdw.ethz.ch
2Niels Bohr Institute, University of Copenhagen, 2100 Copenhagen, Denmark

Accepted 2018 November 21. Received 2018 September 28; in original form 2018 July 3

SUMMARY
We present the theory for and applications of Hamiltonian Monte Carlo (HMC) solutions of linear and nonlinear tomographic problems. HMC rests on the construction of an artificial Hamiltonian system where a model is treated as a high-dimensional particle moving along a trajectory in an extended model space. Using derivatives of the forward equations, HMC is able to make long-distance moves from the current towards a new independent model, thereby promoting model independence, while maintaining high acceptance rates. Following a brief introduction to HMC using common geophysical terminology, we study linear (tomographic) problems. Though these may not be the main target of Monte Carlo methods, they provide valuable insight into the geometry and the tuning of HMC, including the design of suitable mass matrices and the length of Hamiltonian trajectories. This is complemented by a self-contained proof of the HMC algorithm in Appendix A. A series of tomographic/imaging examples is intended to illustrate (i) different variants of HMC, such as constrained and tempered sampling, (ii) the independence of samples produced by the HMC algorithm and (iii) the effects of tuning on the number of samples required to achieve practically useful convergence. Most importantly, we demonstrate the combination of HMC with adjoint techniques. This allows us to solve a fully nonlinear, probabilistic traveltime tomography with several thousand unknowns on a standard laptop computer, without any need for supercomputing resources.

Key words: Inverse theory; Numerical solutions; Probability distributions; Statistical methods; Seismic tomography.

1 INTRODUCTION

The inference of physical properties that are not directly observable, such as elastic moduli inside the Earth or future atmospheric conditions, is the central theme of geophysics. It has defined the science since its entry into the quantitative era, when instrumental observations and predictive mathematical theories became available (e.g. Mallet 1861; von Rebeur-Paschwitz 1889; Oldham 1906). Pioneering works, for instance by Tikhonov (1963) and Backus & Gilbert (1968, 1970) provided an early formalization of indirect (geophysical) inference, and a starting point for the field of inverse theory as a science in its own right.

1.1 Bayesian inference and model space sampling

Geophysical inversion in general and tomography in particular is a delicate balancing act between elegant universality and computational pragmatism. Sparse and erroneous data combined with imperfections of the forward modelling theory make inverse problems ill-posed in the sense that an ensemble of plausible models explains the data to within their observational uncertainties. Solving an inverse problem means to explore and characterize the ensemble (Backus & Gilbert 1968), which can be achieved most comprehensively by computing the posterior probability density in model space using Bayes’ theorem (Bayes & Price 1763; Jaynes 2003; Tarantola 2005; Sen & Stoffa 2013). While being beautifully simple, Bayes’ theorem mostly requires sampling of the model space to approximate the posterior. Depending on its dimension and the numerical cost of solving the forward problem, sampling can be prohibitively expensive. Therefore, the majority of inverse problems are simplified. Possible simplifications include linearization of the forward equations and the assumption of Gaussian priors. Alternatively, the task may be reduced to the estimation of the maximum-likelihood model and the posterior covariances, which may, however, not represent the complete plausible ensemble. In both cases, possible consequences are the interpretation of meaningless models, and unrealistic uncertainties that hamper decision making and the comparison with earlier results.
Since the early work of Keilis-Borok & Yanovskaya (1967) and Press (1968), the number of (nonlinear) geophysical inverse problems that can be solved through Bayesian inference has grown continuously, thanks to increasing computational power and the development of advanced sampling strategies that improve the classical Metropolis–Hastings algorithm (Metropolis et al. 1953; Hastings 1970; Mosègaard & Tarantola 1995). The latter include parallel tempering (e.g. Geyer 1991; Marinari & Parisi 1992; Geyer & Thompson 1995; Sambridge 2014), the neighbourhood algorithm (Sambridge 1999a,b), the reversible-jump algorithm used for trans-dimensional inversion (e.g. Green 1995; Sambridge et al. 2006, 2013; Bodin & Sambridge 2009) and principal-component rotations that help to decouple correlated model parameters (e.g. Dettmers et al. 2008).

A landmark in the development of sampling methods was the recognition, formalized in a series of ‘No-Free-Lunch theorems’ (e.g. Wolpert & Macready 1997; Mosègaard 2012), that all algorithms are from the outset equally inefficient. Superiority of an algorithm can result only from the injection of prior knowledge in a broad sense. This may include, for instance, knowledge on the nature of the forward problem or the likely topology of the misfit surface, which may enable specific tuning strategies. Conversely, specific algorithms may be developed to work efficiently for inverse problems that share certain properties.

Within this context, Hamiltonian Monte Carlo (HMC) was developed for problems where the derivative of the target probability density with respect to the model parameters can be computed quickly (e.g. Neal 2011; Betancourt 2017). Using derivatives helps to focus on those regions of model space that are plausible, thus wasting less samples. Since its first formulation for applications in lattice quantum chromodynamics (Duane et al. 1987), HMC has found widespread use in neural networks and machine learning (Neal 1996; Bishop 2006; Gutmann & Hyvärinen 2012), molecular simulations (e.g. Dukkadam et al. 2016), nuclear physics (e.g. Elhatisari et al. 2015), genomics (e.g. Honkela et al. 2015), signal processing (e.g. Wei et al. 2015) and quantum mechanics (e.g. Seah et al. 2015).

Numerous geophysical inverse problems for which first derivatives are easy to compute also fall within the range of applicability of HMC. These include, for instance, gravity inversion (e.g. Li & Oldenburg 1998; Blakey 2002), seismic source inversion (e.g. Dziewoński et al. 1981; Ekström et al. 2012; Lee et al. 2014) and potential fields inversions (e.g. Jackson et al. 2000; Blakey 2002). Nevertheless, the potential of HMC for geophysical inversion is only slowly being recognized (Muir & Tkáč 2015; Biswas & Sen 2017; Sen & Biswas 2017; Fichtner & Simutė 2018).

HMC, first introduced as hybrid Monte Carlo (Duane et al. 1987), may indeed be considered a hybrid between two end-member approaches: (1) gradient-based optimization that is efficient in finding an optimum but provides only limited uncertainty information and (2) derivative-free Markov chain Monte Carlo (MCMC) methods where potentially useful derivative information is disregarded. HMC attempts to use the best of both.

1.2 Objectives and outline

The primary goal of this manuscript is to introduce HMC as a tool specifically for the solution of tomographic inverse problems, typically characterized by weak nonlinearity. First-arrival travelt ime tomography based on ray theory is quasi-linear by virtue of Fermat’s principle, to the extent that the problem is often linearized (e.g. Aki et al. 1976; Iyer & Hirahara 1993; Nolet 2008). For finite-frequency tomography based on measurements of cross-correlation time-shifts (e.g. Dahlen et al. 2000; Tian et al. 2009), quasi-linearity beyond a π/2 phase shift was shown numerically by Mercerat & Nolet (2013). Similarly, for full-waveform inversion, numerous authors proposed misfit measures that promote linearity by explicitly extracting phase information (e.g. Luo & Schuster 1991; Gee & Jordan 1992; Fichtner et al. 2008; van Leeuwen & Mulder 2010; Rickers et al. 2012). Furthermore, derivatives of the forward problem can be computed efficiently using adjoint techniques (e.g. Tarantola 1986; Tromp et al. 2005; Fichtner et al. 2006; Chen et al. 2007; Liu & Tromp 2008).

The advantage of limiting our attention to weakly nonlinear (tomographic) problems is our ability to derive efficient tuning strategies for such cases. Though HMC has in theory no limitation to a specific class of problems, apart from requiring differentiability, its performance may not always be competitive.

This manuscript is organized as follows: Section 2 is intended to motivate the development of HMC and to describe the basic concept. This is complemented by Appendix A, where we provide a detailed and self-contained proof of the algorithm. In Section 3, we analyse linear problems in order to gain intuition and to derive tuning strategies that will also be useful in nonlinear tomography. The linear and nonlinear inversion examples in Section 4 illustrate different variants of HMC, the importance of tuning and the combination of HMC with adjoint techniques. Finally, the discussion in Section 5 is focused on the potentials and limitations of HMC, also in the context of the No-Free-Lunch Theorem.

2 HAMILTONIAN MONTE CARLO SOLUTION OF GENERAL INVERSE PROBLEMS

To set the stage for the description of HMC and to introduce basic notation, we start with a brief recapitulation of Bayesian inference. This will be followed by a motivation for and a description of the HMC algorithm.
2.1 Bayesian inference and Markov-chain Monte Carlo

Bayesian inference describes information on model parameters \( m \in M \) and data \( d \in D \) in terms of probability density functions (pdfs) \( \rho \) (e.g. Tarantola & Valette 1982; Tarantola 2005), where \( M \) and \( D \) denote the model and data spaces, respectively. Prior information in model space, \( \rho_0(m) \), captures all constraints on model parameters available prior to the analysis of any data, and the prior in data space, \( \rho_d(d|m) \), is the likelihood of a computed data vector \( d \) given some \( m \). The data space prior or likelihood function thus contains the forward problem equations, linking the model and data spaces. Based on the priors \( \rho_0(m) \) and \( \rho_d(d|m) \), Bayes’ theorem gives the posterior pdf in model space, \( \rho_m(m|d) \), which encapsulates all available information on the model parameters that can be deduced from prior knowledge,

\[
\rho_m(m|d) = k^{-1} \rho_0(d|m) \rho_0(m).
\]

The constant \( k \), known as the evidence (Sambridge et al. 2006), normalizes \( \rho_m(m|d) \) so that \( \int_M \rho_m(m|d) \, dm = 1 \). Ideally, we are interested in the potentially high-dimensional posterior. However, since we lack tools to interrogate and interpret this, we mostly focus on lower-dimensional quantities computed from \( \rho_m(m|d) \) via an integral over (parts of) model space. These may include, for instance, the probability \( P(m \in M_{\text{sub}}) = \int_{M_{\text{sub}}} \rho_m(m|d) \, dm \) of \( m \) being inside a model space subvolume \( M_{\text{sub}} \subset M \), the expectation \( \mathbb{E}(m) = \int_M m \rho_m(m|d) \, dm \) or higher moments and marginals.

Unless the forward problem is linear and the priors are Gaussian, an analytical treatment of the integrals can be either difficult or entirely impossible, thus motivating Markov chain Monte Carlo (MCMC) methods that perform a random walk through model space (e.g. Sen & Stoffa 2013). The sampling density of MCMC is designed to be proportional to the posterior \( \rho_m(m|d) \), and may thus be used to approximate the integrals. For independent samples, the approximation error of MCMC is proportional to \( 1/\sqrt{N} \), where \( N \) is the total number of samples (MacKay 2003). When samples are correlated, convergence is typically slower. One of the main goals of HMC is thus to produce largely independent samples.

2.2 Motivating example

To motivate the development and application of HMC, we consider the simple example of a 10-D model vector \( m \) and a Gaussian model space prior

\[
\rho_0(m) = \text{const.} \exp\left( -\frac{1}{2} m^T C^{-1}_M m \right).
\]

Furthermore, we let the forward problem be linear, \( d = Gm \), where we define \( G \) to be a diagonal matrix with entries \( G_{ii} = i/10 \). The data vector components are set to \( d_{\text{obs}} = i/5 \). Assuming Gaussian measurement errors, we obtain the likelihood function

\[
\rho_d(d|m) = \text{const.} \exp\left( -\frac{1}{2} (d^{\text{obs}} - Gm)^T C^{-1}_D (d^{\text{obs}} - Gm) \right).
\]

For simplicity, we (arbitrarily) set the prior model and data covariance matrices to \( C_M = \sigma_M^2 I \) and \( C_D = I \), respectively. While the posterior mean and covariance could easily be found analytically for this example, we employ the extended Metropolis–Hastings algorithm (EMH) of Mosegaard & Tarantola (1995) for the purpose of illustration. EMH is a special variant of the general Metropolis–Hastings method (Metropolis et al. 1953; Hastings 1970), which exploits the fact that the posterior is a product of two probability densities and consequently uses the prior as a proposal, such that only the likelihood function has to be evaluated. Hence, the performance of EMH depends on the extent to which the prior already approximates the posterior. Running EMH for \( \sigma_M = 2 \) with 500 000 samples produces the histograms of the posterior in Fig. 1. As expected, \( m_1 \) is less constrained than \( m_{10} \), because the small \( G_{11} \) allows for large variations in \( m_1 \) that still fall within the measurement error.

The advantage of EMH, which draws proposals from \( \rho_0 \), lies in its ability to account for prior knowledge, the quality of which critically affects the efficiency of the algorithm. When the prior is already a good approximation of the posterior, sampling is very efficient, and a good approximation of the posterior can be obtained with a small number of samples, and vice versa. The magnitude of this effect is illustrated in Fig. 2, which uses the toy problem of Section 2.2. The left-hand panel shows the trajectory of the walker projected onto the \( (m_1, m_{10}) \) plane when the standard deviation of the prior is \( \sigma_M = 1 \). Each red dot represents an accepted test model. There are 60 176 of these dots, meaning that around 12 per cent of the 500 000 proposed samples have been accepted. This is significantly below the 30 per cent acceptance rate that around 12 per cent of the 500 000 proposed samples have been accepted. This undesirable phenomenon is a consequence of the curse of dimensionality (e.g. Tarantola 2005). In a high-dimensional model space, the volume of relevant models defined by the posterior becomes excessively small compared to the volume of the whole model space. If the proposal density, that is the prior in the case of EMH, does not limit the search space very significantly, the probability of finding a new model that is better than an already relatively good one becomes negligibly small. Therefore, the walker will mostly stay at its current position.
2.3 The Hamiltonian Monte Carlo algorithm

HMC is designed to reduce problems related to dependent samples and low acceptance rates. Intuitively, the HMC concept treats a model $m$ as the mechanical analogue of a particle that moves from its current position (current model) to a new position (proposed model) along some trajectory. The geometry of the trajectory is controlled by the misfit that is interpreted as potential energy $U$, as well as the kinetic energy $K$ and mass $M$ of the particle, both of which are artificially introduced auxiliary quantities.

In our description of the HMC algorithm, we closely follow Neal (2011) and Betancourt (2017). Starting from the current model $m$, the walker follows a Hamiltonian trajectory towards a new test model $m'$. The trajectory of the model as a function of an artificially introduced time variable $t$ is controlled by Hamilton’s equations (e.g. Symon 1971; Landau & Lifshitz 1976):

$$\frac{dm_i}{dt} = \frac{\partial H}{\partial p_i}, \quad \frac{dp_i}{dt} = -\frac{\partial H}{\partial m_i},$$

(4)

with $i = 1, \ldots, N$. In Hamiltonian dynamics, $m_i$ plays the role of a generalized position and $p_i$ is a generalized momentum defined through

$$p_i = \sum_{j=1}^N M_{ij} \frac{dm_j}{dt},$$

(5)
with a symmetric and positive definite mass matrix $M_i$. The Hamiltonian $H(m, p)$ is the total energy of the particle, that is $H = K + U$ with the kinetic energy $K(m, p)$ and the potential energy $U(m, p)$. The momenta $p$ are elements of momentum space $P$. Together, $m$ and $p$ are in phase space $X = M \times P$.

The basic concept of HMC is to sample an auxiliary distribution in the $2n$-dimensional phase space, the canonical distribution, from which samples of the posterior can be obtained by ignoring (or marginalizing over) the momentum space component. Defining kinetic and potential energies as

$$K(p) = \frac{1}{2} \sum_{i,j=1}^n p_i M_{ij}^{-1} p_j,$$

and

$$U(m) = -\log \rho_m(m|d),$$

the canonical distribution is defined as

$$\rho_c(m, p) = \exp[-H(m, p)] = \exp[-U(m) - K(p)] = \rho_m(m|d) \exp \left(-\frac{1}{2} \sum_{i,j=1}^n p_i M_{ij}^{-1} p_j \right).$$

In other words, the canonical distribution is the product of the posterior $\rho_m(m|d)$ and a Gaussian distribution of the momenta $p$. If we manage to sample $\rho_c(m, p)$, we may simply take the model space component of the samples to obtain samples of the posterior.

The step of a random walker in HMC consists of two phases: first, momenta $p_i$ are drawn from their Gaussian distribution

$$\rho_p(p) = \text{const.} \exp \left(-\frac{1}{2} \sum_{i,j=1}^n p_i M_{ij}^{-1} p_j \right),$$

as previously defined in eq. (8). Second, the current state $(m, p)$ is propagated for some time $\tau$ along a Hamiltonian trajectory by numerically solving Hamilton’s equation (4). The new state $(m', p')$ is then accepted with probability

$$\min \left(1, \frac{\rho_c(m', p')}{\rho_c(m, p)} \right) = \min \left(1, \frac{\rho_m(m', d)}{\rho_m(m, d)} \exp \left(-K(p') + K(p)\right) \right) = \min \left(1, \exp \left[-K(p') - K(p)\right] \right),$$

which can be seen as a modified version of the Metropolis rule. If the transition is accepted, the new model $(m', p')$ is counted; otherwise, the original model $(m, p)$. Then, the walker’s next step starts again with the first phase, and the procedure is repeated. To obtain samples of the posterior, the momentum component of the phase space samples is simply ignored (or marginalized over). A self-contained proof of this algorithm is provided in Appendix A.

### 2.4 Numerical integration

Since Hamiltonian systems conserve total energy, that is, $H(p_i) = H(p)$, eq. (10) implies that all proposals would be accepted if we were able to solve Hamilton's equations analytically. In practice, symplectic numerical integrators must be employed to maintain the properties of Hamiltonian systems upon which HMC relies critically. As shown in the Appendix A, these are time reversibility, phase space partitioning and volume preservation. In the context of HMC, the most frequently used symplectic integrator is the leapfrog method with time step $\Delta t$:

$$p_i(t + \Delta t/2) = p_i(t) - \frac{1}{2} \Delta t \frac{dU}{dm_i} \bigg|_{t+\Delta t/2}.$$

$$m_i(t + \Delta t) = m_i(t) + \Delta t \frac{dH}{dp_i} \bigg|_{t+\Delta t/2}.$$

$$p_i(t + \Delta t) = p_i(t + \Delta t/2) - \frac{1}{2} \Delta t \frac{dU}{dp_i} \bigg|_{t+\Delta t}. $$

Though alternative (symplectic) integrators are well documented in the literature (e.g. de Vogelaere 1956; Ruth 1983; Leimkuhler & Reich 1994; Sanz-Serna & Calvo 1994; Blanes et al. 2014), we consistently employ the leapfrog scheme in our numerical examples, mostly for simplicity and ease of implementation. As many other symplectic integrators, leapfrog does not preserve $H$ exactly. As a consequence, the acceptance rate of HMC typically differs from the theoretical ideal, which is 1.

### 3 ANALYSIS OF LINEAR (TOMOGRAPHIC) INVERSE PROBLEMS

Following the description of the general HMC algorithm, we proceed in Section 3.1 with the analysis of linear (tomographic) systems (e.g. Iyer & Hirahara 1993; Nolte 2008). Based on this analysis, Section 3.2 will focus on the role of the so far undetermined mass matrix in tuning and (implicitly) regularizing HMC.
Using the matrix sine and cosine, the general solution of eq. (19) is given by

\[ m(t) = \sin(Wt) \, a + \cos(Wt) \, b + (W^2)^{-1} \, f, \]

with constant vectors \( a \) and \( b \) determined by the initial position and momentum of the trajectory. An analogous equation can be obtained for the momentum vector \( p \). Eq. (20) shows that Hamiltonian trajectories for a linear problem with Gaussian uncertainties are harmonic oscillations, the frequencies of which are controlled by the matrix \( W \).

In the context of a linear inverse problem, the matrix-valued frequency \( W \) can be interpreted in terms of the posterior covariance matrix, \( \tilde{C}_M = (G^T C_D^{-1} G + C_M^{-1})^{-1/2} \), corresponding to the Gaussian priors defined in eqs (2) and (3) (Tarantola 2005). Thus, we find \( W = (M \tilde{C}_M)^{-1/2} \), meaning that comparatively small posterior covariances result in comparatively fast oscillations, and vice versa. The mass matrix \( M \) can serve as tuning parameter to adjust the speed with which the phase space is traversed, as we will discuss in Section 3.2.

To make this more concrete, we reuse the example from Section 2.2 with unit covariances \( C_M = I \) and \( C_D = I \). For illustration, we choose a unit mass matrix, \( M = I \). Fig. 3 shows a random realization of a Hamiltonian trajectory through the 10-D model space projected onto Figure 3. Realization of a 10-D Hamiltonian trajectory projected onto the 2-D planes \( (m_1, m_{10}), (m_1, m_6) \) and \( (m_1, m_2) \). Model parameters with small forward problem derivative \( G_m \), such as \( m_1 \), oscillate slowly with large amplitude, and vice versa.
Figure 4. HMC applied to the example problem from Section 2.2 and Fig. 2. The random walker’s trajectory is shown in the left-hand panel. Out of 10,000 proposed models, 2,681 were accepted, that is, ∼27 per cent. The samples are distributed very favourably so that a good approximation of the posterior, shown as 1-D marginals, can be obtained despite their small number.

The various 2-D subspaces. It is clearly visible that model parameters with small derivatives $G_i$ and correspondingly large posterior covariances traverse the model space comparatively slowly, and have a large oscillation amplitude.

An HMC run with 10,000 test models is summarized in Fig. 4. Using a Hamiltonian trajectory length of 20 s, 2,681 of the proposed models, ∼27 per cent, are accepted. Their distribution is so favourable that the posterior can be approximated very well, despite the small number of samples. To put this result into perspective, we compare to the numbers from the introductory example in Fig. 2, where a more informative Gaussian prior with standard deviation $\sigma_M = 10$ led to an acceptance rate of ∼0.0014 per cent. If we wanted to also have around 2,700 accepted models from the EMH algorithm, we would need to propose more than 19 million test models.

3.2 The mass matrix

The mass matrix $M$, so far not further constrained, plays an important role in tuning and regularizing the HMC sampler. Below, we consider the design of an optimal mass matrix for exactly linear forward problems and the least-squares misfit functional. A more general discussion can be found in Section 5.

3.2.1 Tuning

An obvious potential improvement by tuning is already apparent in Fig. 3. While $m_1$ traverses the energy level set roughly once during the Hamiltonian trajectory, $m_{10}$ traverses it more than 6 times, thus visiting similar positions repeatedly without exploring new regions of model space. For linear forward problems and Gaussian priors, this deficiency may be avoided by choosing the mass matrix as the inverse posterior covariance $M = \tilde{C}_M^{-1} = G^T C_D^{-1} G + C_M^{-1}$. (21)

Following eqs (18) and (20), the matrix-valued oscillation frequency $W$ is then equal to the unit matrix $I$. Therefore, all components oscillate through model space with an identical circular frequency of 1. For illustration, we again return to the toy example from Section 2.2. Choosing the mass matrix according to eq. (21) results in the trajectories shown in Fig. 5. In contrast to Fig. 3, where we chose $M = I$, all components of the model space vector now oscillate with equal frequency, thus traversing the phase space with equal speed.

While being attractive through its apparent simplicity, the choice of eq. (21) deserves some additional remarks. (i) Already having to know the inverse posterior covariance from the outset clearly defeats the purpose of sampling. Nevertheless, this special case of linear forward modelling and Gaussian priors establishes an optimistic baseline that may be used for comparison, as for example in Section 4.1. It furthermore serves as a useful means to gain intuition. (ii) When the prior model covariance $C_M^{-1}$ vanishes or when the prior distribution is uniform, eq. (21) transforms to $M = \tilde{C}_M^{-1} = G^T C_D^{-1} G$, which may not be a proper covariance matrix with strictly non-zero eigenvalues. In this case, $M$ may require regularization, for instance, by adding a scaled unit matrix, $\varepsilon I$, with sufficiently large $\varepsilon > 0$. (iii) Most importantly, for nonlinear problems, eq. (21) does not apply directly. However, when nonlinearity is sufficiently weak, the Hessian $H$ of the potential energy $U$, evaluated at the prior maximum-likelihood model, can be a useful mass matrix. This variant, is referred to as Hessian Hamiltonian Monte Carlo (H-HMC). Differentiating eq. (14) shows that the Hessian of $U$ is equal to the inverse posterior covariance $\tilde{C}_M^{-1}$ of the linear Gaussian problem, thus establishing it as a special case of nonlinear H-HMC.

The possibility to tune HMC sufficiently well for arbitrary priors and potentially nonlinear forward equations is strongly problem-dependent, as we will also see in Section 4 and further discuss in Section 5.3. The possibility to approximate an optimal mass matrix for weakly nonlinear problems is the main reason for our focus on these cases. Any universal statements on more general problems seem difficult to make.
3.2.2 Regularization

The choice of the mass matrix $M$ controls the nature of the proposed models. To see this, we substitute the derivatives (15) and (16) into the first iteration of the leapfrog scheme from Section 2.4. The first model update along a trajectory is then given by

$$m(\Delta t) = m(0) + \Delta t M^{-1} \left( p(0) - \frac{1}{2} \Delta t \nabla_m U(0) \right).$$

Eq. (22) shows that $M^{-1}$ acts to modify the update direction $p(0) - \frac{1}{2} \Delta t \nabla_m U(0)$, which is controlled by the randomly chosen initial momentum $p(0)$ and the misfit (potential energy) derivative $\nabla_m U(0)$. When $M$ is chosen to be a covariance matrix, as, for instance, in eq. (21), $M^{-1}$ effectively acts as a smoothing operator. Consequently, proposed models will tend to be smooth.

Regularizing, or specifically smoothing, proposed models is not a regularization in the sense of least-squares inversion, where it is used to improve the conditioning of a matrix that needs to be inverted. HMC converges for any positive definite $M$. A specific choice, such as a smoother, biases the sampler, and therefore affects convergence. Depending on the specific problem, convergence may improve provided that most of the probability mass of the posterior is indeed located around smooth models. Similar statements apply to other forms of regularization.

4 EXAMPLES

We continue with a series of tomographic/imaging applications with variable dimensionality and degree of nonlinearity. Our objectives are threefold: (i) illustrate practical aspects related to the numerical set-up and tuning, (ii) introduce problem-adapted variants of HMC and (iii) investigate properties of HMC, such as the independence of samples and convergence.

4.1 Linear traveltime tomography

Relating to the formal analysis of linear problems in Section 3, our first example is a linear traveltime tomography based on ray theory. Though linearized tomography ignores the dependence of the ray path on slowness perturbations relative to the background, it remains widely used in both seismic and medical applications (e.g. Fishwick et al. 2005; Ritsema et al. 2011; Koelemeijer et al. 2015; Ozmen et al. 2015; Korta et al. 2017). For simplicity, we assume Gaussian priors. Though this may have limited meaning in practice, it allows us to compare results with the least-squares solution and to investigate the critical role of the mass matrix without excessive numerical effort.

Our numerical set-up, summarized in Fig. 6, mimics a 2-D cross-hole tomography where sources with 1 m spacing emit waves recorded by receivers with 1 m spacing. The domain is 100 m × 100 m wide and discretized by 1 m × 1 m blocks of constant slowness. Thus, the total number of model parameters is $101 \times 101 = 10 201$. The target model consists of a 10 m × 10 m chequerboard pattern, with slowness in each block being a model parameter.

We consider two choices for the mass matrix: (i) the inverse posterior covariance $M_1 = \tilde{C}_M^{-1} = G^T C_D^{-1} G + C_{M}^{-1}$, as suggested in eq. (21) and (ii) the diagonal of $\tilde{C}_M^{-1}$, that is, $M_2 = \text{diag} C_M^{-1}$. In the central parts of the model, the number of significant off-diagonal elements is around 5.

Results for the first scenario are shown in Fig. 7, where the exact posterior means and variances of the least-squares solution are displayed in the leftmost panels. Using only $10^3$ samples, that is, $\sim 10$ per cent of the number of unknowns, a visible approximation of the chequerboard pattern emerges, though still with significant artefacts related to undersampling. The posterior variance, however, is already a close approximation of the posterior, that would be sufficiently accurate in many practical applications. Increasing the number of samples by a factor of 10 improves the posterior mean significantly, especially in those parts of the model that are comparatively poorly covered.
A. Fichtner, A. Zunino and L. Gebraad

Figure 6. Set-up of the synthetic linear traveltime tomography. (a) The target model consists of a chequerboard pattern within a 2-D domain of 100 m × 100 m. For discretization, the domain is partitioned into 101 × 101 = 10 201 constant-velocity blocks. (b) The source-receiver geometry mimics a cross-hole set-up where sources with 1 m spacing along the left boundary emit waves recorded by receivers with 1 m spacing along the right boundary.

Figure 7. Posterior means and covariances for the linear tomography example. (a) Exact solution of the least-squares problem. (b) 1000 samples, using the inverse posterior covariance as mass matrix, that is, \( M = \tilde{C}_M^{-1} \). (c) The same as in panel (b) but for 10 times more samples.

For comparison, Fig. 8 shows results from the HMC sampler for the second scenario where the diagonal of the inverse posterior covariance is used as mass matrix (\( M = \text{diag} \tilde{C}_M^{-1} \)). Neither the posterior mean nor the posterior covariances are close to the least-squares solution, even when 10^5 samples are used, that is, 10 times more than in the rightmost column of Fig. 7.

While requiring the posterior covariance from the outset may seem to defeat the purpose of sampling, the case of \( M = \text{diag} \tilde{C}_M^{-1} \) nevertheless establishes an optimistic baseline that may be used for comparison. In Section 4.3 on nonlinear traveltime tomography, we will demonstrate that other non-diagonal approximations of the posterior covariance can lead to efficient sampler. These examples show that the choice of the mass matrix is not a detail, but an important factor that may change the number of required samples by orders of magnitude.

4.2 Reflection seismology

We continue with a conceptual example where seismic reflection data are inverted for a 1-D subsurface model that illustrates the generation of independent samples by HMC. We consider only a single recording and a linear forward model relating reflection coefficients \( \mathbf{m} \) to a seismogram \( \mathbf{d} \) via the matrix \( \mathbf{G} \). The latter contains the source wavelet and performs a convolution operation, according to the standard
Figure 8. Posterior means and covariances for the linear tomography example when the mass matrix equals the diagonal of the posterior covariance. (a) \(10^3\) samples, (b) \(10^4\) samples and (c) \(10^5\) samples. The colour scales are the same as in Fig. 7, where the full inverse posterior covariance was used.

convolutional model of reflection seismology (e.g. Yilmaz 2001). In this example, we use 128 reflection coefficients and a seismogram consisting of 128 wave amplitude values. Fig. 9 shows the synthetic observed data with the addition of Gaussian noise (left-hand side) generated from the target model, and the prior reflectivity model (right-hand side). With a linear forward problem and Gaussian errors, this example falls into the framework discussed previously in Section 3. As mass matrix we use an approximation of the inverse posterior covariance, \(M = \mathbf{G}^T \mathbf{C}_M^{-1} \mathbf{G}\).

To assess the ability of HMC to produce independent samples, we first consider the correlation coefficients of successive reflectivity models and the autocorrelation \(a_n\) at lag \(n\) of the Markov chain. To establish a baseline, we again use the EMH (Mosegaard & Tarantola 1995) for comparison. As further discussed in Section 5.3, this is not intended to be a quantitative performance benchmark.

Results are shown in Fig. 10. The correlation coefficients are computed for 100 successive models, starting from iteration 10 000 to skip the burn-in phase. While the off-diagonal correlation coefficients of the HMC samples are mostly below 0.75, the correlation coefficients of the EMH samples are generally above 0.994. This suggests that EMH explores model space only slowly compared to HMC. This is also due to the fact that in HMC all model parameters are updated in each iteration, while in EMH only subsets of parameters are updated in each iteration to maintain a relatively high level of acceptance. This strategy is common practice, particularly when sophisticated forms of prior information are used (e.g. Zunino et al. 2015).
Figure 10. Comparison of sample independence for HMC (top row) and EMH (bottom row). (a and b) Correlation coefficients of 100 successive reflectivity models, starting from iteration 10,000. (c and d) Autocorrelation $a_n$ of 5000 successive samples. (e and f) Posterior histograms of the reflection coefficients, with the target model shown as black curve.

Figure 11. Illustrations of relative sample independence in EMH and HMC runs. (a) Misfit, $-\log \rho_m$ as a function of iteration number. While misfit for EMH varies rather slowly, the long-distance moves of EMH result in rapidly varying misfits. The burn-in phase of the HMC run, shown in the smaller inset, only consists of a few samples. (b) Percentage of independent models, defined as $N/N_{\text{eff}}$ with $N = 5000$ successive samples taken after the burn-in phase. The effective sample size $N_{\text{eff}}$ is computed with eq. (23) for each individual model parameter. The HMC chain shows complete independence of realizations for most of the parameters, while the EMH run shows an average percentage of independent samples below 1 per cent.

Figs 10(c) and (d) show the autocorrelation $a_n$ of the sample chain. The autocorrelation for HMC immediately drops close to 0. In contrast, EMH requires around 500 samples to achieve the same level of independence. This again indicates that HMC produces mostly uncorrelated samples, and that the long-distance moves of HMC promote mixing of the chain. The efficient model space exploration of HMC also benefits the representation of the posterior, as shown in Figs 10(e) and (f).

4.3 Nonlinear traveltime tomography

In this section, we present a nonlinear traveltime tomography that accounts for the deflection of ray paths in response to lateral heterogeneities (e.g. Cerveny 2001; Nolet 2008). Nonlinear arrival time tomography has a long history of application from global to local scales (e.g. Zhang & Toksöz 1998; Bijwaard & Spakman 2000; Widiyantoro et al. 2000; Rawlinson et al. 2006). This example is an extension of the linear case presented in Section 4.1, with posterior distributions being non-Gaussian due to the nonlinear relation between seismic velocity (the model...
with computations and memory to construct the Fréchet derivative the gradient of the misfit functional in the abovementioned way may be detrimental for the HMC method, as it requires a large amount of the misfit function can be computed, and linear inversion methods can be used iteratively to solve the inverse problem. However, computing the misfit function is given by

$$
S(v) = \frac{1}{2} \sum_{i=1}^{N_r} \frac{(\tau_{calc}^{(i)} - \tau_{obs}^{(i)})^2}{\sigma_i^2},
$$

where $i$ denotes the receiver index ranging from 1 to $N_r$. The synthetic traveltime data were calculated from the target model in Fig. 12(a), with the addition of uncorrelated Gaussian noise, thus producing 780 traveltime measurements (26 sources and 30 receivers).

The traditional approach to solve the inverse problem, that is, to determine the velocity structure from the traveltime data, is to linearize the forward relationship around a given model $v_0$ and then to compute all individual rays between source–receiver pairs. From this, a matrix $G$ relating a velocity model and the traveltime vector $\tau$ can be constructed: $\tau = \tau_{calc}(v_0) + G(v_0)(v - v_0)$. Subsequently, the gradient of the misfit function can be computed, and linear inversion methods can be used iteratively to solve the inverse problem. However, computing the gradient of the misfit functional in the abovementioned way may be detrimental for the HMC method, as it requires a large amount of computations and memory to construct the Fréchet derivative $G$.

We thus rely on a different approach based on the adjoint method and compute directly the gradient of the misfit as proposed in Leung & Qian (2006) and Taillandier et al. (2009). In the adjoint formulation, the gradient of the misfit is calculated directly using numerical methods, which roughly have the computational cost of two forward simulations and require a limited amount of memory. This provides a significant improvement in the computation of the gradient of the misfit function, making the HMC implementation significantly more efficient. It furthermore improves accuracy by avoiding the linearization of the forward problem, so that “fat-rays” are automatically taken into account.

To compute the traveltime given a velocity model we use the code FDTIMES from Podvin & Lecomte (1991), while for the adjoint calculations we use an in-house code based on the fast sweeping method (Zhao 2005; Leung & Qian 2006; Taillandier et al. 2009; Bretaudeau et al. 2014). The HMC has been implemented and parallelized using the Julia language (Bezanson et al. 2017).

Regarding the set-up of the HMC algorithm, we use a mass matrix equivalent to an estimated inverse of the covariance matrix on the model parameters $M = C_{M}^{-1}$. Moreover, we manually tuned the integration time step $\Delta t$ and the trajectory length of the leapfrog algorithm to achieve an acceptance rate of $\sim 65$ per cent. Since velocity can only be positive, we employed the constrained HMC algorithm described in Appendix B. In total, we produced $10^8$ HMC samples, of which we discarded the first 150 000 as burn-in. To reduce storage requirements, we saved every 25th sample to produce a final collection of of 34 000 posterior velocity models.

---

**Figure 12.** Set-up and results of nonlinear tomography using HMC. (a) The target velocity model used to compute, with the addition of Gaussian noise, the synthetic observations depicted in panel (c). Dots represent source positions, and triangles the positions of receivers where the traveltime is recorded. (b–d) Selection of results from the posterior ensemble: panel (b) is the highest likelihood model, panel (c) a randomly selected model (number 11 000) and (d) the mean model. (e) The synthetic observations (orange) and the traveltimes calculated for model number 11 000. (f) Posterior histogram of velocity at locations $(x, y) = (10, 26)$ km and $(x, y) = (50, 20)$ km, whose position is indicated in plot (d) by stars. Target velocity values are marked by black vertical lines.
While HMC was developed more than 3 decades ago (Duane et al. 1987) and applied in various domains for at least 2 decades (Neal 1996), its potential for the solution of geophysical inverse problems is only starting to be recognized (Muir & Tkalčić 2015; Biswas & Sen 2017; Sen & Biswas 2017). Though the precise reasons for this delayed migration of HMC into geophysics are difficult to quantify, the fact that most existing HMC literature is highly domain-specific is likely to play a role.

The potential of HMC lies in its ability to efficiently sample high-dimensional model spaces, as illustrated for instance in the examples of Sections 4.1 and 4.3. Fundamentally, this ability rests on the exploitation of derivative information that other Monte Carlo methods, such as EMH, genetic algorithms, or the Neighbourhood Algorithm, do not consider.

Though the use of derivatives is also the main limitation of HMC, many geophysical inverse problems with easily computable derivatives are likely to fall within its range of applicability. These include the large classes of tomographic and potential field inversions. For emerging applications where the forward problem is solved purely numerically, adjoint techniques may be used for the efficient computation of derivatives, as proposed in seismology (e.g. Tromp et al. 2005; Fichtner et al. 2006; Plessix 2006), geodynamics (e.g. Bunge et al. 2003; Liu & Gurnis 2008; Colli et al. 2018), geomagnetism (Li et al. 2014), helioseismology (Hanasoge et al. 2011) and other fields. Furthermore, the extension to second-order adjoints (Santosa & Synes 1988; Fichtner & Trampert 2011) offers the possibility to use an approximation of the Hessian as nearly optimal mass matrix.

In the following paragraphs we provide a more detailed discussion of the computational requirements of HMC, its applicability to the probabilistic solution of tomographic problems and the (nearly impossible) comparison with other MCMC methods.

5 DISCUSSION
While HMC was developed more than 3 decades ago (Duane et al. 1987) and applied in various domains for at least 2 decades (Neal 1996), its potential for the solution of geophysical inverse problems is only starting to be recognized (Muir & Tkalčić 2015; Biswas & Sen 2017; Sen & Biswas 2017). Though the precise reasons for this delayed migration of HMC into geophysics are difficult to quantify, the fact that most existing HMC literature is highly domain-specific is likely to play a role.

The potential of HMC lies in its ability to efficiently sample high-dimensional model spaces, as illustrated for instance in the examples of Sections 4.1 and 4.3. Fundamentally, this ability rests on the exploitation of derivative information that other Monte Carlo methods, such as EMH, genetic algorithms, or the Neighbourhood Algorithm, do not consider.

Though the use of derivatives is also the main limitation of HMC, many geophysical inverse problems with easily computable derivatives are likely to fall within its range of applicability. These include the large classes of tomographic and potential field inversions. For emerging applications where the forward problem is solved purely numerically, adjoint techniques may be used for the efficient computation of derivatives, as proposed in seismology (e.g. Tromp et al. 2005; Fichtner et al. 2006; Plessix 2006), geodynamics (e.g. Bunge et al. 2003; Liu & Gurnis 2008; Colli et al. 2018), geomagnetism (Li et al. 2014), helioseismology (Hanasoge et al. 2011) and other fields. Furthermore, the extension to second-order adjoints (Santosa & Synes 1988; Fichtner & Trampert 2011) offers the possibility to use an approximation of the Hessian as nearly optimal mass matrix.

In the following paragraphs we provide a more detailed discussion of the computational requirements of HMC, its applicability to the probabilistic solution of tomographic problems and the (nearly impossible) comparison with other MCMC methods.

5.1 Computational requirements
The main computational requirements of HMC, compared to derivative-free MCMC variants, come from the need to evaluate the derivative of the Hamiltonian along the trajectory, that is, in each time step of the numerical integration. Thus, the computational cost of HMC is strongly controlled by the average trajectory length.

Fig. 3 indicates that trajectories can be too short or too long. If too short, the next proposal is very close to the current position, leading to slow exploration of model space. In contrast, a trajectory that is too long, revisits points in model space that it has seen before, thereby wasting valuable computing time. Furthermore, the time needed to traverse a specific energy level set depends on the energy itself. Low-energy level sets require less, and high-energy level sets require more time for a traversal. An additional complication arises in the presence of periodic trajectories, such as those shown for the linear problem in Fig. 3. Accidentally fixing the trajectory length to an integer multiple of the oscillation frequency will cause the random walker to always revisit the same positions, thus preventing convergence towards the posterior distribution.

At least for (nearly) linear problems with Gaussian priors, we can estimate suitable trajectory lengths. In the case of the optimal mass matrix (21), we have $W = I$, and a trajectory length of $\pi$ will achieve the maximum distance from the starting point of the trajectory. When the forward problem is weakly nonlinear or when a diagonal approximation of eq. (21) is used, a trajectory length of approximately $\pi$ may
still serve as a useful guiding value. To avoid locking in periodic trajectories, the trajectory length may be chosen randomly, as suggested by Mackenzie (1989) and Neal (2011). Alternatively, the length of the trajectory may be chosen adaptively, for example, the No-U-Turn criterion that terminates the integration when the trajectory begins to return towards its starting point (Gelman et al. 2013; Hoffmann & Gelman 2014; Betancourt 2017).

In our examples, we found that 5–10 time steps per trajectory are optimal. Using fewer time steps, limits model space exploration and results in slow convergence. A larger number of time steps further decreases the accuracy of the numerical integrator, thereby leading to low acceptance rates.

For the linear traveltime tomography of Section 4.1, the derivative is given explicitly through $G$, thus not requiring any additional computations. Similarly, in the reflection seismology example in Section 4.2, the derivative is given analytically. The situation, however, is different in the nonlinear traveltime tomography (Section 4.3) where each derivative requires the numerical solution of a forward and an adjoint problem. Despite this computational overhead, our HMC runs for several thousand free parameters did not require any parallel computing.

5.2 HMC for traveltime tomography

Sections 3 and 4.1 demonstrate that the choice of a suitable mass matrix $M$ is the critical element of HMC. Setting $M$ equal to the inverse posterior covariance in our linear tomography example with 10 201 unknowns, allowed us to obtain a stable posterior using only several thousand samples. When only the diagonal elements of the posterior covariance were used, 10^5 samples were not nearly sufficient.

In the case of a perfectly linear problem, the choice of the inverse posterior covariance, or a regularized approximation of it, has the effect of transforming all Hamiltonian trajectories into (high-dimensional) unit circles. As a consequence, all coordinate directions in model space are sampled identically. For weakly nonlinear problems, including the traveltime tomography in Section 4.3, an approximation of the Hessian—evaluated, for instance, at the prior model—can serve the same purpose. The Hessian may be computed, for instance, using second-order adjoints (Santosa & Symes 1988; Fichtner & Trampert 2011) or by making a Gauss–Newton approximation (e.g. Pratt et al. 1998).

One of the main results of this work is the nonlinear traveltime tomography presented in Section 4.3. It demonstrates that a nonlinear tomographic problem with nearly 3000 free parameters can be solved fully probabilistically by combining adjoint techniques and HMC. We emphasize that the inversion was run on a standard laptop computer (2.5 d on a 6-core Intel Xeon), thus not making use of any supercomputing resources. Though extrapolations are inherently difficult, this suggests that the probabilistic solution of 3-D nonlinear tomographic problems is within reach.

5.3 Limitations: No Free Lunch

While HMC is widely recognized as an efficient alternative to other—and at least in geophysics more established—sampling methods, it is important to note that efficiency is unavoidably context-dependent. As formalized by various No-Free-Lunch Theorems (e.g. Wolpert & Macready 1997; Mosegaard 2012), efficiency is not a universal property of a method. Instead, it arises from the combination of the method with prior knowledge about the nature of a specific problem.

In the case of HMC, the No-Free-Lunch Theorems act on two levels. First, concerning the method as a whole, HMC can only be efficient when the derivative of the forward problem can be computed quickly and with sufficient accuracy. Otherwise, the solution of Hamilton’s equations will be too time consuming. Second, within HMC, there is no universally optimal tuning, which includes the choice of the mass matrix, the numerical integration scheme and the length of the trajectories. Again, all tuning is problem specific.

Consequently, it is inherently impossible to make general statements concerning the efficiency of HMC, however quantified. As other inverse methods, HMC requires some trial and error, for both tuning and comparison with other methods. In this context, we note that HMC is not the only method designed to limit the curse of dimensionality at least to some extent. Alternatives include the Neighbourhood Algorithm (Sambridge 1999a,b), parallel tempering (e.g. Marinari & Parisi 1992; Geyer & Thompson 1995; Sambridge 2014) and trans-dimensional sampling (e.g. Green 1995; Sambridge et al. 2006, 2013 Bodin & Sambridge 2009).

Since universally valid statements on the efficiency of HMC are difficult to make, we limited this work to forward problems that are nearly linear. In these cases, an optimal mass matrix can be approximated using the Jacobian $G$ of the forward problem at some point in model space. Again, as a consequence of the No-Free-Lunch Theorems, the term nearly linear is not sharply defined, but application specific.

In this context, we again note that our comparisons of HMC and EMH presented in this paper are not intended as quantitative benchmarks. EMH merely serves as a baseline and as a convenient educational example that includes many of the deficiencies that HMC attempts to overcome.

5.4 Comparison to other methods with focus on local Metropolis–Hastings

The No-Free-Lunch theorems also affect the extent to which HMC can be compared meaningfully to other (MCMC) methods. Since the class of problems for which HMC is efficient is not very large, it is difficult to compare its performance to other methods that are either general
purpose or built for other specific problem classes. One may, nevertheless, argue that a well-tuned Metropolis–Hastings algorithm with local proposal density could potentially compete with HMC; and it is indeed instructive to investigate this specific case.

For simplicity, we consider a Gaussian, \( \rho(\mathbf{m}) = \text{const.} \, e^{-m^T \mathbf{m}} \). According to the Metropolis rule, a move from the current model \( \mathbf{m} \) to a proposed model \( \mathbf{m} + \Delta \mathbf{m} \) is accepted with probability

\[
\min \left( \frac{\rho(\mathbf{m} + \Delta \mathbf{m})}{\rho(\mathbf{m})} \right) = \min \left( 1, e^{-r^2} e^{-2\Delta m^T \mathbf{m}} \right),
\]

with \( r = \sqrt{\Delta m^T \Delta m} \). In a Metropolis–Hastings algorithm with local proposal density, \( \mathbf{m} + \Delta \mathbf{m} \) may be drawn such that it falls within a small radius of \( \mathbf{m} \), such that \( r \) is on average equal to some preset search radius \( r_0 \). Eq. (26) implies that \( -2\Delta \mathbf{m}^T \mathbf{m} \approx r_0^2 \) is required to ensure that the proposed model is likely to be accepted. Assuming, without losing generality for this line of arguments, that \( \mathbf{m} = (-1/2, 0, 0, \ldots)^T \), we thus need \( \Delta m_i \approx r_0^2 \). However, as dimension \( n \) increases, the average magnitude of the components of \( \Delta \mathbf{m} \) decreases as \( 1/\sqrt{n} \). Therefore, it becomes increasingly unlikely that the first component of \( \Delta \mathbf{m} \) is roughly as large as \( r_0^2 \). As a consequence, the search radius \( r_0 \) must be reduced to ensure reasonable acceptance rates. This, however, implies an increasingly slow model space exploration with increasing dimension. It follows that, in contrast to HMC, Metropolis–Hastings with local proposals struggles to maintain high acceptance rates while still exploring a high-dimensional model space efficiently.

We note that this analysis makes no specific assumption on the proposal density used to draw samples in the Metropolis–Hastings algorithm, other than it being local. Thus, the argument also holds for proposal densities that are specifically tuned to approximate the posterior locally.

**ACKNOWLEDGEMENTS**

The authors gratefully acknowledge discussions with Christian Boehm, Amir Khan, Lion Krischer, Klaus Mosengaard, Malcolm Sambridge, Saulé Simutė and Peter-Jan van Leeuwen. The careful reviews of Andrew Valentine and Jan Dettmer helped us to improve the manuscript. This work received funding from the European Research Council under the European Union’s Horizon 2020 research and innovation programme (grant agreement no. 714069), and computational support from the Swiss National Supercomputing Center (CSCS project s741). Part of this work has been funded by the Danish Hydrocarbon Research and Technology Centre through the project ‘Outcrop analog studies of chalk—integrated geology, geophysics and geostatistics’. The first and second author contributed equally to this manuscript.

**REFERENCES**


APPENDIX A: PROOF OF THE HAMILTONIAN MONTE CARLO ALGORITHM

This appendix provides a detailed proof of the HMC algorithm described in Section 2.3. To prepare the stage, we start with a brief review of random walks.

A1 Random walks

MCMC methods produce samples via random walks through model space $M$ that we partition into subsets $M^i$ with equal volumes $M$ around models $m^i$. Assuming the $M^i$ are sufficiently small, the model prior is $P^i = M_{p0}(m^i)$. Similarly, we define $P_{d} = M_{p0}(d|m^i)$ for the discretized likelihood function, and $P_{d} = M_{p0}(m^i'(d)$ for the discretized posterior. Furthermore, we denote by $P^i = M_{p}(m^i)$ a generic probability with density $p^i$.

During a random walk, a sequence of models $(m^i)_{i=1}^{N}$ is visited randomly according to some rules. When the rules are designed such that the number of visits to $M^i$ is proportional to $P^i$, the walk is said to sample the distribution $p$. Assume the walker is currently at $m^i$. The conditional probability of going from $m^i$ to $m^i'$ is called the transition probability $T^{i→i'} = T(m^i|M^i')$. Thus, if the probability for currently being at $m^i$ is $P^i$, the unconditional probability of walking to $m^i'$ is $T^{i→i'}P^i$. Since the walker must go somewhere, including the option of staying at $m^i$, the transition probabilities satisfy

$$\sum_j T^{i→j} = 1.$$  \hfill (A1)

Our particular interest is in random walks that equilibrate asymptotically as $N → ∞$. Equilibrium means that the combined probability to arrive at $m^i$ from any $m^i'$ is equal to the probability of actually being at $m^i$, that is,

$$\sum_j T^{i→j} P^j = P^i.$$  \hfill (A2)

Eq. (A2) can be understood in terms of flows. For this, we interpret $P^i$ as the number of some imaginary particles located at $m^i$. In this sense, $T^{i→j}P^j$ is the number of particles flowing from a specific $m^i$ to $m^i'$, and eq. (A2) states that the total number of particles flowing from all $m^i$ into $m^i'$ is equal to the number of particles at $m^i$. A sufficient condition for equilibration at the distribution $P^i$ is detailed balance of the flow, meaning that as many particles flow from $m^i$ to $m^i'$ as from $m^i'$ to $m^i$, that is,

$$T^{i→j} P^j = T^{j→i} P^i.$$  \hfill (A3)
Indeed, summing eq. (A3) over $j$, and using eq. (A1), we retrieve the equilibration condition (A2).

A2 Detailed balance for HMC

To demonstrate that the canonical distribution $\rho_c$ is indeed the equilibrium distribution of HMC, we need to assert reversibility of the transition in phase space in the sense of the detailed balance eq. (A3). Thus, for two phase space points $(m', p') \in \mathcal{X}'$ and $(m, p) \in \mathcal{X}$, we must show

$$T(\mathcal{X}'|\mathcal{X})P_c(\mathcal{X}) = T(\mathcal{X}|\mathcal{X}')P_c(\mathcal{X}'),$$

for any $\mathcal{X}'$ and $\mathcal{X}$ that partition the phase space. All images of $\mathcal{X}'$ under movement along a Hamiltonian trajectory partition the phase space as well. This implies that $(m', p')$ must be inside some $\mathcal{X}_i'$, and is therefore equal to some particular image, that is, $(m', p') = (m_i, p_i')$. Since Hamiltonian dynamics uniquely translates $\mathcal{X}'$ into its image $\mathcal{X}_i'$, the transition probability $T(\mathcal{X}'|\mathcal{X'})$ must be zero unless $i = j$. Furthermore, since Hamiltonian dynamics is time reversible, $T(\mathcal{X}'|\mathcal{X'})$ is also zero unless $i = j$. In summary,

$$T(\mathcal{X}'|\mathcal{X'}) = T(\mathcal{X}|\mathcal{X}') = 0, \quad \text{if } i \neq j.$$  

Thus, eq. (A4) is trivially satisfied for $i \neq j$, and it remains to consider the more special case

$$T(\mathcal{X}'|\mathcal{X}_i')P_c(\mathcal{X}) = T(\mathcal{X}|\mathcal{X}_i')P_c(\mathcal{X}_i').$$

Under the assumption that our partitioning is so fine that we can consider the canonical distribution constant within each $\mathcal{X}_i'$, we have

$$P_c(\mathcal{X}_i') = X_{c(i)}(m_i', p_i') \quad \text{and} \quad P_c(\mathcal{X}') = X_{c(i)}(m_i', p_i').$$

Note that eq. (A7) only holds because Hamiltonian mechanics preserves the volume $X$ along the trajectory (also see the Appendix). With the help of eq. (10), which determines the transition probability, eq. (A6) now transforms to

$$\min \left[ 1, \frac{\rho_c(m_i', p_i')}{\rho_c(m_i', p_i')} \right] \rho_c(m_i', p_i') = \min \left[ 1, \frac{\rho_c(m_i', p_i')}{\rho_c(m_i', p_i')} \right] \rho_c(m_i', p_i').$$

That eq. (A8), and therefore detailed balance, indeed holds, can now be shown by considering the two cases $\rho_c(m_i', p_i') > \rho_c(m_i', p_i')$ and $\rho_c(m_i', p_i') \leq \rho_c(m_i', p_i')$.

APPENDIX B: CONSTRAINED HMC SAMPLING

Sometimes the model parameters we intend to investigate are subject to constraints, such as being positive or being bounded within a minimum and a maximum value. However, the standard version of the HMC algorithm assumes the variables $m$, which represent the position, to be continuous and defined as real numbers from minus to plus infinity. In this section, we show how it is possible to modify the standard algorithm to obtain a version where the model parameters $m$ are bounded within lower and upper constraints $l_i \leq m_i \leq u_i$. We follow here the strategy proposed in Neal (2011) and we refer the reader to that work for details. The core idea is to set the potential energy $U(m)$ to be infinite for values of $m_i$ which do not satisfy the constraints, inducing a zero probability for those values. The latter can be obtained by modifying the potential energy by adding a term that goes to infinity outside the constraints and consequently changing the dynamics of the system.

To actually run the modified dynamics one can use the technique of ‘splitting’ the Hamiltonian (Neal 2011, and references therein): the Hamiltonian is expressed as a sum of terms $H(m, p) = H_1(m, p) + H_2(m, p) + H_3(m, p) + \ldots$ and then the dynamics (eq. 4) are simulated for each term sequentially for a time $\Delta t$ and this procedure is repeated until the desired total time of simulation is reached. As an illustration of this approach, consider the symmetric splitting:

$$H(m, p) = H_1(m) + H_2(p) + H_3(m) = \frac{U(m)}{2} + K(p) + \frac{U(m)}{2}. \quad \text{(B1)}$$

If we apply the dynamics to each of the three terms separately for a time $\Delta t$, we retrieve the leapfrog method. To constrain the model parameters with upper and lower bounds $l_i \leq m_i \leq u_i$, one possible approach is then to use the following splitting of the Hamiltonian:

$$H(m, p) = H_1(m) + H_2(m, p) + H_3(m) = \frac{U_0(m)}{2} + (K(p) + U_1(m)) + \frac{U_0(m)}{2}. \quad \text{(B2)}$$

where we introduce a new term for the potential energy $U_1(m_i)$ that goes to infinity if any of the components $m_i$ are outside the bounds and is equal to zero if all $m_i$ are within the bounds. Applying the splitting method to the new Hamiltonian, we obtain an algorithm identical to the leapfrog method for $H_1$ and $H_3$ (i.e. steps 11 and 13), while we have a modification for $H_2 = K(p) + U_1(m)$. After running the first step (eq. 11) and calculating the new $m_i$ by means of the second step (eq. 12) we check, one at a time, whether the new components of $m$ are within the bounds. If that is the case, we have $U_1(m_i) = 0$ by definition. Therefore, the dynamics are not influenced by the new term $U_1$, and we can continue with the last step of the leapfrog algorithm (eq. 13). Otherwise, if a component $m_i$ is found to be outside the bounds, $U_1(m_i)$ plays a role and the dynamics change. The steep potential wall produced by $U_1(m_i)$ causes the trajectory to reflect with opposite momentum $p_i' = -p_i$ in order to respect the constraints. In practice, the following procedure is applied after the fist step (eq. 11):
(1) For each $i$, set $p_i' = p_i(t + \Delta t/2)$ and $m_i' = m_i(t) + \Delta t \frac{\partial K}{\partial p_i'}|_{t+\Delta t/2}$ (eq. 12).

(2) For each $i$, while $m_i$ is outside the bounds $l_i$ and $u_i$
   (i) if $m_i > u_i$ set $m_i' = u_i - (m_i' - u_i)$ and $p_i' = -p_i'$.
   (ii) if $m_i < l_i$ set $m_i' = l_i + (l_i - m_i')$ and $p_i' = -p_i'$.

(3) Set $m_i(t + \Delta t) = m_i'$ and $p_i(t + \Delta t/2) = p_i'$.

Finally, the third step of the leapfrog algorithm is applied (eq. 13), and the whole procedure is repeated until the total time of simulation is reached. This ensures that each of the model parameters is within bounds while running the dynamics. A pseudo-code listing of the constrained HMC algorithm is provided in the section below.

APPENDIX C: PSEUDO-CODE FOR REGULAR AND CONSTRAINED HMC

C1 Regular HMC

The following is a pseudo-code listing for the regular HMC algorithm from Section 2.3, combined with the leapfrog integration from Section 2.4.

Algorithm 1 Hamiltonian Monte Carlo

1: procedure ONE ITERATION OF HMC
2: draw $p_{\text{cur}}$ from $\mathcal{N}(0, \mathbf{M})$
3: $p_{\text{new}} = p_{\text{cur}}$
4: $q_{\text{new}} = q_{\text{cur}}$
5: $U(q_{\text{cur}}) = -\log(\text{posterior}(q_{\text{cur}}))$
6: $K(p_{\text{cur}}) = p_{\text{cur}}^T \mathbf{M}^{-1} p_{\text{cur}}$
7: $H_{\text{cur}} = U(q_{\text{cur}}) + \frac{1}{2} K(p_{\text{cur}})$
8: $p_{\text{new}} = p_{\text{new}} - \epsilon \frac{\nabla U(q_{\text{new}})}{2}$
9: for $l = 1 \rightarrow L$ do
10:   $q_{\text{new}} = q_{\text{new}} + \epsilon \nabla K(p_{\text{new}})$
11:   if ($l < L$) then
12:      $p_{\text{new}} = p_{\text{new}} - \epsilon \nabla U(q_{\text{new}})$
13:   end if
14: end for
15: $p_{\text{new}} = p_{\text{new}} - \epsilon \frac{\nabla U(q_{\text{new}})}{2}$
16: $p_{\text{new}} = -p_{\text{new}}$
17: $U(q_{\text{new}}) = -\log(\text{posterior}(q_{\text{new}}))$
18: $K(p_{\text{new}}) = p_{\text{new}}^T \mathbf{M}^{-1} p_{\text{new}}$
19: $H_{\text{new}} = U_{\text{new}} + K_{\text{new}}$
20: if (rannum $\leq \exp(-(H_{\text{new}} - H_{\text{cur}})))$ then
21:   $q_{\text{cur}} = q_{\text{new}}$
22: else
23:   $q_{\text{cur}} = q_{\text{cur}}$
24: end if
25: end procedure

C2 Constrained HMC

The following is a pseudo-code listing for the constrained HMC covered in Appendix B.
Algorithm 2 Hamiltonian Monte Carlo with constraints

```
procedure ONE ITERATION OF HMC WITH CONSTRAINTS
2: draw \( p_{\text{cur}} \) from \( N(0, M) \)
    \[ p_{\text{new}} = p_{\text{cur}} \]
4: \( q_{\text{new}} = q_{\text{cur}} \)
    \[ U(q_{\text{cur}}) = -\log(\text{posterior}(q_{\text{cur}})) \]
6: \( K(p_{\text{cur}}) = \frac{p_{\text{cur}}^T M^{-1} p_{\text{cur}}}{2} \)
    \[ H_{\text{cur}} = U(q_{\text{cur}}) + K(p_{\text{cur}}) \]
8: \( p_{\text{new}} = p_{\text{new}} - \frac{\epsilon}{2} \nabla U(q_{\text{new}}) \)
for \( l = 1 \rightarrow L \) do
10: \( q_{\text{new}} = q_{\text{new}} + \epsilon \nabla K(p_{\text{new}}) \)
    for \( i = 1 \rightarrow N \) do
12: \( q_{\text{tmp}} = q_{\text{new}} \)
    \( p_{\text{tmp}} = p_{\text{new}} \)
    for \( i = 1 \rightarrow \text{length}(q) \) do
        while \( q(i)_{\text{new}} > \text{upc}(i) \) or \( q(i)_{\text{new}} < \text{lowc}(i) \) do
            if \( q(i)_{\text{tmp}} > \text{upc}(i) \) then
                \( q(i)_{\text{tmp}} = \text{upc}(i) - (q(i)_{\text{tmp}} - \text{upc}(i)) \)
            end if
            if \( q(i)_{\text{tmp}} < \text{lowc}(i) \) then
                \( q(i)_{\text{tmp}} = \text{lowc}(i) + (\text{lowc}(i) - m(i)_{\text{tmp}}) \)
            end if
        end while
16: \( q_{\text{new}} = q_{\text{tmp}} \)
    \( p_{\text{new}} = p_{\text{tmp}} \)
    end for
end for
if \( l < L \) then
30: \( p_{\text{new}} = p_{\text{new}} - \frac{\epsilon}{2} \nabla U(q_{\text{new}}) \)
end if
32: end for
34: \( p_{\text{new}} = -p_{\text{new}} \)
    \[ U(q_{\text{new}}) = -\log(\text{posterior}(q_{\text{new}})) \]
36: \( K(p_{\text{new}}) = \frac{p_{\text{new}}^T M^{-1} p_{\text{new}}}{2} \)
    \[ H_{\text{new}} = U_{\text{new}} + K_{\text{new}} \]
38: if \( \text{rannum} \leq \exp(-(H_{\text{new}} - H_{\text{cur}})) \) then
    \( q_{\text{cur}} = q_{\text{new}} \)
else
    \( q_{\text{cur}} = q_{\text{cur}} \)
end if
end procedure
```