Bayesian inference based on algorithms: MH, HMC, Mala and Lip-Mala for Prestack Seismic Inversion

Richard Perez-Roa[1], Saba Infante[2,3], Gabriel Barragan[2], Raul Manzanilla[2]

[1]Geology and Applied Geophysics Research Group, School of Earth Sciences, Energy and Environment, Yachay Tech University, Hacienda San José, Urcuquí, 100119, Ecuador
[2]Numerical Analysis and Data Science Research Group, School of Mathematical and Computational Sciences, Hacienda San José, Urcuquí, 100119, Ecuador
[3]Department of Mathematics, FACyT Universidad de Carabobo, Naguanagua, Valencia, 2005, Venezuela

Correspondence to: Richard Pere-Roa (rperez@yachaytech.edu.ec)

Abstract. Seismic data inversion for estimating elastic properties is a crucial technique for characterizing reservoir properties post-drilling. The choice of inversion method significantly impacts results. Markov chain Monte Carlo (MCMC) algorithms enable Bayesian inference, incorporating seismic data uncertainty and expert information via prior distribution. This study compares the performance of four inversion methods—Metropolis-Hastings (MH), Hamiltonian Monte Carlo (HMC), and two Lagrangian Diffusion variants [(MALA -(Metropolis-adjusted Langevin algorithm) and Lip-MALA (Lipschitz-MALA)]—in prestack seismic inversion, using synthetic and real-world data from an eastern Venezuelan hydrocarbon reservoir. All four methods show acceptable performance but differ in specific strengths and weaknesses. Gradient-based methods (HMC, MALA, and Lip-MALA) outperform MH in velocity estimation. Density estimation is more challenging; MH and HMC yield unsatisfactory results, whereas MALA and Lip-MALA show promise. Execution time varies significantly: MH and MALA are substantially faster than HMC and Lip-MALA. Therefore, both accuracy and computational efficiency should be considered when choosing a method. The study evaluates the mean values and standard deviations of the subsequent parameters: P-wave (V_P), S-wave velocity (V_S) and density (ρ). The quality of the MCMC sample is checked using correlations, objective function plots, seismic trace and Root Mean Square Error (RMSE) estimation. Acceptance rate and execution time assessments reveal HMC has the lowest acceptance rate, and MH the shortest execution time. Future research aims to extract additional elastic parameters (e.g. P-impedance and S-impedance) and reservoir properties (e.g. Clay volume, Porosity and Water saturation), enhancing subsurface understanding. Integrating well log conditioning into the model could improve vertical resolution near wells and align the model with well data at drilling locations.

1 Introduction

The accurate characterization of hydrocarbon reservoirs is crucial for effective reservoir management. This process necessitates the integration of two distinct information sets: general reservoir knowledge and reservoir-specific observations. General reservoir knowledge encompasses insights gleaned from analogous reservoir studies, coupled with established principles in seismic and rock physics. In contrast, reservoir-specific observations include direct measurements of the reservoir under study, including well data, seismic surveys, and historical production data.

Seismic data play an important role in reservoir characterization due to their wide spatial coverage. Unlike well logs, which are limited to individual well locations, seismic surveys provide a comprehensive picture of the entire reservoir... To leverage this information for reservoir characterization, we require methods to transform seismic amplitudes into rock properties relevant for reservoir description. Seismic inversion stands as a prominent technique for extracting ~~this~~ those elastic and/or petrophysical properties from seismic data.

Seismic inversion is a geophysical inverse problem. It aims to indirectly extract information about the subsurface medium (elastic properties, lithology, etc.) from observed seismic data. This necessitates a robust mathematical framework, typically represented by an equation or system of equations, that accurately describes the physical relationship between the medium (geological model) properties and the recorded seismic response. The process of mapping the parameters of a geological model to quantities in the data space is known as forward modelling, generally of the type (Tarantola, 2005):

$$d_{obs} = \cancel{F(m) + \epsilon}\ g(\mathbf{m}) + \xi, \tag{1}$$

where d_{obs} is the observed data, g~~F~~ is the function that relates the parameters of the medium **m** with the observed data and ~~ε~~ξ represents noise due to data and/or modelling errors. For the particular case of amplitude versus offset (AVO) prestack seismic inversion (Helland-Hansen et al., 1997; Ma, 2002; Buland and Omre, 2003) is an ill-posed problem (Landa and Treitel, 2016), there is usually inconsistency, and the solution is extremely sensitive and unstable to measurement errors. The most important physical parameters for seismic inversion are P-wave (V_P) and S-wave velocity (V_S) and density (ρ). These parameters can be used to derive Lame parameters, which are sensitive to fluid and saturation in rocks (Clochard et al., 2009). Petrophysical parameters, such as porosity, sand/shale ratio, and gas saturation, can then be estimated from the inverted Lame parameters (Goodway, 2001). Petrophysical parameters are very important in the interpretation of seismic data, which is a crucial process in oil exploration and production projects. By understanding the petrophysical properties of the earth's surface, geologists and engineers can better identify potential reserves of oil and gas.

The approach of this work is based on computer statistics, which allows to include uncertainty in seismic data, prior knowledge of model parameters, and~~,~~ through the application of Monte Carlo methods, to generate samples that allow to estimate the posterior distribution~~.~~ (solutions of the inverse problem). Bayesian inference take into account a likelihood of the seismic data, a prior distribution containing rock property information, ~~combined~~ combining these two sources of information for later applies Bayes' Theorem (or Bayes' Rule) to approximate the solution.

There is an extensive literature related to Markov chain Monte Carlo (MCMC) that explore spaces in high dimensions. The Metropolis-Hastings (MH) algorithm was popularized by Metropolis et al., (1953) and Hastings, (1970), initially it was used to simulate the distribution of states of a system of idealized molecules. The MH is a method that facilitates the construction of a stationary Markov chain that converges to a posterior distribution. A more general algorithm is the Hamiltonian Monte

Carlo (HMC) that was originally developed in the context of lattice quantum chromodynamics (e.g., Duane et al., 1987). Subsequently, the method was extended to Bayesian neural networks (Neal, 1996) and incorporated into widely recognized textbooks (MacKay, 2003; Bishop, 2006).is another approach to molecular simulation introduced by Alder and Wainwright, (1959) and Duane et al., (1987) and popularized by Neal, (2012) and Betancourt, (2018). The HMC is applied in many disciplines such as: Neural Networks and Machine Learning, (Bishop, 2006); in molecular simulations, (Dubbeldam et al., 2016). In inverse geophysics problems, Bosch et al., (2007) solved an inverse problem following the MCMC methodology, where they quantify the uncertainties of geophysical data and petrophysical properties, combining seismic information with powerful computational methods, establishing a relationship between porosity and acoustic impedance in reservoir areas. In Wu et al., (2019) proposed a MCMC method to reduce the sampling range and improve the efficiency and resolution of impedance inversion, using a Gaussian MH algorithm with data handling for the sampling function [(Gaussian MH sampling with data driving (GMHDD) approach)]. In Gebraad et al., (2020) developed a Bayesian inversion methodology to treat the full elastic waveform, their proposal is based on HMC sampling of the posterior distribution, use adjoint techniques, and compute the mass matrix considering different sensitivities of seismic velocities and densities. In Izzatullah et al., (2021) studied the seismic inversion problem under a Bayesian approach, implement a MCMC algorithm inspired by Langevin dynamics, and propose a rule for determining the adaptive step size in MCMC algorithm that replaces the MH acceptance step. In Fichtner and Simuté, (2018) developed a model of probabilistic inversion that considers the heterogeneous 3D structure of the earth, the method is based on numerical simulations of wave fields in complex media and on HMC sampling. In de Lima et al., (2023) used a full waveform inversion (FWI) method, the proposed technique is of high resolution and is used in geophysics to evaluate the physical parameters and build subsurface models in a noisy scenario and with limited data, proposed a new way to adjust the mass matrix based on the seismic survey acquisition geometry, and demonstrate significant improvements of the ability of the HMC method in reconstructing reasonable seismic models with manageable computational costs. Both MALA and Lip-MALA algorithms belong to the family of Langevin Monte Carlo algorithms, which are derived from the Langevin dynamics. Lip-MALA algorithm implements a locally Lipschitz adaptive step-size (Izzatullah et al., 2021, Welling and Teh, (2011), Roberts and Tweedie 1996; Nemeth and Fearnhead 2020).

This article studies the impact of the choice of the inversion method on the results of the inversion. We compare the performance of four algorithms: MH, HMC, MALA and Lip-MALA for prestack seismic inversion. We validate the algorithms with synthetic data and measure the quality of the samples generated by the MCMC algorithms through diagnostic methods. This article is structured by first reviewing the theory of seismic inversion, then we review the theory of the 4 methods used and the AVO theory, in the results part we show what was obtained for synthetic data and real data and finally we have the discussion and conclusions.

2 The seismic inversion problem

Seismic inversion is a way to use seismic waves to understand ~~about~~ the subsurface $\mathbf{m} \in R^{N_m}$, such as seismic velocities and densities from the observed seismic data $\mathrm{d_{obs}} \in \mathrm{R}^{\mathrm{N_D}}$, where $\mathrm{N_m}$ and $\mathrm{N_D}$ are the dimensions of the model parameters and the observed seismic data. This can be solved using a Bayesian framework that treats the inversion problem as a statistical inference problem.

In Bayesian inference, we start from the prior probability distribution of the parameters of the subsurface models. This prior distribution represents our knowledge of the ground before seeing the seismic data. We then update this prior distribution with the seismic data using Bayes' theorem to obtain the posterior probability distribution of the subsurface model parameters.

The posterior probability distribution encodes the degree of confidence in ~~our~~ subsurface model parameter estimate. This distribution allows us to quantify the uncertainty of the underground parameters, considering the seismic data, the prior data and the forward model.

To fully characterize the posterior probability distribution, we usually need to estimate the model parameters. ~~several samples in the parameter space of the model.~~ This can be computationally expensive.

In this section, we develop a general approach to seismic Bayesian inference. This framework can be used to make Bayesian inference more practical in real-world applications.

2.1 Bayesian inference framework for seismic data

~~Interest for Bayesian statistics methods for high-dimensional models has recently received very attention motivated by machine learning application.~~ Bayesian methods attempt to sample the full posterior distribution over the parameters and possibly latent variables which provides a way to assert uncertainty in the model.

Under the statistical approach of Bayesian inversion, the objective is to find the posterior distribution of the latent states (unknown parameters) $\mathbf{m}$ given the observed data $\mathrm{d_{obs}}$. To solve Bayesian seismic inversion, we need to know about the prior probability density $\rho(\mathbf{m})$ and the likelihood function $\mathrm{L}(\mathbf{m})$. The prior probability density tells us how confident we are in our knowledge of the subsurface model parameters, before we look at the seismic data. The likelihood function tells us how likely it is that a particular set of subsurface model parameters would produce the seismic data that we actually observed (Izzatullah et al., 2021).

Bayes' theorem combines the prior probability density and the likelihood function to give us the posterior probability density $\sigma(\mathbf{m})$. The posterior probability density tells us how confident we are in our knowledge of the subsurface model parameters, after we have looked at the seismic data (Bosch, 2004).

$$\sigma(\mathbf{m}) = c\mathrm{L}(\mathbf{m})\rho(\mathbf{m}) \tag{2}$$

where $\sigma(\mathbf{m})$ is posterior probability density, c is a normalization constant, $\mathrm{L}(\mathbf{m})$ is likelihood and $\rho(\mathbf{m})$ is prior probability density. ~~In other words, the prior probability density tells us what we think we know about the subsurface before we look at~~

the data. The likelihood function tells us how much the data changes our mind about the subsurface. And the posterior probability density tells us what we think we know about the subsurface after we have looked at the data. In this paper, we will focus on the posterior probability distribution, which can be expressed mathematically as

$$\sigma(\mathbf{m}) = c\exp(-S), \tag{3}$$

with the half-sum of squares *S* being:

$$S = \frac{1}{2}\left(d_{obs} - g(\mathbf{m})\right)^T C_d^{-1}\left(d_{obs} - g(\mathbf{m})\right) + \frac{1}{2}\left(\mathbf{m} - \mathbf{m}_{prior}\right)^T C_m^{-1}\left(\mathbf{m} - \mathbf{m}_{prior}\right), \tag{4}$$

with $g: \mathbf{m} \to d_{obs}$, being the function solving the seismic forward problem, C_d being the data covariance matrix that describes second-order statistics on the data uncertainties, C_m an appropriate covariance matrix describing variability and correlation between parameters of the medium and $\mathbf{m}_{prior}$ is a prior model.

Usually the equation given in (3) is analytically intractable, but it can be approximated numerically by using the simulated samples $\mathbf{m} \sim \sigma(\mathbf{m})$, using Markov chain Monte Carlo (MCMC) computational algorithms, (Metropolis et al., 1953; Hastings, 1970; Estévez et al., 2012; Sanchez et al., 2016).

3 Theoretical Background for Metropolis-Hastings, Hamiltonian Monte Carlo and Langevin Diffusion

3.1 Metropolis-Hastings (MH)

The Metropolis algorithm is a Markov Chain Monte Carlo (MCMC) method used to generate samples from Markov chains. MH algorithm, introduced by Metropolis et al. (1953) and later generalized by Hastings (1970), defines a transition probability that ensures the Markov chain is ergodic, satisfies detailed balance, and exhibits reversibility, (Chib and Greenberg, 1995). This algorithm generates a sequence of values m forming a Markov chain, which can be used to approximate a posterior density σ(m). The Metropolis-Hastings algorithm, which was proposed by Metropolis et al., (1953) and later generalized by Hastings, (1970).

In MH, a candidate configuration is produced from a source sampling distribution, which is not the target distribution. The source sampling distribution can be anything, but it is desirable for the efficiency of the algorithm that it is somehow close to the target distribution, which is to be sampled. The algorithm is based on comparing the candidate configuration and the current configuration, to decide whether the candidate is accepted as the next step of the chain or if it is rejected, repeating the current configuration as the new link.

In order to establish this comparison, it is necessary to calculate the multivariate density for both configurations, or the ratio between them. Consider a a prior model source density, $\rho(\mathbf{m})$, and a likelihood function, $L(\mathbf{m})$, where $\mathbf{m}$ is a point in the sample space. We will assume that we have a chain that converges to the source distribution. Starting from any configuration in the parameter space, and with the chain in the configuration corresponding to the nth step, m_n $\mathbf{m}^{(t-1)}$, the MH Metropolis algorithm defines a chain that converges to the target density $\sigma(\mathbf{m})$.

A variety of proposal functions can be used and the speed of convergence and the quality of the estimators obtained substantially depend on the quality of this proposal (Symmetric MH, Independent MH, Random walk MH). The acceptance rate and the effective sample size are used to calibrate the MH algorithm. This means that a range of values of the parameters involved in the proposal must be compared and the value that maximizes the objective function must be selected, (Robert 2016).

The configuration for the new step ~~m_{n+1}~~ $\mathbf{m}^{(t)}$ according to the Metropolis transition rule is as follows:

1. Initialization: Choose a proposal function $q\left(\mathbf{m}/_{\mathbf{m}^{(t-1)}}\right)$, an initial state $\mathbf{m}$. ~~Generate a candidate configuration $\tilde{\mathrm{m}}$ from the transition rule of the convergent chain to the source probability density.~~
2. Propose a new state: At iteration t generate a candidate values $\tilde{\mathbf{m}}$ from the proposal probability density $\tilde{\mathbf{m}} \sim q\left(\tilde{\mathbf{m}}/_{\mathbf{m}^{(t-1)}}\right)$, and generate $u \sim U(0,1)$, where $q\left(\mathbf{m}/_{\mathbf{m}^{(t-1)}}\right) = q\left(\mathbf{m}^{(t-1)}/_{\mathbf{m}}\right)$, is symmetric probability distribution. ~~Calculate the value of $\mathrm{L}(\tilde{\mathrm{m}})$.~~
3. Compute the acceptance probability, ~~Accept the candidate by setting $\mathrm{m} = \tilde{\mathrm{m}}$ with probability,~~

$$\mathrm{p}_{\mathrm{accept}} = \min\left(1, \frac{\mathrm{L}(\tilde{\mathbf{m}})}{\mathrm{L}(\mathbf{m})}\right) \qquad (5)$$

4. If $u \leq \mathrm{p}_{\mathrm{accept}}$, accept $\tilde{\mathbf{m}}$ and set $\mathbf{m}^{(t)} = \tilde{\mathbf{m}}$. Otherwise reject $\tilde{\mathbf{m}}$ and set $\mathbf{m}^{(t)} = \mathbf{m}^{(t-1)}$. ~~If $\tilde{\mathrm{m}}$ is better than the m, make $\mathrm{m} = \tilde{\mathrm{m}}$. Otherwise, keep the current model as-is. Then, repeat this process returning to 1.~~

Iteratively repeating the MH rule generates a chain that converges to a sample of the target probability density.
The MH algorithm has some advantages and disadvantages allows sampling from arbitrary objective distributions, it is not necessary to determine the marginals, it is simple to implement, and it has a better acceptance and rejection rate in high-dimensional spaces than other competing algorithms. In addition, it can have a poor convergence rate when samples are correlated, it has problem when the target distribution is multimodal, and it is sensitive to the step size between draws, choosing too large or small a step can affect the convergence of the parameters.

3.2 Hamiltonian Monte Carlo (HMC)

Hamiltonian Monte Carlo (HMC) is a sampling algorithm that was originally developed for molecular dynamics (Duane et al., 1987). It is now commonly used for sampling problems where the gradients of the posterior probability distribution $\mathrm{p}(\mathbf{m}|d_{obs})$ with respect to the model parameters m are easy to compute. HMC is more efficient than standard Metropolis-Hastings for high-dimensional problems. However, the cost of generating independent samples with HMC grows faster than the cost of generating samples with Metropolis-Hastings. Specifically, the cost of generating independent samples with HMC grows as $\mathcal{O}\left(\mathrm{n}^{5/4}\right)$ (Neal, 2012), where n is the dimension of the model parameter space. The cost of generating independent samples with Metropolis-Hastings grows as $\mathcal{O}(\mathrm{n}^2)$ (Creutz, 1988).

Hamiltonian Monte Carlo (HMC) is a Markov chain Monte Carlo (MCMC) algorithm that uses classical Hamiltonian mechanics (Landau and Lifshitz, 1976) to sample from an arbitrary *n*-dimensional probability density function (PDF) $\mathrm{p}(\mathbf{m}) = \sigma(\mathbf{m})$. HMC regards the current state m of the Markov chain as the location of a physical particle in *n*-dimensional space M. The particle moves under the influence of a potential energy, *U*, which is defined as the negative logarithm of the PDF (Gebraad et al., 2020):

$$\mathrm{U}(\mathbf{m}) = -\ln \mathrm{p}\,(\mathbf{m}) \tag{6}$$

If the probability density function p of the subsurface model parameters is Gaussian, then the potential energy U of the system is equal to the least squares misfit $\mathrm{S}(\mathbf{m})$, up to an additive constant. To make the system physically complete, we need to add momentum variables p and mass matrices for each dimension of the model parameter space. The momentum variables represent the velocity of the Markov chain as it moves through the parameter space, and the mass matrix M of dimension $n \times n$ represents the resistance to change. The kinetic energy of the system is defined by the momenta and the mass matrix as

$$\mathrm{K}(\mathrm{p}) = \frac{1}{2}\mathrm{p}^{\mathrm{T}}\mathrm{M}^{-1}\mathrm{p} \tag{7}$$

The HMC algorithm uses a random momentum p, drawn from a multivariate Gaussian distribution with covariance matrix M. The potential energy of the system depends on the location, and the kinetic energy depends on the momentum. The total energy of the system, also known as the Hamiltonian, is the sum of the potential and kinetic energies,

$$\mathrm{H}(\mathbf{m}, \mathrm{p}) = \mathrm{U}(\mathbf{m}) + \mathrm{K}(\mathrm{p}) \tag{8}$$

Hamilton's equations

$$\frac{\mathrm{d}\mathbf{m}}{\mathrm{d}\tau} = \frac{\partial \mathrm{H}}{\partial \mathrm{p}}, \quad \frac{\mathrm{d}\mathrm{p}}{\mathrm{d}\tau} = -\frac{\partial \mathrm{H}}{\partial \mathbf{m}} \tag{9}$$

Making an analogy with the physical problem, ~~W~~we want to find how the particle's position changes over time, as represented by the artificial time variable τ. Hamilton's equations tell us how the position and momentum of a particle change over time, but they can be complicated. We can simplify them by using the fact that the kinetic energy of a particle depends only on its momentum and its potential energy depends only on its position~~.~~,

$$\frac{\mathrm{d}\mathbf{m}}{\mathrm{d}\tau} = \mathrm{M}^{-1}\mathrm{p}, \quad \frac{\mathrm{d}\mathrm{p}}{\mathrm{d}\tau} = -\frac{\partial \mathrm{U}}{\partial \mathbf{m}} \tag{10}$$

In HMC, the model parameters $\mathbf{m}$ and their moment p are represented as a state. It then evolves the state $(\mathbf{m}, \mathrm{p})$ over time τ using Hamiltonian dynamics. This generates a distribution of the possible states of the system with new position $\tilde{\mathbf{m}}$, momentum $\tilde{\mathbf{p}}$, potential energy $\tilde{\mathrm{U}}$, and kinetic energy $\tilde{\mathrm{K}}$, which is a sample of the joint momentum and model space. Since we are only interested in the model parameters, we marginalize over the momenta to obtain a sample of the posterior distribution of the model parameters. This results in samples from the posterior distribution.

$$p(\mathbf{m}) = \exp\left(-U(\mathbf{m})\right) \quad (11)$$

If we could solve Hamilton's equations exactly, we could generate an infinite number of valid samples of the posterior probability distribution of the subsurface model parameters p(**m**). However, Hamilton's equations cannot be solved analytically for nonlinear forward models, so we must use numerical integration. Suitable integrators for numerical integration are symplectic, which means that they preserve time reversibility, phase space partitioning, and volume (Neal, 2012; Fichtner and Zunino, 2019). However, explicit time stepping schemes do not exactly preserve the Hamiltonian. In this work, we use the leapfrog method~~ for numerical integration~~, as described in (Neal, 2012). Since the Hamiltonian is not preserved exactly, the leapfrog method introduces a small error into the samples of p(**m**). The Metropolis-Hastings correction step is a way to "fine-tune" the results of numerical integration to make sure that they are as accurate as possible.

To summarize, samples of the model parameters are generated by starting with a random model **m** and then following these steps (Gebraad et al., 2020):

1. Generate random momenta **m** values from a Gaussian distribution with mean **0** and covariance matrix **M**.
2. Evaluate the Hamiltonian *H* of model **m**, using its momenta p.
3. Given the current values of the model parameters **m** and p, and a time step τ, use a numerical integrator to calculate the updated values of **m** and p, $\tilde{\mathbf{m}}$ and $\tilde{p}$, after a time period of τ.
4. Calculate the Hamiltonian $\tilde{H}$ of the model $\tilde{\mathbf{m}}$ with momenta $\tilde{p}$.
5. Permit the suggested change from **m** to $\tilde{\mathbf{m}}$ to occur with probability.

$$p_{accept} = \min\left(1, \exp\left(H - \tilde{H}\right)\right), \quad (12)$$

6. If the new state is better than the current state, accept $\tilde{\boldsymbol{m}}$ and change it to the current state. Otherwise, keep the current state. Then go back to step 1...

The acceptance rate of the leapfrog integration algorithm is largely influenced by how well it conserves energy in the trajectory. If the time steps are too large or the gradients of the fitting function are incorrectly calculated, the algorithm will save less energy, and the acceptance rate will decrease. Simply put, the leapfrog integration algorithm works by bouncing model parameters back and forth across the simulated energy landscape. The acceptance rate determines how often the algorithm accepts a new proposed model parameter. If the time steps are too large or the gradients are calculated incorrectly, the algorithm cannot follow the energy landscape accurately and will likely reject the proposed model parameters. This results in lower acceptance.

3.3 The Langevin dynamics

Langevin dynamics are a mathematical model of Brownian motion, named after the French physicist Paul Langevin (Lemons and Gythiel, 1997) who developed them in 1908. Langevin dynamics is a simplification of Albert Einstein's approach to Brownian motion, which is based on Newton's second law of motion. The Langevin dynamics for target distribution $p(\mathbf{m}_t|d_{obs})$, is a continuous-time stochastic process $(\mathbf{m}_t)_{t\geq 0}$ in R^n that evolves following the stochastic differential equation (Roberts and Stramer, 2002; Nemeth et al., 2016; Izzatullah et al., 2021) and (Infante et al., 2019),

$$d\mathbf{m_t} = -\Sigma\nabla \log p\,(\mathbf{m_t}|d_{obs})dt + \sqrt{2}\,\Sigma^{-\frac{1}{2}}dW_t \qquad (13)$$

where $(W_t)_{t\geq 0}$ is a standard *n*-dimensional Brownian motion, Σ is a symmetric positive definite matrix, $\nabla \log p\,(\mathbf{m_t}|d_{obs})$ is the drift term of the Brownian particle $\mathbf{m_t}$ and p(.) is a stationary posterior distribution.

3.3.1 Metropolis-adjusted Langevin algorithm (MALA)

In the practice, a standard approach is to discretise the equation (13) using the Euler-Maruyama discretisation (Stuart et al., 2004) and we obtained the Unadjusted Langevin algorithm (ULA) given by

$$\mathbf{m} = \widetilde{\mathbf{m}} + - \tau_t\Sigma\nabla \log p\,(\widetilde{\mathbf{m}}|d_{obs}) + \sqrt{2\tau_t}\,\Sigma^{-\frac{1}{2}}\epsilon_t, \quad \epsilon_t \sim N(0, I_{n\times n}) \qquad (14)$$

where τ is the step-length for each iteration. ULA is simple in its implementation, yet it introduces a bias, then we need to introduce the acceptance-rejection step through the MH algorithm. By introducing MH algorithm into ULA, we will obtain the Metropolis-Adjusted Langevin algorithm (MALA), (Izzatullah et al., 2020, Izzatullah et al., 2021). The procedure consists of constructing a Markov chain at each step t, given $\widetilde{\mathbf{m}}$, a new observation *m* is generated from a proposal density q(**m**). The candidate is then accepted with probability p_{accept} given by,

The procedure consists of constructing a Markov chain at each step *t*, given $\widetilde{\mathbf{m}}$, a new observation $\mathbf{m}$ is generated from the candidate density $\rho(\mathbf{m})$. The candidate value is accepted with probability,

$$p_{accept} = \min\left(1, \frac{L(\widetilde{\mathbf{m}})\,q(\widetilde{\mathbf{m}}, \mathbf{m_{t-1}})}{L(\mathbf{m})\,q(\mathbf{m_{t-1}}, \widetilde{\mathbf{m}})}\right) \qquad (15)$$

In summary, MALA algorithm is obtained s follows: Combining the MH and ULA algorithms, the MALA MCMC algorithm is obtained and the general steps for MALA MCMC is presented below:

1. Choose an initial solution $\mathbf{m_{prior}}$ and the discretization step-length τ.
2. Draw $\epsilon_t \sim N(0, I_{n\times n})$ and simulate a new sample from the Langevin diffusion:

$$\mathbf{m} = \widetilde{\mathbf{m}} + - \tau_t\Sigma\nabla \log p\,(\widetilde{\mathbf{m}}|d_{obs}) + \sqrt{2\tau_t}\,\Sigma^{-\frac{1}{2}}\epsilon_t, \qquad (16)$$

3. Compute the accept-reject probability

$$p_{accept} = \min\left(1, \frac{L(\tilde{\mathbf{m}})}{L(\mathbf{m})}\right) \quad (17)$$

4. If Draw u from a uniform distribution $u \sim U(0,1)$, if $p_{accept} < u$ then accept $m_t = \tilde{m}$, else $\mathbf{m_t} = \mathbf{m_{t-1}}$

4.5. If the proposed subsurface model, $\tilde{m}$, is better than the current subsurface model, $\mathbf{m}$, then replace $\mathbf{m}$ with $\tilde{\mathbf{m}}$. Otherwise, keep the current model as is. Then, repeat this process until convergence.

The main advantage of the MALA algorithm is that high-dimensional density samples are obtained using the gradient of the logarithm of the posterior distribution. The MALA algorithm is a MCMC method that uses simulations from the discretization by the Euler-Maruyama algorithm of an SDE whose target density has a stationary distribution. The algorithm is inspired by stochastic models of molecular dynamics and is a multivariate extension of a Metropolis random walk, including partial derivatives to improve the mixing rate. It is general purpose, has good theoretical properties, in particular, it can scale better to high-dimensional problems than standard MCMC algorithms, geometric convergence is well established, has an acceptance rate between 40-80%. One drawback is that it requires calculating a gradient at each iteration and successively evaluating the objective function.

3.3.2 MALA with locally Lipschitz adaptive step size

In the MALA algorithm, it is required to calibrate the step-size τ, because τ must decrease with dimension, *n*. then τ can be turned such that the MCMC achieve better mixing performance. An extension of ULA and similar in spirit with Stochastic Gradient Langevin Dynamics algorithm proposed by Welling and Teh, (2011) by suppressing the MH acceptance steps. In (Izzatullah et al., 2021) propose ULA with the step-length τ based on the Lipschitz condition,

$$\tau_t = \frac{1}{2}\frac{|m_{t+1} - m_t|_2}{|\nabla \log p\,(m_t|d_{obs}) - \nabla \log p\,(m_t|d_{obs})|_2} \quad \frac{|\mathbf{m}_{t+1} - \mathbf{m}_t|_2}{|\nabla \log \mathrm{p}\,(\mathbf{m}_{t+1}|\mathrm{d_{obs}}) - \nabla \log \mathrm{p}\,(\mathbf{m}_t|\mathrm{d_{obs}})|_2}$$

The general steps for MALA MCMC with locally Lipschitz adaptive step size are:

1. Choose an initial solution $\mathbf{m}_{\text{prior}}$, the discretization step-length τ, $\beta_0 = +\infty$ and $L_c = N_{\boldsymbol{m}}^{-1/3}$.
2. Draw $\epsilon_t \sim N(0, I_{n\times n})$ and simulate a new sample from the Langevin diffusion:

$$\mathbf{m} = \tilde{\mathbf{m}} - \tau_t \Sigma \nabla \log \mathrm{p}\,(\tilde{\mathbf{m}}|\mathrm{d_{obs}}) + \sqrt{2\tau_t}\,\Sigma^{-\frac{1}{2}}\epsilon_t, \quad (19)$$

5.3. Compute the accept-reject probability

$$p_{accept} = \min\left(1, \frac{L(\tilde{\mathbf{m}})}{L(\mathbf{m})}\right) \quad (20)$$

6. Draw $u \sim U(0,1)$, if $p_{accept} < u$ then accept $\mathbf{m_t} = \tilde{\mathbf{m}}$, then update, If the proposed subsurface model, $\tilde{\mathbf{m}}$, is better than the current subsurface model, $\mathbf{m}$, then replace $\mathbf{m}$ with $\tilde{\mathbf{m}}$ and update.

$$\tau_t = \min\left\{\sqrt{1+\beta_{t-1}}\,\tau_{t-1}, L_c \frac{\|\mathbf{m_t} - \mathbf{m_{t-1}}\|}{|\Sigma\nabla \log p\,(\mathbf{m_t}|\mathrm{d_{obs}}) - \Sigma\nabla \log p\,(\mathbf{m_{t-1}}|\mathrm{d_{obs}})|}\right\} \quad (21)$$

$$\beta_t = \frac{\tau_t}{\tau_{t-1}} \tag{22}$$

7.4. Else reject $m_t = m_{t-1}$. Otherwise, keep the current model as is. Then, repeat this process until convergence.

4 The AVO method

The AVO method was created in the early 1980s to analyze the amplitudes of seismic CMP gathers as a function of angle to find hydrocarbons. The Aki-Richards equation (Aki and Richards, 2002 is the foundation of AVO analysis. The original form of the equation can be rewritten for a weak-contrast interface to give (Buland and Omre, 2003; Niu et al., 2020):

$$\mathrm{R_{pp}}(\theta) = \mathrm{c_1}(\theta)\frac{\Delta \mathrm{V_p}}{\overline{\mathrm{V_p}}} + \mathrm{c_2}(\theta)\frac{\Delta \mathrm{V_s}}{\overline{\mathrm{V_s}}} + \mathrm{c_3}(\theta)\frac{\Delta\rho}{\bar{\rho}}, \tag{23}$$

where

$$\mathrm{c_1}(\theta) = \frac{1}{2}(1 + \tan^2\theta), \tag{24}$$

$$\mathrm{c_2}(\theta) = -4\frac{\overline{\mathrm{V_s}}}{\overline{\mathrm{V_p}}}\sin^2\theta, \tag{25}$$

$$c_3(\theta) = \frac{1}{2}\left(1 - 4\frac{\overline{V_s}}{\overline{V_p}}\sin^2\theta\right), \tag{26}$$

In equations (23 - 26), the incident angle θ is the angle at which a wave hits a surface. $\mathrm{V_p}, \mathrm{V_s}$ and ρ represent the velocities of P-waves, S-waves, and the density of a material, respectively. $\Delta\mathrm{V_p}, \Delta\mathrm{V_s}$ and $\Delta\rho$ are the changes in V_p, V_s and ρ across a reflective interface. $\overline{\mathrm{V_p}}, \overline{\mathrm{V_s}}$ and $\bar{\rho}$ are the average values of V_p, V_s and ρ, respectively.

To obtain the seismic trace for a certain theta angle we can use the approximation for small reflectivity (Russell et al., 2006),

$$\mathrm{T}(\theta) = \frac{1}{2}\mathrm{c_1 W}(\theta)\mathrm{DL_{V_p}} + \frac{1}{2}\mathrm{c_2 W}(\theta)\mathrm{DL_{V_s}} + \frac{1}{2}\mathrm{c_3 W}(\theta)\mathrm{DL_\rho}, \tag{27}$$

where $\mathrm{L_{V_p}} = \ln(\mathrm{V_p})$, $\mathrm{L_{V_s}} = \ln(\mathrm{V_s})$, $\mathrm{L_\rho} = \ln(\rho)$, *W* is the wavelet matrix and *D* is the derivative matrix. Equation 27 can be implemented in matrix form as

$$\begin{bmatrix} T(\theta_1) \\ T(\theta_2) \\ \vdots \\ T(\theta_n) \end{bmatrix} = \frac{1}{2}\begin{bmatrix} c_1 W(\theta_1)D & c_2 W(\theta_1)D & c_3 W(\theta_1)D \\ c_1 W(\theta_2)D & c_2 W(\theta_2)D & c_3 W(\theta_2)D \\ \vdots & \vdots & \vdots \\ c_1 W(\theta_n)D & c_2 W(\theta_n)D & c_3 W(\theta_n)D \end{bmatrix}\begin{bmatrix} L_{V_p} \\ L_{V_s} \\ L_\rho \end{bmatrix} \tag{28}$$

A practical approach to solve equation 28 is to initialize the solution to, $[L_{V_p} \quad L_{V_s} \quad L_\rho]^T = [L_{V_{p0}} \quad L_{V_{s0}} \quad L_{\rho_0}]^T$ where $\mathrm{L_{V_{p0}}}, \mathrm{L_{V_{s0}}}$ and $\mathrm{L_{\rho_0}}$ is the prior model for P-wave and S-wave velocities and bulk density respectively, and then to iterate towards a solution using in our case MH, HMC, MALA and Lip-MALA.

5 Results

5.1 Synthetic test

We test our algorithms with using noise-free synthetic seismic traces that were obtained from real data of V_p, V_s and ρ for which synthetic seismic traces were generated from the equation 28 for the angles $\theta_1 = 9^o, \theta_2 = 18.5^o$ and $\theta_3 = 27.5^o$ and these synthetic seismic traces will be our observed data. We ran the sampling algorithms described in section 3, producing a large chain of realizations, starting from a prior model configuration corresponding to a low frequency model of V_p, V_s and ρ.

Figure 1 shows the objective function variation curves for the different sampling algorithms. Each iteration involves randomly perturbing the velocities and density of a subset of layers and recalculation of seismic traces. The vertical axis represents the objective function calculated using Equation 4. The horizontal axis shows the number of steps in the Markov chain, each associated with an accepted or rejected perturbation of the velocity and density configuration. The first stage of the chain, associated with the initial configuration and large residues, is called the burn-in or warm-up stage. After subtracting the residuals, the model realizations of velocities and densities satisfactorily explain the seismic data within the data errors. This is called the sampling phase. Realizations produced during the sampling phase are treated as samples from the probability density.

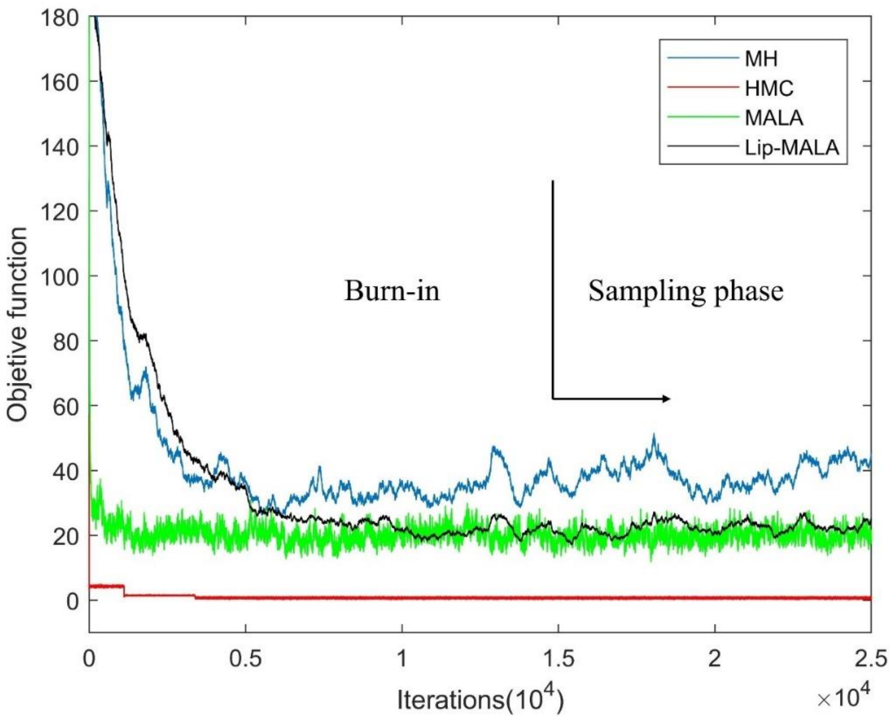


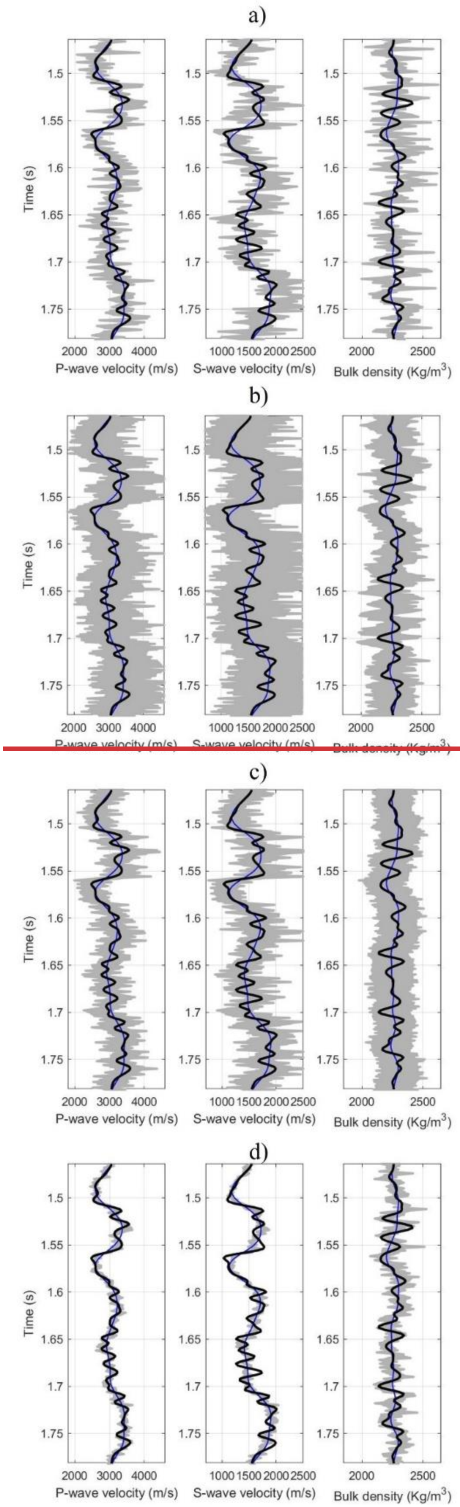
Figure 1: Progress with iterations in the MH (blue line), HMC (red line), MALA (green line) and Lip-MALA (black line) sampling algorithms for synthetic test.

The model settings were modified during the sampling phase, but remain within the probability function, as shown in Figure 2. Figure 3 shows all realizations taken (gray area) in the chain sampling phase for the different algorithms tested in this work, all adjusting the observed seismic data and within the uncertainties of the data. These realizations indicate the features and variability of the velocities and density. Table 1 shows the statistical parameters of mean and standard deviation (Sd) which we will compare then with the data obtained from the inference in the different algorithms used.

Table 1: Mean and Standard deviation of elastic parameters used in the synthetic test

Parameter	Mean	Sd
$\boldsymbol{V_p}$ (m/s)	3068.21	278.29
$\boldsymbol{V_s}$ (m/s)	1553.81	240.60
$\boldsymbol{\rho}$ (Kg/m^3)	2263.57	54.68

~~**Table 1: Mean and Standard deviation of elastic parameters used in the synthetic test**~~**.**


a)
Time (s)
P-wave velocity (m/s)
S-wave velocity (m/s)
Bulk density (Kg/m3)
b)
Time (s)
P-wave velocity (m/s)
S-wave velocity (m/s)
Bulk density (Kg/m3)
c)
Time (s)
P-wave velocity (m/s)
S-wave velocity (m/s)
Bulk density (Kg/m3)
d)
Time (s)
P-wave velocity (m/s)
S-wave velocity (m/s)
Bulk density (Kg/m3)

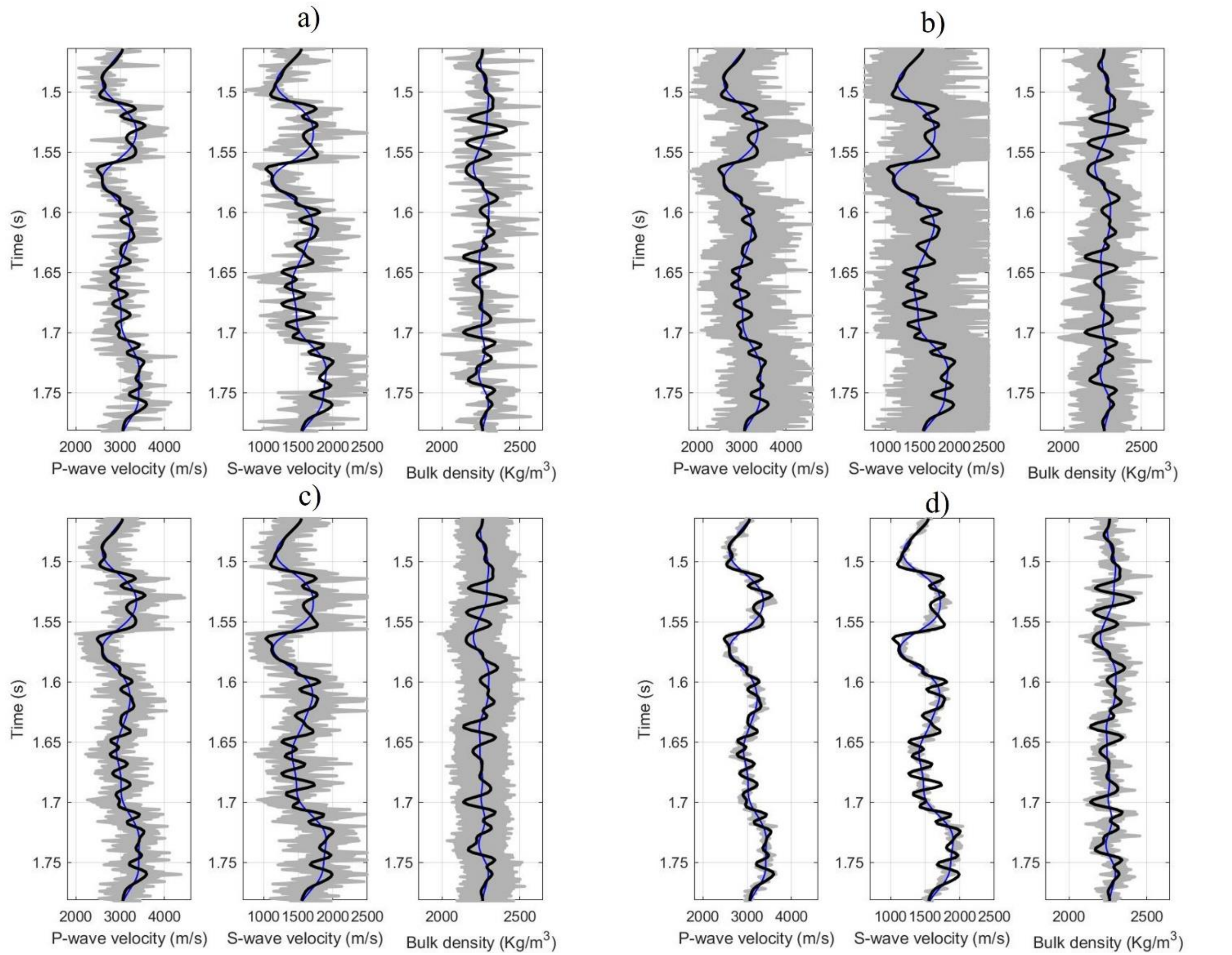
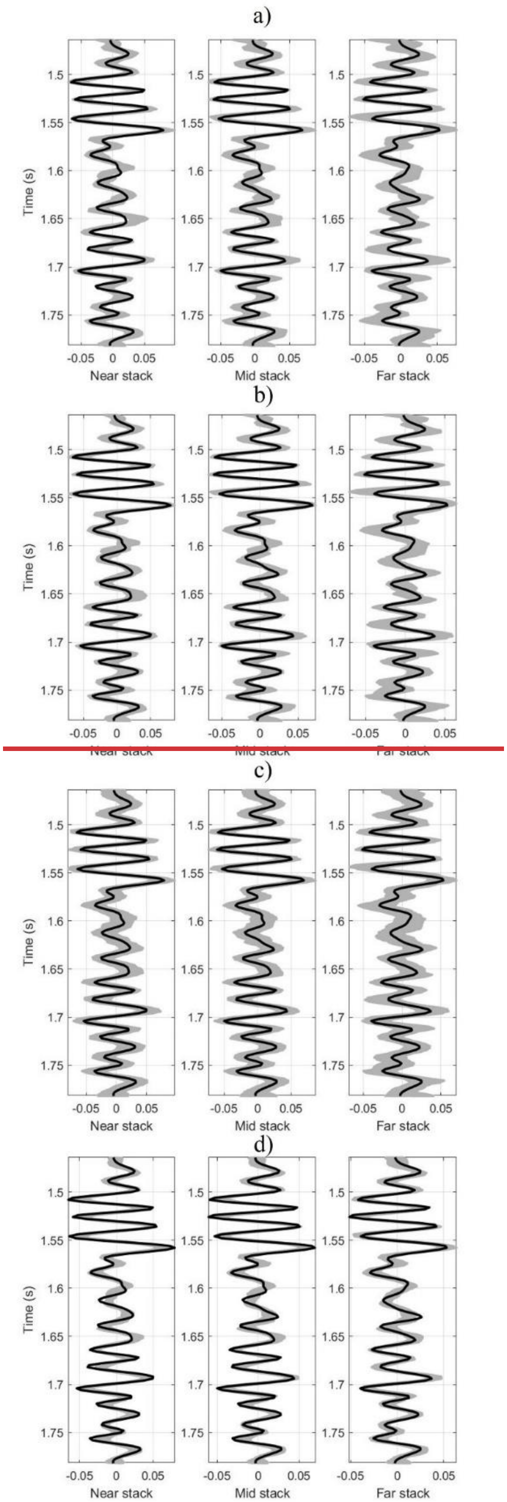


Figure 2: True data (black line), prior model (~~red~~ blue line), and accepted realizations of the model (gray) for the synthetic test~~.~~ where a) MH, b) HMC, c) MALA, and d) Lip-MALA.


a)
b)
c)
d)
Time (s)
1.5
1.55
1.6
1.65
1.7
1.75
-0.05
0
0.05
Near stack
Mid stack
Far stack

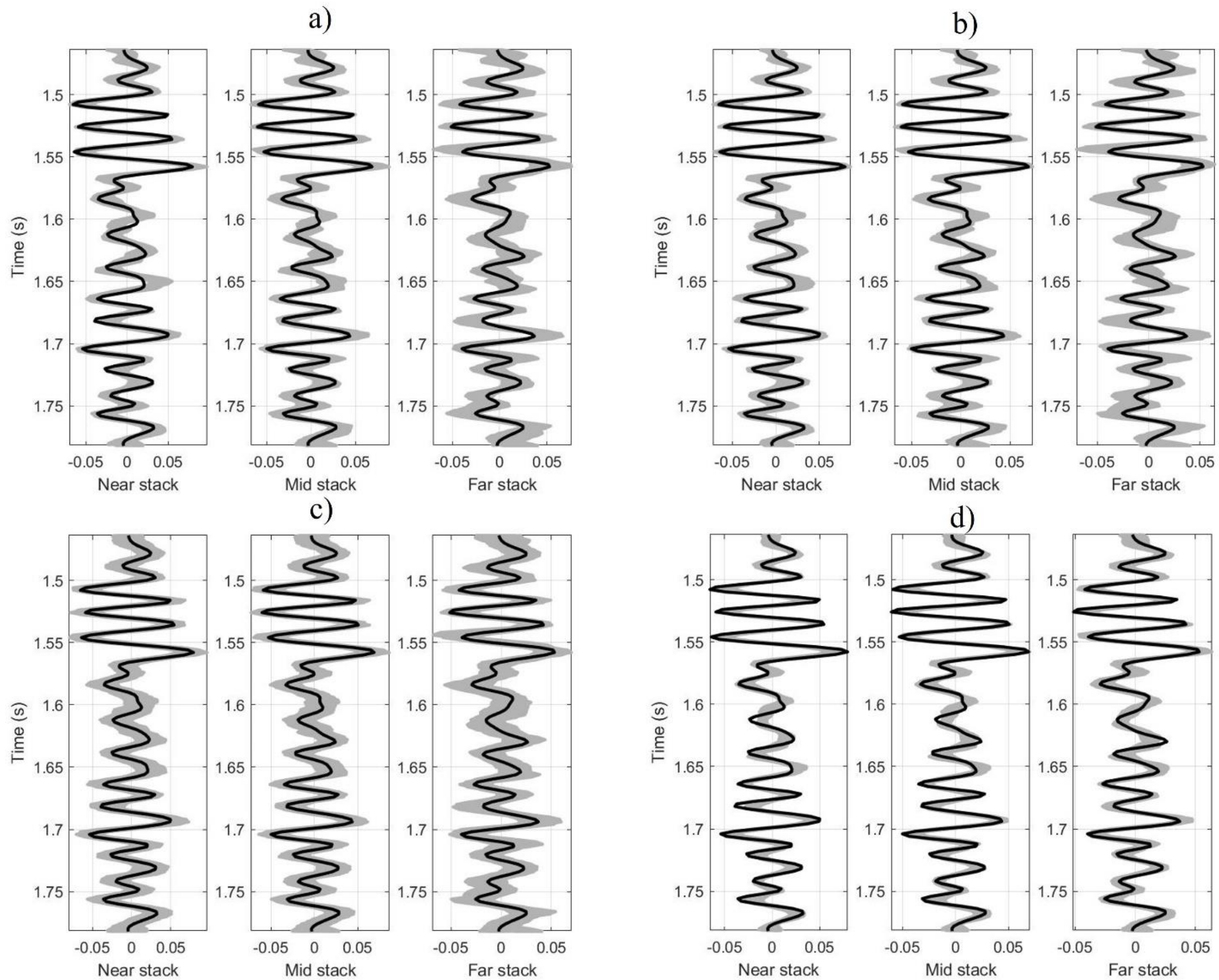
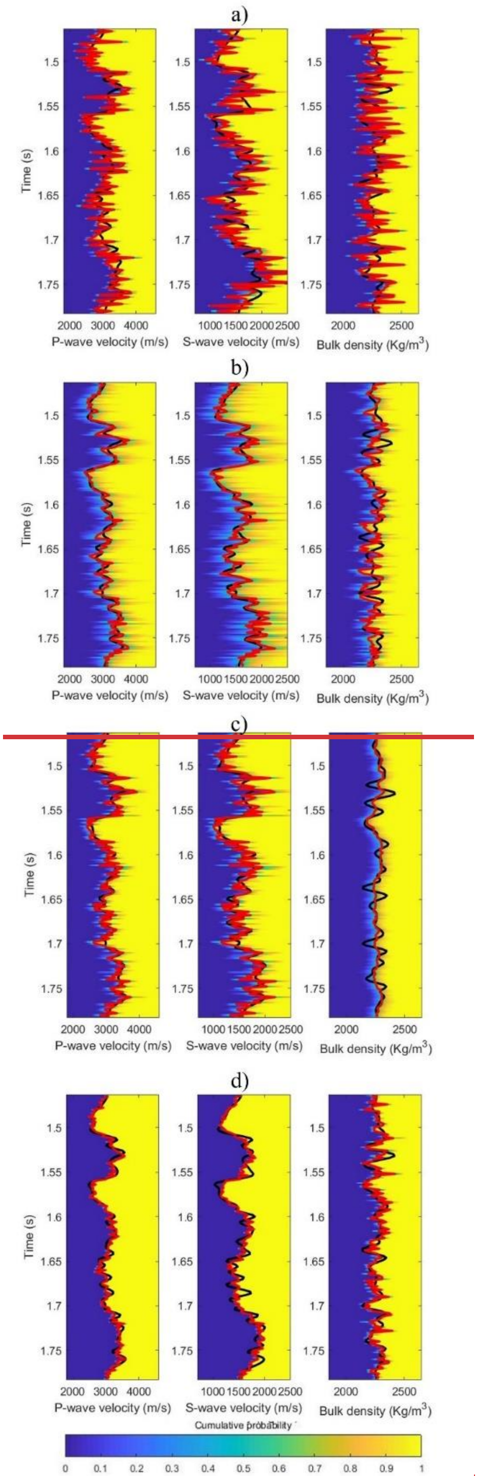


Figure 3: Observed seismic data (black line) and seismic traces obtained from the accepted model realizations (gray) for the synthetic test. where a) MH, b) HMC, c) MALA, and d) Lip-MALA.


a)
b)
c)
d)
Time (s)
1.5
1.55
1.6
1.65
1.7
1.75
2000 3000 4000
P-wave velocity (m/s)
1000 1500 2000 2500
S-wave velocity (m/s)
2000 2500
Bulk density (Kg/m^3)
Cumulative probability
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1

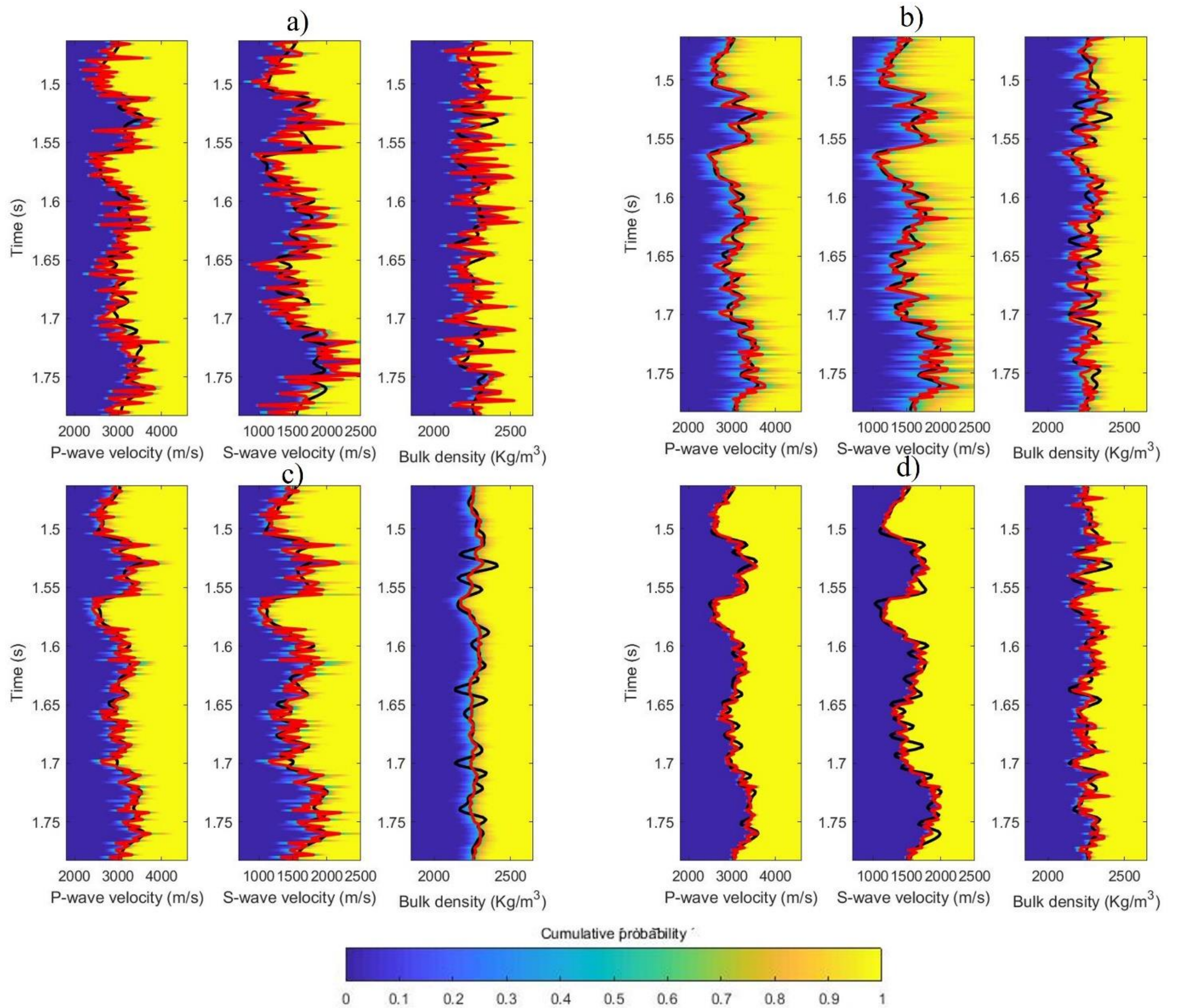


Figure 4: Marginal cumulative probability distributions (color map), true data (black line) and ~~result of~~ seismic inversion model result (red line) for the synthetic test~~.~~, where a) MH, b) HMC, c) MALA, and d) Lip-MALA.

Our chain sampling phase yielded 10,000 realizations. From these realizations, we calculated the expected values and marginal probabilities of P-wave and S-wave velocities and density as a function of two-way reflection time. These calculations were based on averaging the model performances over the sampling phase. Figure 4 presents the marginal cumulative probability distributions for P and S wave velocities and density, as estimated by the inversion, along with the actual P and S wave velocities and density of the synthetic test. The figure demonstrates the successful prediction of the actual values for all tested algorithms, accurately identifying the main stratification characterized by high and low velocities and the corresponding high and low density.

Table 2 summarizes the performance of the different algorithms tested in predicting P-wave and S-wave velocities and density. The mean, Standard Deviation (Sd), correlation, and Root Mean Squared Error (RMSE) are presented for each parameter. The mean and standard deviation values indicate that the predicted values are closely aligned with the true values. Regarding correlation, MH exhibits the lowest correlation for velocity prediction, while HMC achieves the highest. For density prediction, MH and HMC show correlations below 0.29, while MALA and Lip-MALA achieve correlations above 0.60.

In terms of RMSE, MH demonstrates the highest error for velocity prediction, while HMC achieves the lowest. For density prediction, MH and HMC exhibit errors above 75.75, while MALA and Lip-MALA maintain errors below 51.04.

Table 2: Statistical parameters for the results obtained for algorithms tested for the synthetic test.

Parameter	Mean	Sd	Corr	RMSE
MH				
V_p (m/s)	3058.90	394.79	0.64	302.96
V_s (m/s)	1577.99	359.38	0.64	277.26
ρ (Kg/m^3)	2278.00	112.32	0.29	110.39
HMC				
V_p (m/s)	3075.62	312.57	0.90	135.23
V_s (m/s)	1566.98	271.01	0.90	118.15
ρ (Kg/m^3)	2256.42	61.61	0.16	75.75
MALA				
V_p (m/s)	3051.93	326.28	0.85	174.52
V_s (m/s)	1544.13	277.40	0.80	165.30
ρ (Kg/m^3)	2262.27	29.39	0.68	40.64
Lip-MALA				
V_p (m/s)	3062.09	265.42	0.91	112.94
V_s (m/s)	1552.04	217.04	0.89	110.60
ρ (Kg/m^3)	2266.78	59.00	0.60	51.04

~~**Table 2: Statistical parameters for the results obtained for algorithms tested for the synthetic test.**~~

Table 3 presents various performance parameters, including acceptance rate and total execution time. Lip-MALA exhibits the highest acceptance rate, while HMC exhibits the lowest. Conversely, MH boasts the lowest total execution time, while HMC demonstrates the highest.

Table 3: Other parameters for synthetic test.

Method	Acceptance rate (%)	Total execution time (s)
MH	36.50	24.12
HMC	17.53	9356.99
MALA	25.28	694.54
Lip-MALA	38.49	3337.02

~~**Table 3: Other parameters for synthetic test.**~~

Finally, the convergence of the samples was analyzed a posteriori of the unknown parameters (seismic data parameters) m obtained from the different algorithms used. The multivariate effective sample size (mESS) statistic was used. The mESS is a measure that determines the size of an independent and identically distributed sample with the same covariance structure as the sample obtained from an MCMC method for the multivariate case- If we want to know if the chain converges by we can calculate minimum effective sample size (minESS) so that if mESS>minESS we say that the chain converges, if the reader is recommended to review Vats et al. (2019) to delve deeper into the convergence test used in this work. Table 4 shows the summary of mESS and minESS obtained for each method.

Table 4: Convergence test for synthetic data.

Method	mESS	minESS
MH	8150.89	7458
HMC	8561.10	7458
MALA	7472.03	7458
Lip-MALA	8119.88	7458

~~**Table 4: Convergence test for synthetic data.**~~

5.2 Application to real data

To demonstrate the effectiveness of the algorithms, we applied them to a real dataset of an oil field in eastern Venezuela. The site is located in a formation dominated by clastic rocks, a type of sedimentary rock characterized by alternating layers of sand and shale. The fluids in the pore spaces of these rocks are brine water and oil, without gas. As a preliminary step, we upscaled the P-wave and S-wave velocities obtained from well log data to the corresponding seismic scale using a bandpass filter. This process ensures that the velocity data is consistent with the frequency range of seismic waves. Table 5 presents the descriptive statistics, including mean and standard deviation (Sd), for the real data. These values will serve as a baseline for comparison with the results obtained from the inference procedures employed by the various algorithms under consideration.

Table 5: Mean and Standard deviation of elastic parameters used for real data.

Parameter	Mean	Sd
$\boldsymbol{V_p}$ (m/s)	2642.92	249.40

$\boldsymbol{V_s}$ (m/s)	1289.86	205.84
$\boldsymbol{\rho}$ (Kg/m^3)	2180.06	111.89

~~Table 5: Mean and Standard deviation of elastic parameters used for real data.~~

The seismic traces were obtained from partial stacks for the angles θ_1 = 19º, θ_2 = 24º, and θ_3 = 29º. Utilizing V_p, V_s and ρ logs in seismic scale and wavelets were extracted from the partial stacked seismic data using the frequency content of the data, the synthetic trace was generated using equation 28. The synthetic trace obtained was correlated with observed traces for seismic well tie (see figure 5) obtaining a correlation value of 0.55. The sampling algorithms described in section 3 were implemented, generating a large chain of realizations starting from a prior model configuration corresponding to a low-frequency model of V_p, V_s and ρ.

As depicted in figure 6~~5~~, the objective function variation curves for each sampling algorithm are presented. During each iteration, a subset of layers undergoes a random perturbation of their velocities and density, followed by a recalculation of the seismic trace. The objective function, calculated using equation 4, is represented on the vertical axis, while the horizontal axis represents the number of steps in the Markov chain. Each step corresponds to an accepted or rejected perturbation of the velocities and density configuration.

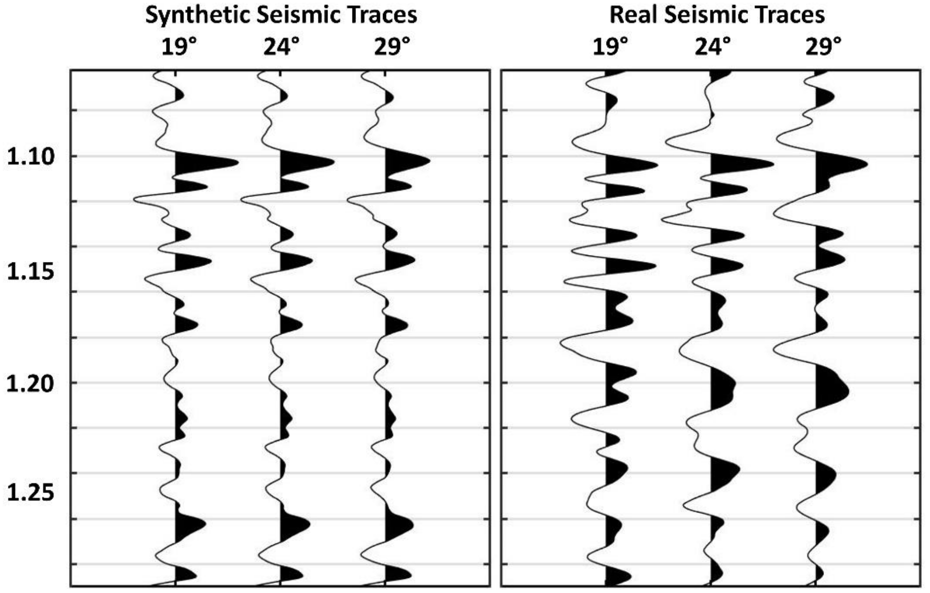


Figure 5: Seismic well tie for real data used.

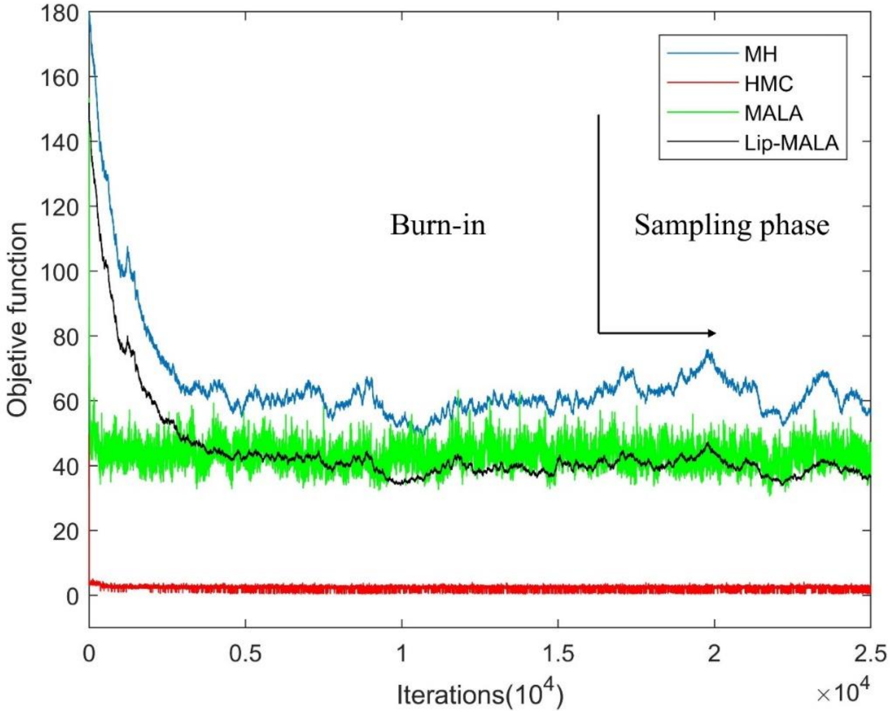
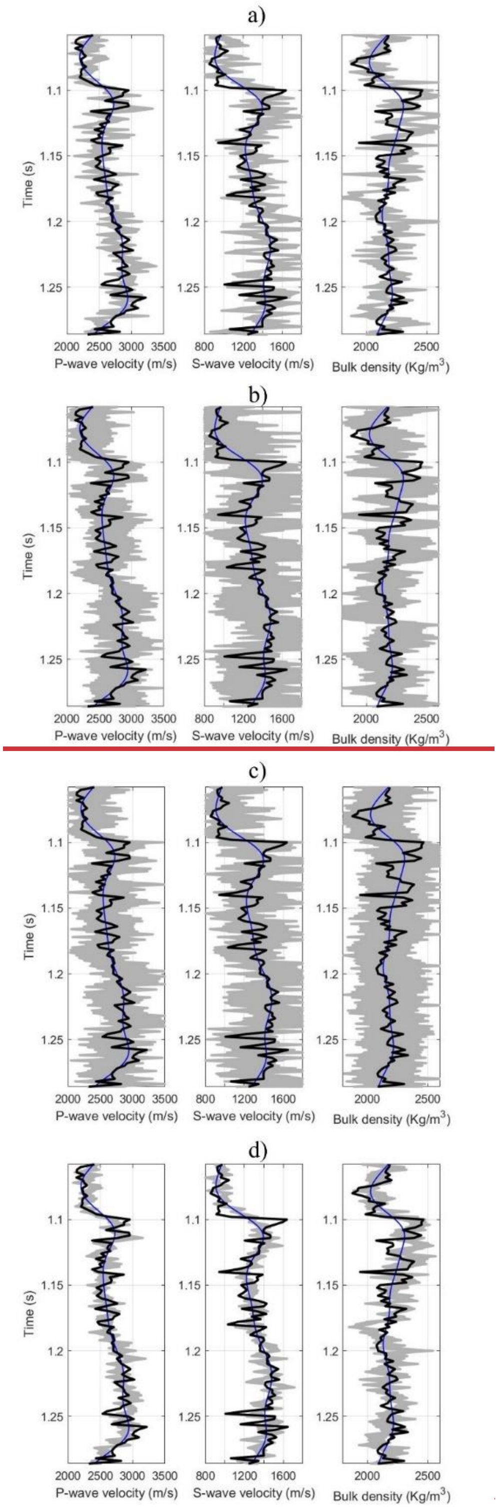


Figure ~~5~~ 6: Progress with iterations in the MH (blue line), HMC (red line), MALA (green line) and Lip-MALA (black line) sampling algorithms for real data.


a)
b)
c)
d)
Time (s)
1.1
1.15
1.2
1.25
2000 2500 3000 3500
P-wave velocity (m/s)
800 1200 1600
S-wave velocity (m/s)
2000 2500
Bulk density (Kg/m^3)

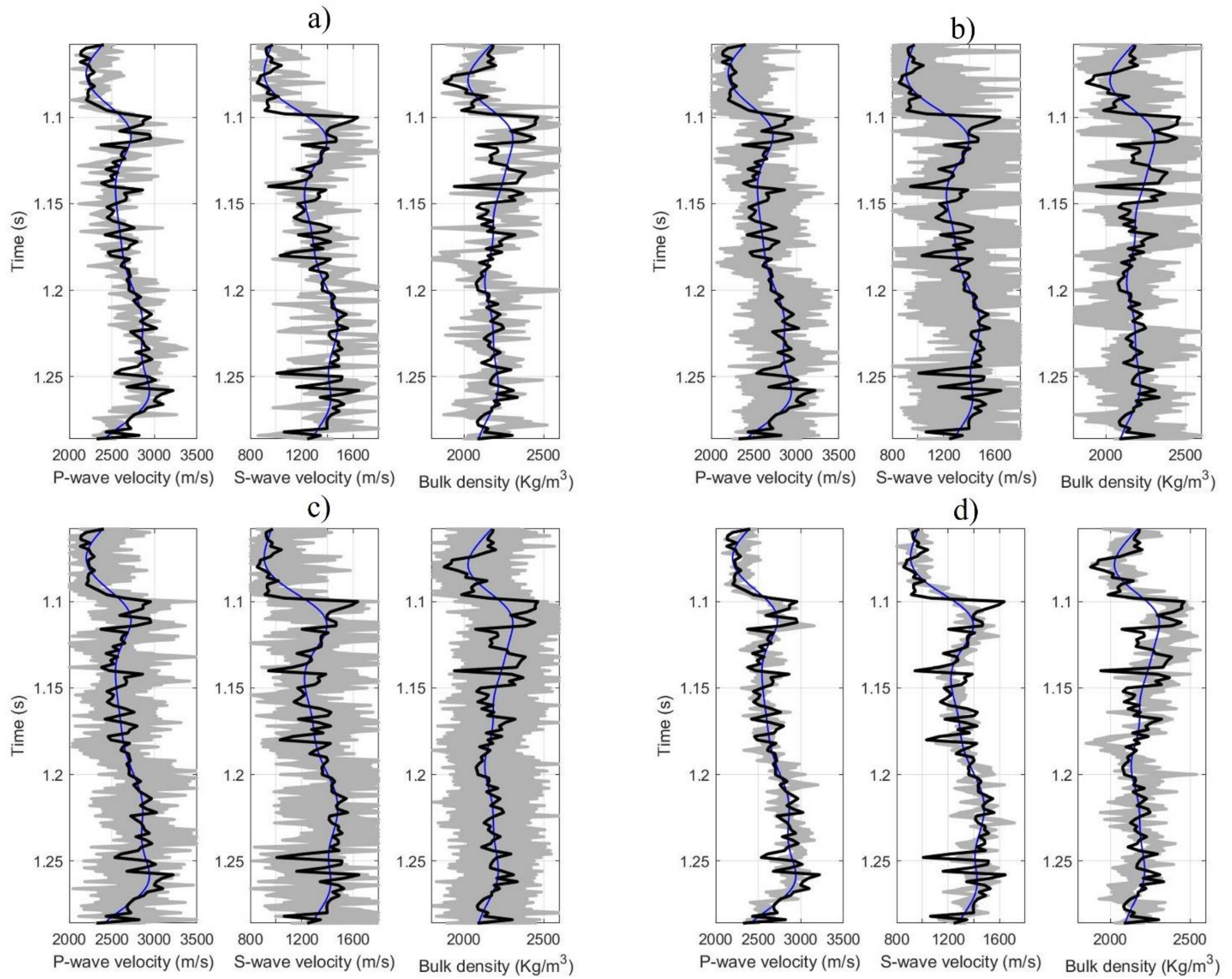
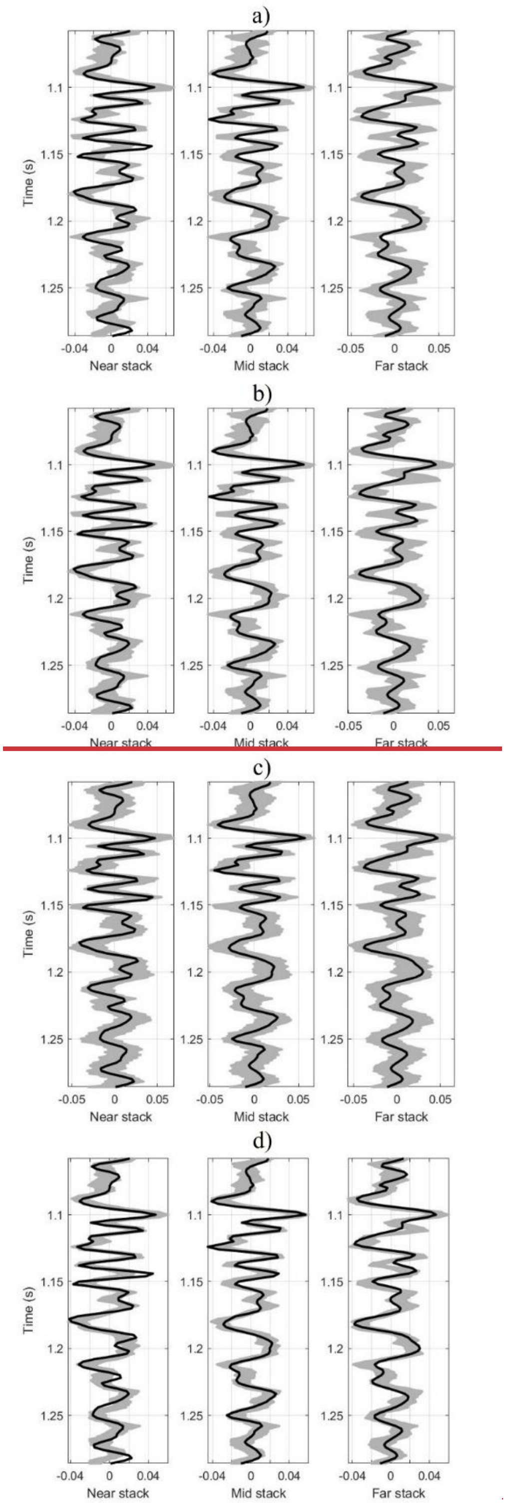


Figure ~~6~~ 7: ~~True data (black line), prior model (red line) and accepted realizations (gray) for real data.~~ True data (black line), prior model (red blue line), and accepted realizations of the model (gray) for real data where a) MH, b) HMC, c) MALA, and d) Lip-MALA.


a)
Time (s)
1.1
1.15
1.2
1.25
-0.04 0 0.04
Near stack
-0.04 0 0.04
Mid stack
-0.05 0 0.05
Far stack
b)
Time (s)
1.1
1.15
1.2
1.25
-0.04 0 0.04
Near stack
-0.04 0 0.04
Mid stack
-0.05 0 0.05
Far stack
c)
Time (s)
1.1
1.15
1.2
1.25
-0.05 0 0.05
Near stack
-0.05 0 0.05
Mid stack
-0.05 0 0.05
Far stack
d)
Time (s)
1.1
1.15
1.2
1.25
-0.04 0 0.04
Near stack
-0.04 0 0.04
Mid stack
-0.04 0 0.04
Far stack

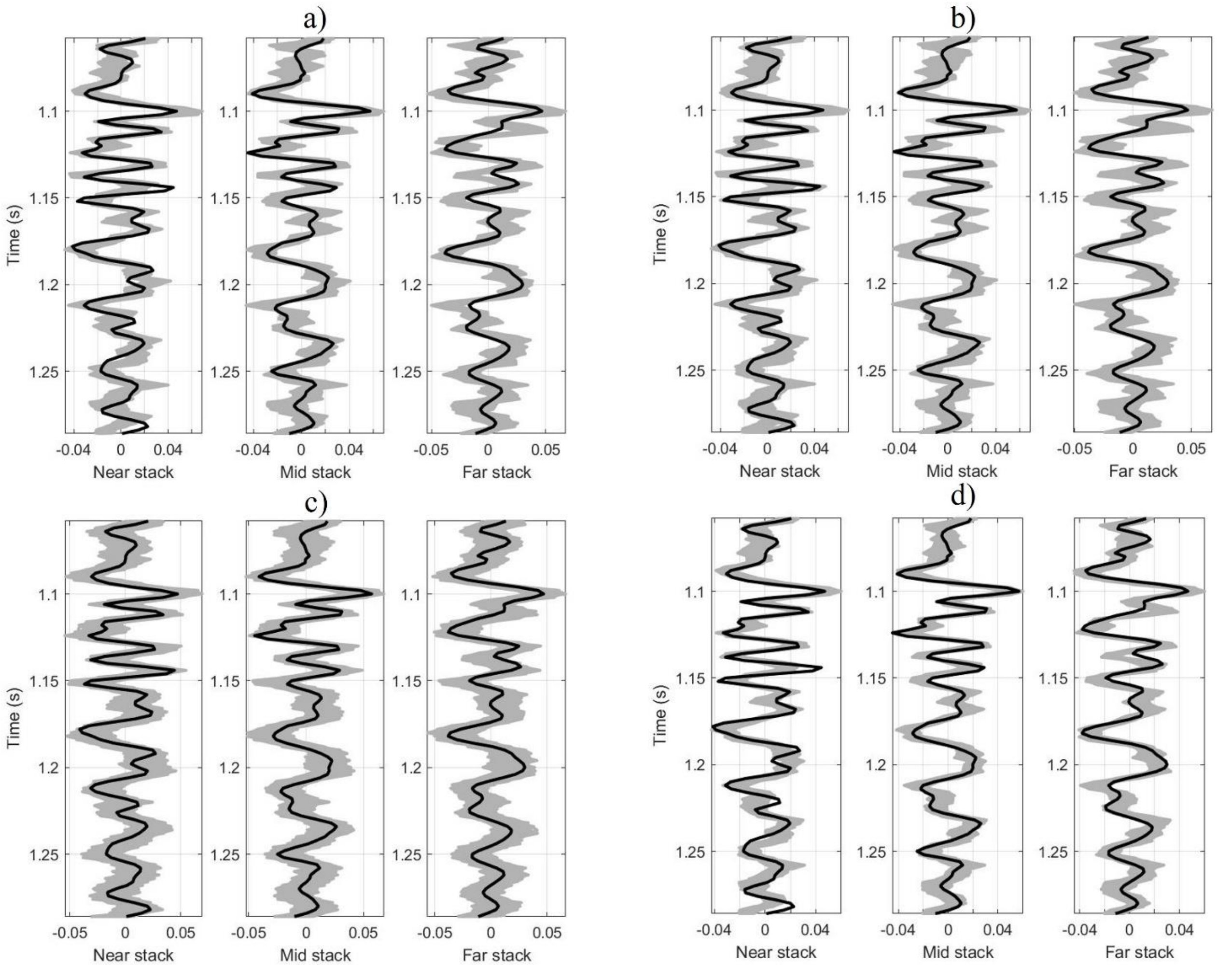
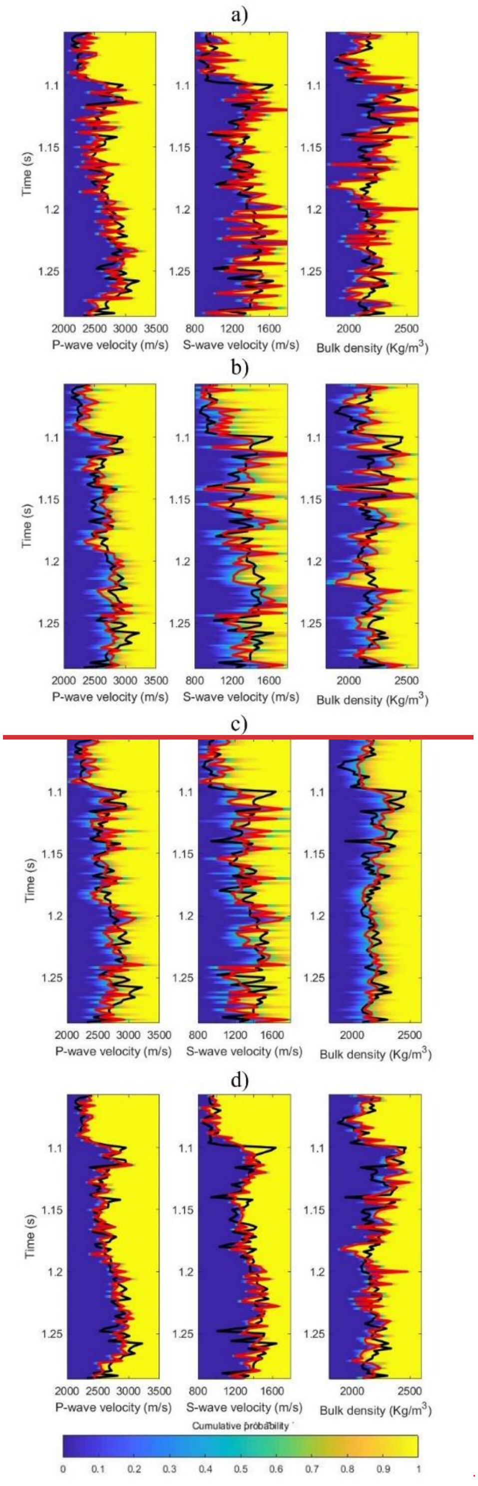


Figure ~~7~~8: ~~Observed data (black line) and accepted realizations (gray) for real data.~~ Observed seismic data (black line) and seismic traces obtained from the accepted model realizations (gray) for real data where a) MH, b) HMC, c) MALA, and d) Lip-MALA.


a)
b)
c)
d)
Time (s)
1.1
1.15
1.2
1.25
2000 2500 3000 3500
P-wave velocity (m/s)
800 1200 1600
S-wave velocity (m/s)
2000 2500
Bulk density (Kg/m3)
Cumulative probability
0 0.1 0.2 0.3 0.4 0.5 0.6 0.7 0.8 0.9 1

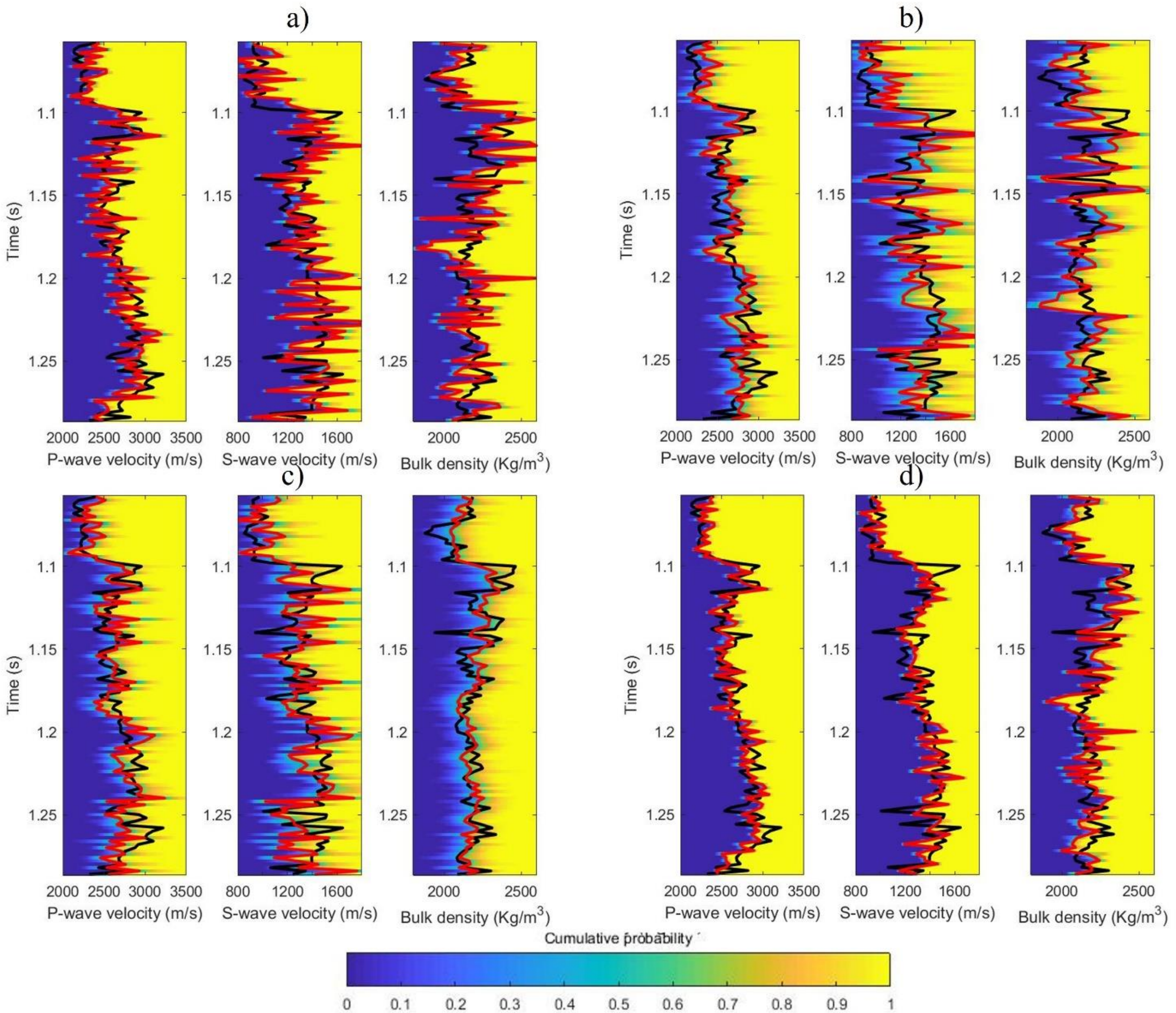


Figure ~~9~~8: ~~Marginal cumulative probability distributions (color map), true data (black line) and result of seismic inversion (red line) for real data.~~ Marginal cumulative probability distributions (color map), true data (black line) and seismic inversion model result (red line) for real data where a) MH, b) HMC, c) MALA, and d) Lip-MALA.

The model settings were adjusted during the sampling phase, ensuring they remained within the probability function (Figure ~~6~~7). Figure ~~7~~8 illustrates all realizations sampled (gray area) in the chain sampling phase for the various algorithms tested in this study, all of which align with the observed seismic data and fall within the data's uncertainty bounds. These realizations highlight the characteristics and variability of the velocities and density.

Employing a chain sampling scheme, we generated 9,000 realizations from which we extracted the expected values and marginal probabilities of P-wave and S-wave velocities and density, all as functions of two-way reflection time. These calculations were derived by averaging the model performances across the sampling phase. Figure ~~8~~9 depicts the marginal cumulative probability distributions for P and S wave velocities and density, as inferred from the inversion process, alongside the actual P and S wave velocities and density of the synthetic test.

Table 6 summarizes the performance of the tested algorithms in predicting P-wave and S-wave velocities and density. The mean, Standard Deviation (Sd), correlation, and Root Mean Squared Error (RMSE) are presented for each parameter. The predicted values closely align with the true values as evidenced by the mean and standard deviation values. MH exhibits the lowest correlation for velocity prediction, while Lip-MALA achieves the highest. For density prediction, MH and HMC show correlations below 0.28, while MALA and Lip-MALA achieve correlations above 0.48. MH demonstrates the highest error for velocity prediction, while Lip-~~MALA achieves~~MALA achieves the lowest. For density prediction, MH and HMC exhibit errors above 151.41, while MALA and Lip-MALA maintain errors below 122.22.

Table 6: Statistical parameters for the results obtained for algorithms tested for real data.

Parameter	Mean	Sd	Corr	RMSE
MH				
V_p (m/s)	2634.66	255.65	0.64	215.01
V_s (m/s)	1327.43	241.58	0.51	224.74
ρ (Kg/m^3)	2197.22	170.73	0.35	168.44
HMC				
V_p (m/s)	2640.91	199.32	0.69	182.23
V_s (m/s)	1307.19	218.66	0.52	207.86
ρ (Kg/m^3)	2186.65	138.72	0.28	151.41
MALA				
V_p (m/s)	2634.50	217.55	0.65	196.22
V_s (m/s)	1283.90	202.36	0.55	193.84
ρ (Kg/m^3)	2177.61	72.40	0.65	84.42
Lip-MALA				
V_p (m/s)	2642.07	223.19	0.79	155.40
V_s (m/s)	1295.25	175.54	0.75	138.46
ρ (Kg/m^3)	2194.84	125.50	0.48	122.22

~~**Table 6: Statistical parameters for the results obtained for algorithms tested for real data.**~~

Table 7 presents various performance parameters, including acceptance rate and total execution time. Lip-MALA exhibits the highest acceptance rate, while HMC exhibits the lowest. Conversely, MH boasts the lowest total execution time, while HMC demonstrates the highest.

Table 7: Other parameters for real data.

Method	Acceptance rate (%)	Total execution time (s)
MH	32.66	15.48
HMC	3.94	3970.74
MALA	7.38	292.83
Lip-MALA	37.89	1215.87

~~**Table 7: Other parameters for real data.**~~

And a final step, as in the synthetic data, was to test the convergence of the chains, this study employed a posteriori analysis to assess the convergence of samples obtained for the unknown seismic data parameters (denoted by m) using various algorithms. The multivariate effective sample size (mESS) statistic served as the convergence metric. The mESS quantifies the equivalent size of an independent and identically distributed (iid) sample possessing the same covariance structure as the sample generated by a Markov Chain Monte Carlo (MCMC) method in the multivariate case.

To formally determine chain convergence, a minimum effective sample size (minESS) threshold can be established. If the mESS value surpasses the minESS threshold, convergence is achieved. For a more in-depth exploration of the convergence test employed in this work, readers are referred to Vats et al. (2019). Table 8 summarizes the mESS and minESS values obtained for each method.

Table 8: Convergence test for real data.

Method	mESS	minESS
MH	7936.83	7555
HMC	10405.54	7555
MALA	10146.90	7555
Lip-MALA	7979.45	7555

~~**Table 8: Convergence test for real data.**~~

6 Discussion

This study presents a comparative analysis of Markov Chain Monte Carlo (MCMC) methods for estimating elastic properties from seismic amplitudes. We demonstrate the application of these methods in a field case, employing the following assumptions: (1) a one-dimensional reservoir model represented by stacked seismic traces, (2) seismic data simulation using the small reflectivity approximation, and (3) the Aki-Richards equation for weak contrast to establish the relationship between seismic data and elastic parameters. Notably, the proposed general formulation transcends these assumptions, allowing for the integration of more sophisticated seismic simulation techniques and comprehensive petrophysical models within a similar framework.

The four methods studied demonstrate acceptable performance, but in-depth analysis reveals notable differences:

- Velocity estimation: In both the synthetic and real-world scenarios, methods that incorporate gradient calculations (HMC, MALA, and Lip-MALA) outperform MH in estimating velocities.
- Density estimation: Density estimation proves to be the most challenging parameter, with MH and HMC exhibiting unsatisfactory results. However, MALA and Lip-MALA showcase more promising performance.
- Execution time: A significant difference emerges in execution time between methods. MH and MALA exhibit shorter execution times compared to HMC and Lip-MALA, which are considerably more time-consuming.

A natural progression of this research would be to invert prestack seismic data to extract additional elastic parameters and reservoir properties, revealing a more comprehensive subsurface understanding. Similarly, incorporating well log conditioning into the model holds promise, as it could enhance vertical resolution near wells and guarantee that the model aligns with well data at drilling locations.

7 Conclusions

This study compares various pre-stack inversion methods under an MCMC framework for the estimation of elastic parameters. We invert pre-stacked seismic data to infer velocities (V_p, V_s) and density (ρ), which are linked to the seismic data via the Aki-Richards equation. All methods employed effectively handle the inherent uncertainties associated with seismic and elastic data.

The proposed algorithms allow estimating several important aspects of the posterior distribution, such as the means and standard deviations of the posterior parameters. We rigorously validated the algorithms by measuring the quality of the MCMC sample through correlations, plotting the objective function, seismic traces and estimating the RMSE.

The four methods evaluated in this study exhibit acceptable performance overall, but a closer examination reveals notable differences in their specific capabilities. Velocity estimation: In both the simulated and real-world scenarios, methods that leverage gradient calculations (HMC, MALA, and Lip-MALA) demonstrate superior performance in estimating velocities compared to MH. Density estimation: Density estimation poses the most significant challenge, with MH and HMC exhibiting unsatisfactory results. However, MALA and Lip-MALA demonstrate more promising performance in this area. Execution time: A clear distinction emerges in execution time between the methods. MH and MALA exhibit significantly shorter execution times compared to HMC and Lip-MALA, which are considerably more time-consuming.

Author contributions

RPR and SI designed the study, performed the research, analyzed data, and wrote the paper. GB and RM contributed to refining the ideas, proof the results, carrying out additional analyses, and finalizing this paper.

Competing interests

The authors declare that they have no conflict of interest.

Acknowledgements

The authors would like to thank Yachay Tech University for funding this study through the Research Project REG-INV-18-02946.

References

Aki, K., Richards, P., 2002. Quantitative Seismology. University Science Books.

~~Alder, B.J., Wainwright, T.E., 1959. Studies in Molecular Dynamics. I. General Method. The Journal of Chemical Physics 31, 459–466. doi:10.1063/1.1730376.~~

~~Betancourt, M., 2018. A conceptual introduction to hamiltonian monte carlo. arXiv:1701.02434.~~

Bishop, C.M., 2006. Pattern Recognition and Machine Learning (Information Science and Statistics). Springer-Verlag, Berlin, Heidelberg.

Bosch, M., 2004. The optimization approach to lithological tomography: Combining seismic data and petrophysics for porosity prediction. GEOPHYSICS, 69, 1272–1282. doi:10.1190/1.1801944.

Bosch, M., Cara, L., Rodrigues, J., Navarro, A., Díaz, M., 2007. A monte carlo approach to the joint estimation of reservoir and elastic parameters from seismic amplitudes. Geophysics 72. doi:10.1190/1.2783766.

Buland, A., Omre, H., 2003. Bayesian linearized avo inversion. GEOPHYSICS, 68, 185–198. doi:10.1190/1.1543206.

Chib, S., & Greenberg, E. (1995). Understanding the Metropolis-Hastings algorithm. The American Statistician, 49(4), 327–335. https://doi.org/10.2307/2684568

Clochard, V., Delépine, N., Labat, K., Ricarte, P., 2009. Post-stack versus pre-stack stratigraphic inversion for monitoring purposes: A case study for the saline aquifer of the Sleipner field. pp. 2417–2421. doi:10.1190/1.3255345.

Creutz, M., 1988. Global monte carlo algorithms for many-fermion systems. Phys. Rev. D 38, 1228–1238. doi:10.1103/PhysRevD.38.1228

de Lima, P.D.S., Corso, G., Ferreira, M.S., de Araújo, J.M., 2023. Acoustic full waveform inversion with hamiltonian monte carlo method. Physica A: Statistical Mechanics and its Applications 617, 128618. doi:10.1016/j.physa.2023.128618.

Duane, S., Kennedy, A.D., Pendleton, B.J., Roweth, D., 1987. Hybrid monte carlo. Physics Letters B 195, 216 – 222. doi:10.1016/0370-2693(87)91197-X.

Dubbeldam, D., Calero, S., Ellis, D., Snurr, R., 2016. Raspa: Molecular simulation software for adsorption and diffusion in flexible nanoporous materials. Molecular Simulation 42, 81–101. doi:10.1080/08927022.2015.1010082.

Estévez, G., Infante, S., Sáez, F., 2012. Estimación de modelos de equilibrio general en economías dinámicas por métodos de monte carlo y cadenas de markov. Revista de Matemática Teoría y Aplicaciones 19(1), 7–36. doi:10.15517/rmta.v19i1.2102.

Fichtner, A., Simuté, S., 2018. Hamiltonian monte carlo inversion of seismic sources in complex media. Journal of Geophysical Research: Solid Earth, 123, 2984–2999. doi:https://doi.org/10.1002/2017JB015249.

Fichtner, A., Zunino, A., 2019. Hamiltonian nullspace shuttles. Geophysical Research Letters 46, 644–651. doi:10.1029/2018GL080931.

Gebraad, L., Boehm, C., Fichtner, A., 2020. Bayesian elastic full-waveform inversion using hamiltonian monte carlo. Journal of Geophysical Research: Solid Earth, 125, e2019JB018428. doi:10.1029/2019JB018428.

Goodway, B., 2001. Avo and lame constants for rock parameterization and fluid detection. CSEG Recorder 26, 39–60.

Hastings, W.K., 1970. Monte carlo sampling methods using markov chains and their applications. Biometrika 57, 97–109. doi:10.1093/biomet/57.1.97.

Helland-Hansen, D., Magnus, I., Edvardsen, A., Hansen, E., 1997. Seismic inversion for reservoir characterization and well planning in the snorre field. The Leading Edge 16, 269–274. doi:10.1190/1.1437616.

Infante, S., sanchez, l., Hernández, A., 2019. Stochastic models to estimate population dynamics. Statistics, Optimization & Information Computing 7, 311–328. doi:10.19139/soic.v7i2.538.

Izzatullah, M., van Leeuwen, T., & Peter, D. (2020). Accelerated Langevin Dynamics for Bayesian Seismic Inversion. Journal of Geophysical Research: Solid Earth, 125(3), e2019JB018428. https://doi.org/10.1029/2019JB018428

Izzatullah, M., van Leeuwen, T., Peter, D., 2021. Bayesian seismic inversion: a fast sampling Langevin dynamics Markov chain Monte Carlo method. Geophysical Journal International 227, 1523–1553. doi:10.1093/gji/ggab287.

Landa, E., Treitel, S., 2016. Seismic inversion: What it is, and what it is not. The Leading Edge 35, 277–279. doi:10.1190/tle35030277.1.

Landau, L., Lifshitz, E., 1976. Mechanics: Volume 1. Course of theoretical physics, Elsevier Science.

Lemons, D.S., Gythiel, A., 1997. Paul Langevin's 1908 paper "On the Theory of Brownian Motion" ["Sur la th´eorie du mouvement brownien," C. R. Acad. Sci. (Paris) 146, 530–533 (1908)]. American Journal of Physics 65, 1079–1081. doi:10.1119/1.18725.

Ma, X., 2002. Simultaneous inversion of prestack seismic data for rock properties using simulated annealing. GEOPHYSICS 67, 1877–1885. doi:10.1190/1.1527087.

MacKay, D. J. C. (2003). Information Theory, Inference, and Learning Algorithms. Cambridge University Press.

Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., Teller, E., 1953. Equation of state calculations by fast computing machines. The Journal of Chemical Physics 21, 1087–1092. doi:10.1063/1.1699114.

Neal, R. M. (1996). Bayesian Learning for Neural Networks. Lecture Notes in Statistics, 118. Springer. https://doi.org/10.1007/978-1-4612-0745-0

Neal, R., 2012. Mcmc using hamiltonian dynamics. Handbook of Markov Chain Monte Carlo doi:10.1201/b10905-6.

Nemeth, C., Sherlock, C., Fearnhead, P., 2016. Particle metropolis-adjusted langevin algorithms. Biometrika 103, 701–717.

Niu, L., Geng, J., Wu, X., Zhao, L., Zhang, H., 2020. Data-driven method for an improved linearised AVO inversion. Journal of Geophysics and Engineering 18, 1–22. doi:10.1093/jge/gxaa065.

Nemeth, C., & Fearnhead, P. (2021). Stochastic Gradient Markov Chain Monte Carlo. Journal of the American Statistical Association, 116(533), 433–450. https://doi.org/10.1080/01621459.2020.1847120

Robert, C. P. (2016). The Metropolis–Hastings algorithm. arXiv Preprint. https://arxiv.org/abs/1504.01896v3

Roberts, G. O., & Tweedie, R. L. (1996). Exponential Convergence of Langevin Distributions and Their Discrete Approximations. Bernoulli, 2(4), 341–363. https://doi.org/10.2307/3318418

Roberts, G., Stramer, O., 2002. Langevin diffusions and metropolis-hastings algorithms. Methodology And Computing In Applied Probability 4, 337–357. doi:10.1023/A:1023562417138.

Russell, B., Hampson, D., , Bankhead, B., 2006. An Inversion Primer. CSEG RECORDER 31, 96–103.

Stuart, A., Voss, J., Wiberg, P., 2004. Conditional path sampling of sdes and the langevin mcmc method. Commun. Math. Sci. 2. doi:10.4310/CMS.2004.v2.n4.a7.

Sánchez, L., Infante, S., Griffin, V., Rey, D., 2016. Spatio-temporal dynamic model and parallelized ensemble kalman filter for precipitation data. Brazilian Journal of Probability and Statistics 30. doi:10.1214/15-BJPS297.

Sarkka, Simo & Merkatas, Christos & Karvonen, Toni. (2021). Gaussian Approximations of SDES in Metropolis-Adjusted Langevin Algorithms. 1-6. 10.1109/MLSP52302.2021.9596301.

Tarantola, A., 2005. Inverse Problem Theory and Methods for Model Parameter Estimation. Society for Industrial and Applied Mathematics.

Vats, D., Flegal, J. M., & Jones, G. L. (2019). Multivariate output analysis for Markov chain Monte Carlo. Biometrika, 106(2), 321-337.

Welling, M., Teh, Y., 2011. Bayesian learning via stochastic gradient langevin dynamics, pp. 681–688.

Wu, H., Chen, Y., Li, S., Peng, Z., 2019. Acoustic impedance inversion using gaussian metropolis–hastings sampling with data driving. Energies 12. doi:10.3390/en12142744.