Comput Geosci (2009) 13:103–122
DOI 10.1007/s10596-008-9108-8

ORIGINAL PAPER

Integrating production data under uncertainty by parallel interacting Markov chains on a reduced dimensional space

Thomas Romary

Received: 1 April 2008 / Accepted: 26 August 2008 / Published online: 16 October 2008

Abstract In oil industry and subsurface hydrology, geostatistical models are often used to represent the porosity or the permeability field. In history matching of a geostatistical reservoir model, we attempt to find multiple realizations that are conditional to dynamic data and representative of the model uncertainty space. A relevant way to simulate the conditioned realizations is by generating Monte Carlo Markov chains (MCMC). The huge dimensions (number of parameters) of the model and the computational cost of each iteration are two important pitfalls for the use of MCMC. In practice, we have to stop the chain far before it has browsed the whole support of the posterior probability density function. Furthermore, as the relationship between the production data and the random field is highly nonlinear, the posterior can be strongly multimodal and the chain may stay stuck in one of the modes. In this work, we propose a methodology to enhance the sampling properties of classical single MCMC in history matching. We first show how to reduce the dimension of the problem by using a truncated Karhunen–Loève expansion of the random field of interest and assess the number of components to be kept. Then, we show how we can improve the mixing properties of MCMC, without increasing the global computational cost, by using parallel interacting Markov Chains. Finally, we show the encouraging results obtained when applying the method to a synthetic history matching case.

T. Romary (✉)
Reservoir Engineering Department, IFP, 1 & 4, avenue de Bois-Preau 92852 Rueil-Malmaison Cedex, France
e-mail: thomas.romary@ifp.fr

Keywords Reservoir characterization · Inverse problem · Karhunen–Loève expansion · Interacting Markov chains · History matching

1 Introduction

Conditioning the reservoir petrophysical properties, e.g., permeability or porosity, to production data, such as cumulative oil production, water cut, is a most challenging task. It consists in solving an ill-posed inverse problem: given a prior knowledge on the random field representing the petrophysical properties of the reservoir, in terms of parameters of a geostatistical model, we aim to find multiple realizations of this model that will exhibit the same dynamical behavior as the one observed. In other words, we want to sample from the posterior distribution defined in the Bayesian inversion framework. The dynamical behavior of a given field is computed by a fluid-flow simulator.

A unique realization of a conditioned field can be found using optimization techniques to find the minimum of an objective function. Some parameterization methods can simplify the problem by reducing the dimension of the space where the optimization is performed; see, e.g., [4, 13]. However, optimization methods are not relevant to cope with uncertainty in reservoir characterization. Although multiple optimizations can be performed and, hence, multiple conditioned realizations can be found by this technique, the uncertainties cannot be assessed clearly. Kitanidis [15] and Oliver [20] proposed a method to characterize uncertainty with optimization procedures, by perturbing the dynamical data. This method implies to solve a huge number of optimization problems to sample correctly

the posterior. Moreover, their method is known to work well with linear or quasilinear operators. The fluid-flow models are often highly nonlinear, however.

The Bayesian formulation of the inverse problem, see [27], shows that this problem can be viewed as a classical statistical inference problem, where we want to sample independent realizations from the posterior distribution, known up to a constant, the posterior being defined from the prior knowledge on the field given by the geostatistical model and the data with its associated measurement error.

The Monte Carlo Markov chains (MCMC) methods seem therefore particularly suited for this problem, as they are known to produce samples of virtually any posterior distribution. Two problems then arise. On one hand, the dimension of the problem is generally the number of grid blocks on which is discretized the random field, often several hundreds of thousands. Therefore, the chain has to be run for an intractable number of iterations to converge on this huge uncertainty space and to achieve an efficient sampling of the posterior. On the other hand, the computational cost of a single fluid flow simulation makes the practitioner wish to minimize the number of iterations.

MCMC methods already have a long story in history matching; see [21] for a review. More recently, [6] proposed to use the Langevin sampler coupled with Karhunen–Loève (KL) expansion and a multiscale approach. This algorithm performs a rapid convergence to the stationary regime but suffers from slow mixing as it performs only a local exploration, as we will illustrate later. In [12], the authors proposed to use adaptive sampling: adjusting the proposal distribution along the chain may improve convergence. However, in this setting, the chain can be trapped into a local mode.

In this work, we first propose a way to reduce the dimension of the inference problem using the KL [18] expansion of geostatistical models. Secondly, we propose a method to improve the global efficiency of the Markov Chain by generating a collection of chains in parallel at different temperatures and allowing them to interact.

The paper is organized as follows: In Section 2, we formalize the inverse problem and introduce some notations. In Section 3, we show how to reduce the dimension of the problem by the use of a truncated KL expansion of the random field of interest and assess the number of components to be kept. In Section 4, we first show some negative results given by classical MCMC algorithms on inverse problems through a toy example. We then introduce the principles of interacting Markov Chains. We show on the previous inverse problem example how they provide better mixing while reducing the autocorrelation along the chain, with respect to classical MCMC. In Section 5, we show the results of the application of our methodology on a synthetic history-matching case. The paper ends with conclusions and perspectives of future work.

2 Notations

In this paper, we consider a Gaussian random field $(X_u)_{u\in\mathbf{U}}$ characterized by its mean μ and covariance Γ, and indexed on $\mathbf{U}$, the physical domain of the reservoir. The permeability is considered log-normal and it is computed as e^X. The production data are denoted D^* and the measurement error is considered Gaussian with a mean zero and a covariance C_D. The production data are computed for a given random field X by a fluid-flow simulator or forward operator $F(X)$, including the exponential function. With these notations, as given in [27], the posterior distribution is known up to a multiplicative constant independent of X:

$$P(X|D^*) \propto e^{\left(-\frac{1}{2}\|D^*-F(X)\|^2_{C_D^{-1}}-\frac{1}{2}\|X-\mu\|^2_{\Gamma^{-1}}\right)}, \tag{1}$$

where $e^{\left(-\frac{1}{2}\|X-\mu\|^2_{\Gamma^{-1}}\right)}$ represents the prior probability distribution up to a constant and $e^{\left(-\frac{1}{2}\|D^*-F(X)\|^2_{C_D^{-1}}\right)}$ is termed the likelihood function in the literature of inverse problems. The latter measures the misfit between the production data computed for a given realization X, $F(X)$, and the one observed D^*.

3 Reducing the dimension with Karhunen–Loève expansion

The KL expansion [18] is a parameterization method based on the structural parameters of the random field of interest, namely, its covariance operator. It consists in representing the random field in a truncated basis of its covariance operator eigenfunctions. This method as been discussed as long ago as 1976; see [8]. Then, it was addressed in [22] in a more realistic framework. This method is regaining popularity in reservoir characterization applications since powerful computing facilities allow us to employ it. Recently, such an application has been done in [25], but addressing the problem of choosing the number of components in a minimalist way and considering the components of a non-Gaussian random field as independent. An interesting work has been done using it in [6], where it has been used conjointly with a MCMC method. We can also cite [5] where it is used in the ensemble Kalman filtering

framework. In a previous work [24], we attempted to assess the number of components that have to be used in a fluid-flow application. After a brief summary of the underlying theory, we recall some of our results.

3.1 Brief summary of the underlying theory

The principle of the KL expansion for Gaussian random field is mainly based on the concept of *reproducing kernel Hilbert space* (RKHS). Given X_u, $u \in \mathbf{U}$ a zero mean second order real process with covariance Γ, the RKHS $\mathcal{H}$ of X, or of kernel Γ, is the space of real functions on $\mathbf{U}$ so that:

1. The vectorial space $\mathcal{H}_0$ generated by the functions $\Gamma(u, .)$ $u \in \mathbf{U}$ is dense in $\mathcal{H}$.
2. For all function h of $\mathcal{H}$,

   $$h(u) = < h(\cdot), \Gamma(u, \cdot) >,$$

   where $< h(\cdot), g(\cdot) >= \int_{\mathbf{U}} h(u)g(u)du$ is the scalar product on $\mathcal{H}$.

$\mathcal{H}$ is isomorphic to the space $\mathbf{H}$ generated by the random variables X_u, $u \in \mathbf{U}$. Furthermore, if (ϕ_i) is an orthogonal family of $\mathcal{H}$, for all family (ξ_i) of independent random variables with law $\mathcal{N}(0, 1)$, the process X defined by:

$$X_u(\omega) = \sum_{i \in I} \xi_i(\omega)\phi_i(u), \ \omega \in \Omega,$$

is a Gaussian process with covariance Γ.

A judicious choice of the base (ϕ_i) allows us to obtain versions with interesting properties. Particularly, the KL's theorem [18] proposes using the family of $\Gamma(u, .)$ eigenfunctions as an orthonormal family of $\mathcal{H}$. Indeed, $\Gamma(u, .)$ being positive definite, Mercer's theorem on positive definite kernels ensures existence and uniqueness of this decomposition. The KL theorem says that, if Γ is a continuous covariance on $\mathbf{U} \times \mathbf{U}$, there exists an orthonormal basis (ϕ_i) of $\mathcal{H}$, constituted by eigenfunctions of $\Gamma(u, .)$. The corresponding basis of $\mathbf{H}$ is the family (ξ_i) of centered uncorrelated random variables. In particular, we have the following representations:

1. $\xi_i(\omega) = \int_{\mathbf{U}} X_u(\omega)\phi_i(u)\, du$.
2. $X_u(\omega) = \sum_{i \in I} \xi_i(\omega)\phi_i(u), \ \omega \in \Omega$.
3. $\mathrm{var}(\xi_i) = \lambda_i$, where λ_i is the eigenvalue associated with ϕ_i.
4. If X_u is a Gaussian process, the (ξ_i) are independent Gaussian random variables with variance λ_i.

Truncating the sum in 2 above, we can construct approximations:

$$X_u^{(M)} = \sum_{i=1}^{M} \xi_i(\omega)\phi_i(u) \tag{2}$$

of the true process that minimize the integrated mean squared error on $\mathbf{U}$: $\varepsilon_M^2 = \int_{\mathbf{U}} E(X_u^{(M)} - X_u)^2\, du$, as can be seen in [10]. Indeed, the λ_i being in decreasing order, the first related terms will represent the major part of the global variance. It is worth noticing that the components (ξ_i) are independent if and only if X is Gaussian. In the contrary case, although the (ξ_i) are uncorrelated, they are not independent, and their dependence links need to be studied carefully before to use the KL expansion. In the following paragraph, we examine the performances of this approximation in terms of variance reproduction.

3.2 Approximation with respect to a statistical criterion

In this paragraph, we show the manner in which a truncated KL expansion of a random field can reproduce the major part of the variance of a nonapproximated random field. We compute the eigendecomposition of the three following classical geostatistical covariance models, for $(u, v) \in \mathbf{U}$:

1. The exponential model: $\Gamma(u, v) = \sigma^2\, \mathrm{e}^{-3\frac{||u-v||}{a}}$
2. The normal model: $\Gamma(u, v) = \sigma^2 \mathrm{e}^{-3\frac{||u-v||^2}{a^2}}$
3. The spherical model: $\Gamma(u, v) = \sigma^2(1 - 3\frac{||u-v||}{2a} + \frac{||u-v||^3}{2a^3})\mathbb{1}_{\{||u-v||<a\}}$, where $\mathbb{1}$ is the indicator function.

As an example, Fig. 1 presents the rates of variance reproduction in function of the cumulative sum of eigenvalues, for the three models, in two dimensions, with a range $a = H/4$, where H is the size of the field side. We change the discretization step from $H/10$ to $H/60$ in order to assess its effect on the number of components.

The first thing we can see in this figure is the impact of the discretization step. Indeed, the number of components needed to reproduce, say, 90% of the total variance increases with the number of grid nodes. However, the amplitude of this phenomenon depends on the covariance model, and beyond a certain thinness of the step, this number does not increase anymore.

If we look at the curves corresponding to a discretization step of $H/60$, hence corresponding to a random field discretized on $60 \times 60 = 3{,}600$ grid blocks, we can see that the number of components necessary to reproduce 90% of the global variance for the normal, spherical, and exponential models is, respectively, 40, 300, and 1,000. This number of components is then

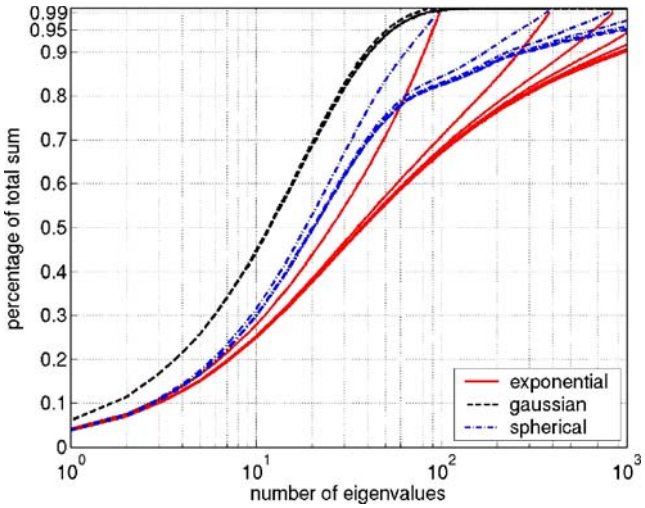


Fig. 1 Cumulative sum of eigenvalues, 2-dimensional case, *range* $= H/4$

related to the random field regularity defined by the covariance model.

Moreover, the dimension of the problem of the field also has an impact on the number of components to use. Indeed, increasing the dimension of the random field considered and, hence, of its covariance operator will translate itself into an increased number of components to reproduce a given rate of global variance.

We have shown here how a small number of components can represent a large part of the global variance of a random field. However, this criterion is somewhat abstract with respect to our application domain. Therefore, in the next section, we will examine dynamical results obtained by flow simulation on our approximated random fields.

3.3 Conditioning to raw data

Often, we dispose of raw data on the field, e.g., at well locations. We explain here briefly how to condition a Gaussian random field X, given its value at two points u and v. In this case, we consider the conditional covariance matrix given by:

$$\Gamma^c_{M-2,M-2} = \Gamma_{M-2,M-2} - \Gamma_{M-2,2}\,\Gamma^{-1}_{2,2}\,\Gamma^t_{M-2,2},$$

where: $\Gamma_{M-2,M-2}$ is the covariance matrix between unconditioned part of the field,
$\Gamma_{M-2,2}$ is the covariance matrix between the two conditioning points and the rest of the field,
$\Gamma_{2,2}$is the covariance matrix between the two conditioning points, $^{-1}$,^{t} are respectively the inverse and transpose operator.

The conditional mean is given by:

$$\mu^c = \mu + \Gamma^t_{M-2,2}\,\Gamma^{-1}_{2,2}\,X_{(u,v)},$$

where $X_{(u,v)}$is the vector of conditioning data in u and v.

We then have $X|X_{(u,v)} \sim \mathcal{N}(\mu^c, \Gamma^c)$. We then calculate the eigenvalues and eigenvectors of Γ^c.

3.4 Dynamical results

In this paragraph, we consider a 2-dimensional case test with the following properties: the permeability field is modeled by a log-normal random field, with a spherical covariance structure of range 600 m, the mean and variance the logarithm of the field are, respectively, 3.8 and 1; the field size is 2500×2500 m^2 and is discretized on a regular grid of 50×50 blocks; the thickness of the field is 10 m; we put two wells on this field: an injector at location $(3, 3)$ and a producer at $(48, 48)$; the permeability field is conditioned with respect to a permeability of 90 md at the well locations; the porosity is assumed constant at 0.25. The field is assumed to be saturated in oil at time 0. The flow is simulated with *3DSL* [26] during 10,000 days with an injection rate at 5,000 m^3/day and a pressure of 200 bars at the producer.

We performed a Monte-Carlo study, that is, we generated 100 realizations of the permeability field X, conditional to the permeability values at well locations, considering each of them at different truncation orders $X^{(M)}$. We then examined the error made on the water cut (WC) by the approximations at different orders and the true reference field. The water cut is the proportion of water in the oil produced in function of time. Hence, D^* and $F(X^{(M)})$ are both functions of time. The error ε_{WC} is then calculated by the following formula: $\varepsilon_{WC} = \sum_{t=0}^{10000}(F_t(X) - F_t(X^{(M)}))^2$. Figure 2 shows the mean and maximum relative error on the water cut in function of the number of components of the truncation on a logarithmic scale. Furthermore, assuming an allowable error of 2% on the water cut value, we compute a boundary error as the dot line in red for the mean (Fig. 2a). We also plot a boundary error of 5% for the maximum as the dot line (Fig. 2b).

We can see in Fig. 2a that it suffices, on average, to consider about 30 components to obtain an approximation error on the water cut values below the previously defined confidence level. We also see that it is useless to consider more than 50 components. Indeed, the

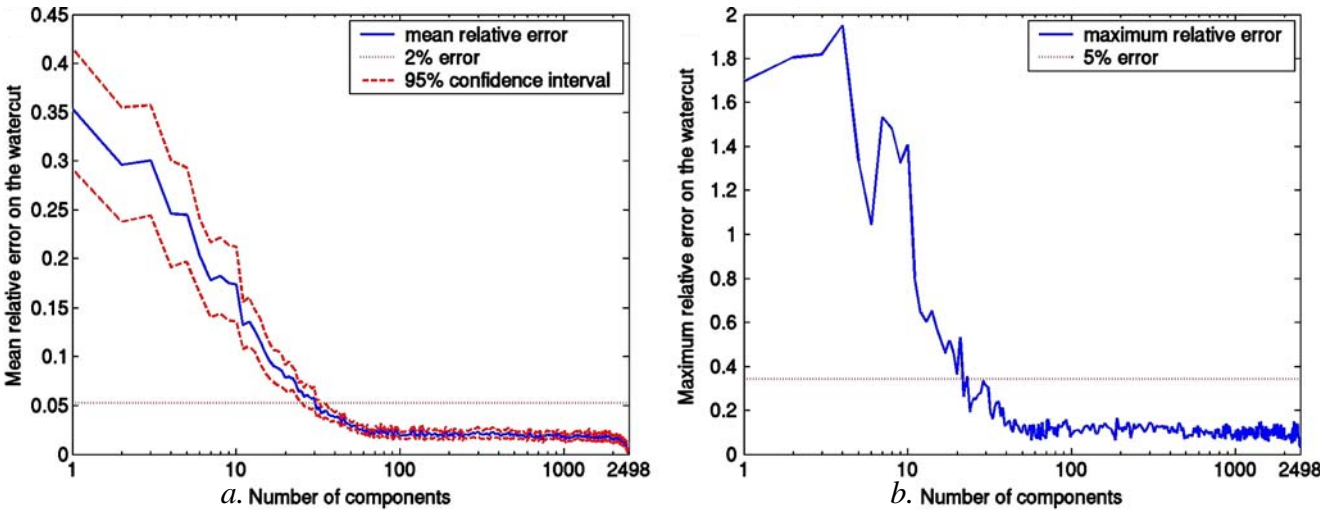


Fig. 2 Mean (**a**) and maximum (**b**) relative error on the water cut in function of the number of components

mean error stabilizes from 50 to 2,000 components, then converges quickly toward zero. The maximum relative error in Fig. 2b exhibits the same behavior. In this example, fulfilling the dynamic criterion requires much fewer components than that for reproducing 90% of the variance. Indeed, if we turn back at Fig. 1, we can see that 30 components corresponds to about 60% of the global variance and that 50 components corresponds to a bit less than 80% of it. We can then think that only this portion of the variance and the associated eigenfunctions suffices to determine the considered dynamical behavior of the flow in a permeability field modeled by such a Gaussian random field.

3.5 The eigenvalue problem

The main numerical problem induced by employing the KL decomposition is the computation of the eigenvalues and eigenvectors of the covariance operator. Formally, it takes the form of an homogeneous Fredholm integral equation of the second kind:

$$\int_{\mathbf{U}} \Gamma(u, v)e(u)\,\mathrm{d}u = \lambda e(v) \tag{3}$$

For certain classes of stochastic process, there exists an analytical solution to Eq. 3, see [10] for some examples. However, for the random fields used in petroleum engineering, Eq. 3 needs to be solved numerically.

In a first step, Eq. 3 is discretized on the grid of the field. This problem then turns out to be a linear eigenvalue problem, whose dimension is the number of grid nodes used in the discretization. Practically, it can be in the order 10^5 to 10^7. This is a very challenging problem. Although there exists numerical methods to extract some of the eigenvalues for such high dimensional problems, like [17] (of which there exists now a parallel version), the following idea can be used as a premise:

We consider a gaussian random field X with mean μ and covariance Γ. We cut this field into two blocks X_{B_1} and X_{B_2}. We can then write Γ as follows:

$$\Gamma = \left(\begin{array}{c|c} \Gamma_{B_1} & \Gamma_{B_1,B_2} \\ \hline \Gamma_{B_2,B_1} & \Gamma_{B_2} \end{array}\right),\ \mu^t = (\mu_{B_1}, \mu_{B_2})^t$$

We can simulate a realization of X in the following way:

1. simulate $X_{B_1} \sim \mathcal{N}(\mu_{B_1}, \Gamma_{B_1})$ and get a realization x_{B_1} of X_{B_1},
2. then simulate $(X_{B_2}|X_{B_1} = x_{B_1}) \sim \mathcal{N}\left(\mu_{B_2} + \Gamma^t_{B_2,B_1}\Gamma^{-1}_{B_1}x_{B_1},\ \Gamma_{B_2} - \Gamma_{B_2,B_1}\Gamma^{-1}_{B_1}\Gamma_{B_1,B_2}\right)$.

This method can be easily generalized to n blocks and can be then seen as a *sequential block simulation*. There is still a numerical difficulty, as we need to compute the inverse of the matrix Γ_{B_1}, but it is of a rather smaller dimension than the whole field.

We have shown in this section how the dimension of the history matching inverse problem can be drastically reduced by the use of a truncated KL expansion when the variable of interest is modeled by a second-order random field. This method allows us to describe approximately a discretized second-order random field with a small number of components, while preserving both spatial variability and model uncertainty space.

Its assessment on a dynamical flow test shows that only a few components govern the fluid flow behavior. This method is then of great interest for solving the history matching inversion problem. Its assessment on a dynamical flow test shows that only a few of those components govern the flow behavior.

However, different parameters can influence the number of components such as the range, the regularity of the covariance, and the dimension of the random field. Hence, a preliminary study has to be performed each time we want to use the decomposition, in order to choose the number of components with respect to a given criterion. Otherwise, this number can be determined according to the available computational resources.

Note also that a fundamental property of the KL expansion for Gaussian random field is the independence of its random components. Once the desired eigenvalues and associated eigenfunctions are known, it is straightforward to generate an approximate random field that is of capital importance for applying the MCMC algorithm.

4 MCMC for inverse problem

In this section, we first recall the main principles of MCMC. Secondly, we illustrate the drawbacks of using MCMC for inverse problems through a toy example. We finally propose a strategy to surpass these drawbacks, introducing the principles of parallel interacting Markov Chains.

4.1 General principle

The MCMC method, introduced by Metropolis et al. [19], is a popular method for generating samples from virtually any distribution π defined on $(\mathcal{X}, \mathcal{B}(\mathcal{X}))$. In particular, there is no need for the normalizing constant to be known, and the space $\mathcal{X} \subseteq \mathbb{R}^d$ (for some integer d) on which it is defined can be high-dimensional. The method consists in simulating an ergodic Markov chain $\{X_n, n \geq 0\}$ on $\mathcal{X}$ with transition probability P such that π is a *stationary* density for this chain, i.e., $\forall A \in \mathcal{B}(\mathcal{X})$:

$$\int_{\mathcal{X}} P(x, A)\pi(x)\mathrm{d}x = \pi(A) \tag{4}$$

Such samples can be used, e.g., to compute integrals

$$\pi(g) = \int_{\mathcal{X}} g(x)\pi(x)\mathrm{d}x, \tag{5}$$

estimating this quantity by

$$S_n(g) = \frac{1}{n}\sum_{i=1}^{n} g(X_i), \tag{6}$$

for some $g : \mathcal{X} \to \mathbb{R}$. We illustrate the principles of MCMC with the Metropolis–Hastings (MH) update. It requires the choice of a *proposal distribution* q. The role of q consists in proposing potential transitions for the Markov chain. Given that the chain is currently at x, a candidate y is accepted with probability $\alpha(x, y)$ defined as:

$$\alpha(x, y) = \begin{cases} \min\left(1, \frac{\pi(y)}{\pi(x)}\frac{q(y,x)}{q(x,y)}\right) & \text{if } \pi(x)q(x, y) > 0 \\ 1 & \text{else.} \end{cases} \tag{7}$$

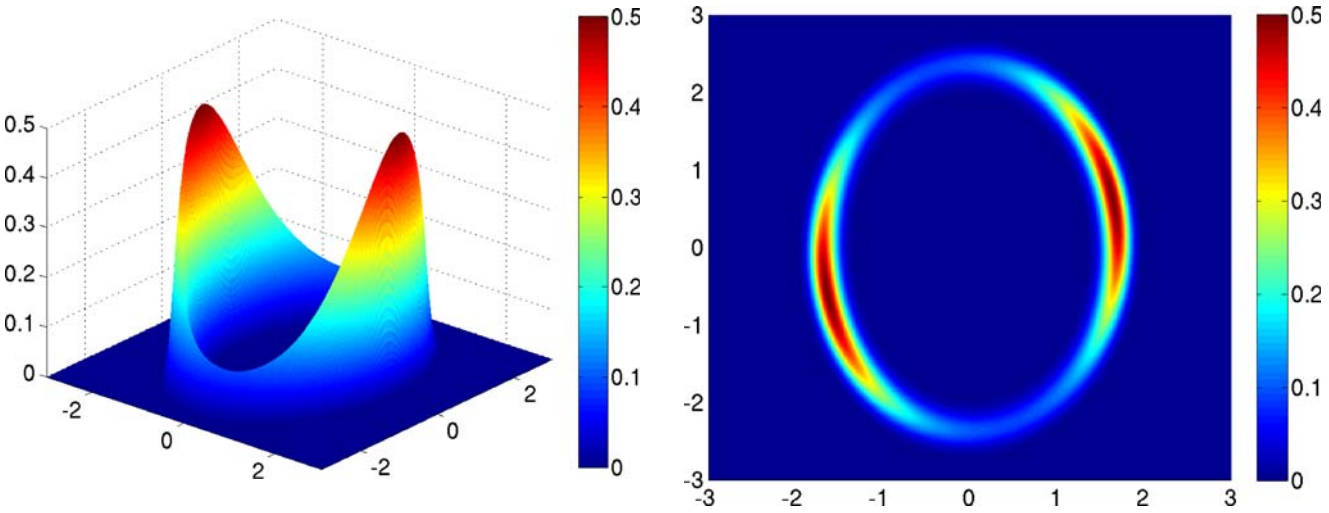
Otherwise, it is rejected and the Markov chain stays at its current location x. The transition kernel P of this Markov chain takes the form, for $(x, A) \in \mathcal{X} \times \mathcal{B}(\mathcal{X})$:

$$P(x, A) = \int_A \alpha(x, y)q(x, y)\mathrm{d}y + \mathbb{1}_A(x)\int_{\mathcal{X}} (1 - \alpha(x, y))q(x, y)\mathrm{d}y, \tag{8}$$

The Markov chain defined by P is reversible with respect to π and therefore admits π as invariant distribution. Conditions on the proposal distribution q that guarantee irreducibility and positive recurrence are easy to meet, and many satisfactory choices are possible. However, we will see in the next paragraph that the practice shows difficulty, particularly for nonlinear inverse problems purposes, where the posterior can be highly multimodal.

4.2 Performances of classical MCMC samplers on a toy example

Three important pitfalls exists in the practice of MCMC. First, MH algorithms, such as the symmetric increments random-walk sampler, the sequential independent sampler that is changing at each step a unique component of the field, or the Langevin sampler, suffer from slow mixing or even lack of convergence, especially for a small number of iterations. The second drawback of classical MCMC is the weak rate of acceptation, when we compute the empirical estimate of Eq. 7 and it generally worsens when the dimension increases. Particularly in history matching, where one iteration has a high computational cost, one wishes to have a good acceptance rate when using MCMC. The third drawback is the autocorrelations length along the chain: as the realizations are drawn from a Markov Chain, they are dependent from one another. The



a. perspective view. *b. view from the top.*

Fig. 3 Posterior distribution (**a**, **b**)

variance of the quantities (Eq. 6) we want to estimate are thus bigger than the empirical variance of the sample. In fact, the longer the autocorrelations are, the bigger the variance will be.

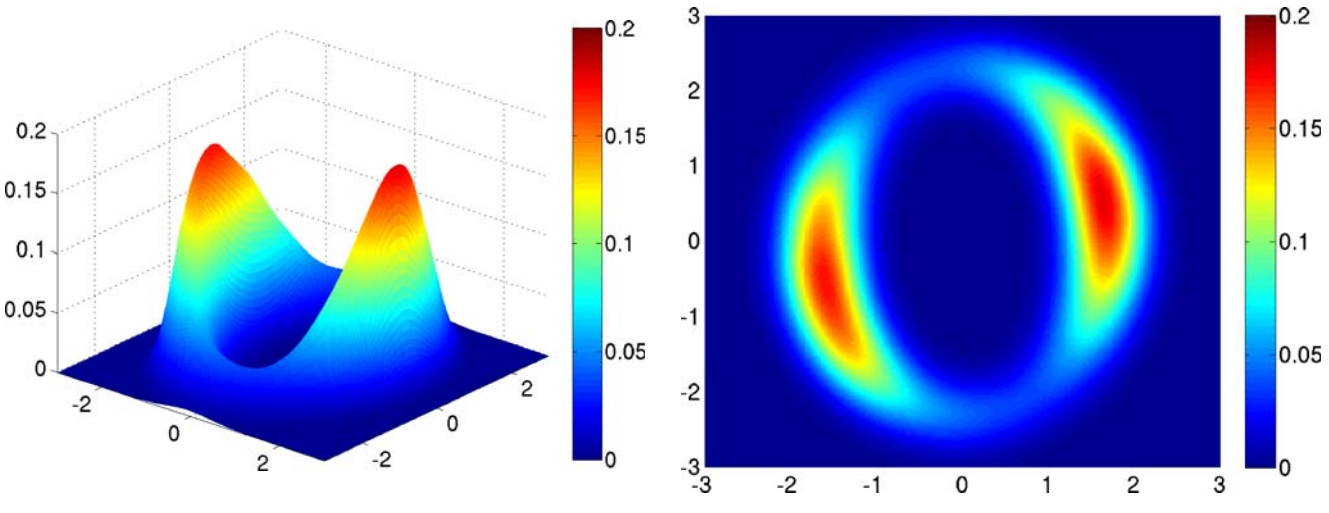
We now describe some classical MCMC samplers and set evidence on the drawbacks of using MCMC methods in inverse problems on a toy example. Our toy inverse problem is characterized by the following function:

$$\begin{aligned} F &: \mathbb{R}^2 \mapsto \mathbb{R} \\ X = (X_1, X_2) &\to 2\,X_1^2 + X_2^2, \end{aligned} \tag{9}$$

given the following prior on X:

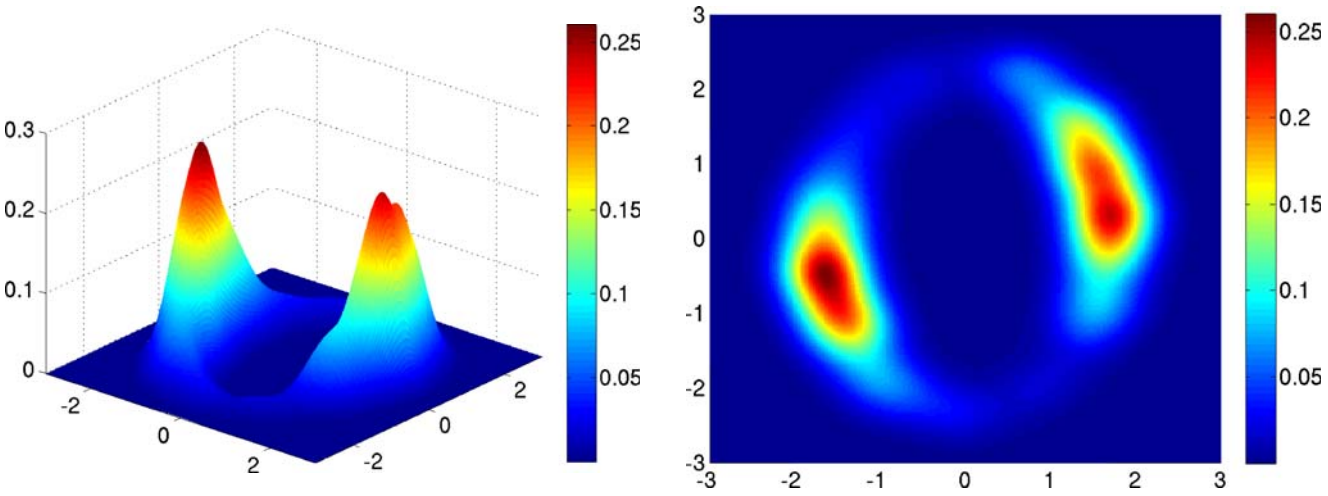
$$X \sim \mathcal{N}\left(0, \begin{pmatrix} 1 & 0.2 \\ 0.2 & 1 \end{pmatrix}\right) \tag{10}$$

Given a particular realization of X, written $X^* = (1.514,\ 1.335)$, we assume that we observe $D^* = F(X^*)$ with an error $\varepsilon \sim \mathcal{N}(0, 0.5)$. We plot the posterior density in Fig. 3a and b from two different viewpoints. We can see its support envelopes the ellipse $\{X \in \mathbb{R}^2 | F(X) = D\}$, and it presents two distinct modes.



a. perspective view. *b. view from the top.*

Fig. 4 KDE of the posterior constructed on an i.i.d. sample of size 10,000 (**a**, **b**)



a. perspective view. *b. view from the top.*

Fig. 5 KDE of the posterior constructed on the states generated by the IMH algorithm (**a**, **b**)

To solve this problem, we try four different MH samplers:

1. The independent sampler (IMH): $q(x, y) = q(y)$, where q is given by Eq. 10.
2. The symmetric increments random-walk sampler (SIMH): $q(x, y) = q(|y - x|)$, where q is the multivariate distribution $\mathcal{N}(0, hI)$, where I is the identity matrix, and $h = 0.1$.
3. The Langevin sampler (LMH): assume that π is differentiable on $\mathcal{X}$, q takes the form:

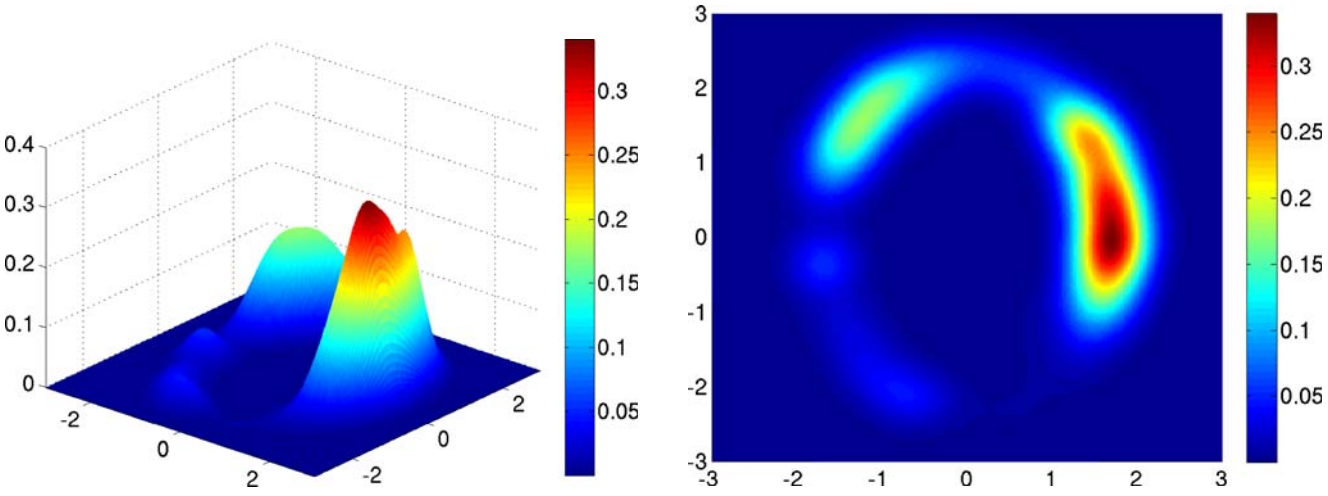
$$q(x, y) \sim \mathcal{N}\left(x + \frac{h^2}{2}\nabla \log(\pi(x)), h^2 I_d\right), \tag{11}$$

where h is chosen 0.1.

4. The adaptive algorithm of [11] (ASIMH): In this algorithm, y is proposed according to $q_{\theta_n}(x, \cdot) = \mathcal{N}(x, \Gamma_n)$, where $\theta = (\mu, \Gamma)$. We also consider a nondecreasing sequence of positive stepsizes $\{\gamma_n\}$, such that $\sum_{n=1}^{\infty} \gamma_n = \infty$ and $\sum_{n=1}^{\infty} \gamma_n^{1+\delta} < \infty$ for some $\delta > 0$. In practice, we use: $\gamma_n = 1/n$, as suggested in [11]. The parameter estimation algorithm takes the following form:

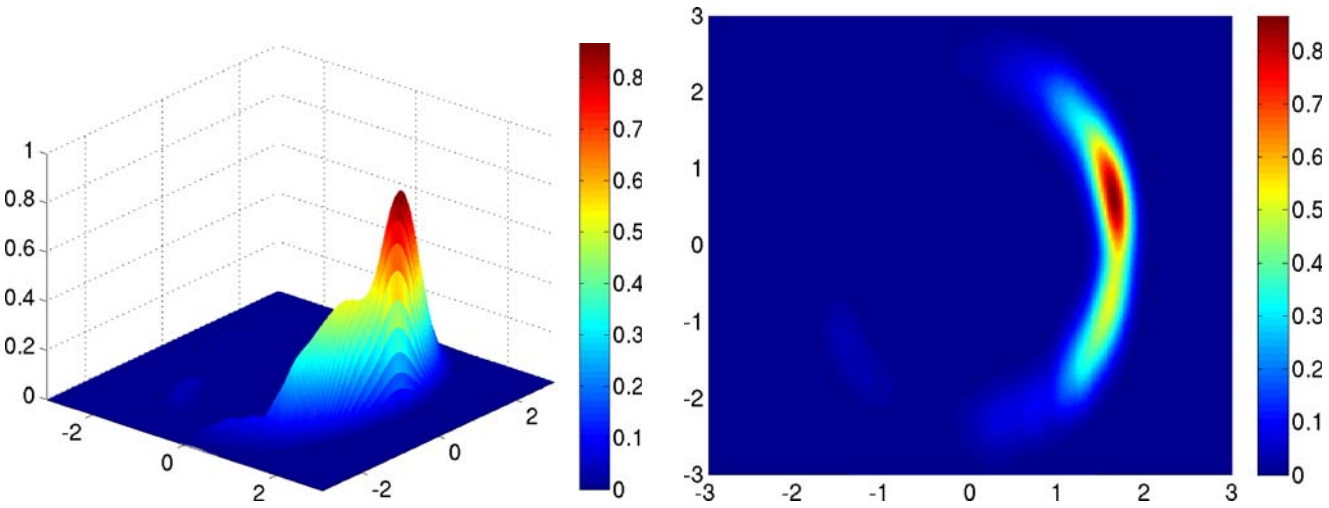
$$\mu_{n+1} = \mu_n + \gamma_{n+1}\left(X_{n+1} - \mu_n\right), n \geq 0,$$
$$\Gamma_{n+1} = \Gamma_n + \gamma_{n+1}\left(\left(X_{n+1} - \mu_n\right)\left(X_{n+1} - \mu_n\right)^T - \Gamma_n\right), \tag{12}$$

We can notice here that the chain generated by the adaptive algorithm is no longer homogeneous, but it



a. perspective view. *b. view from the top.*

Fig. 6 KDE of the posterior constructed on the states generated by the SIMH algorithm (**a**, **b**)



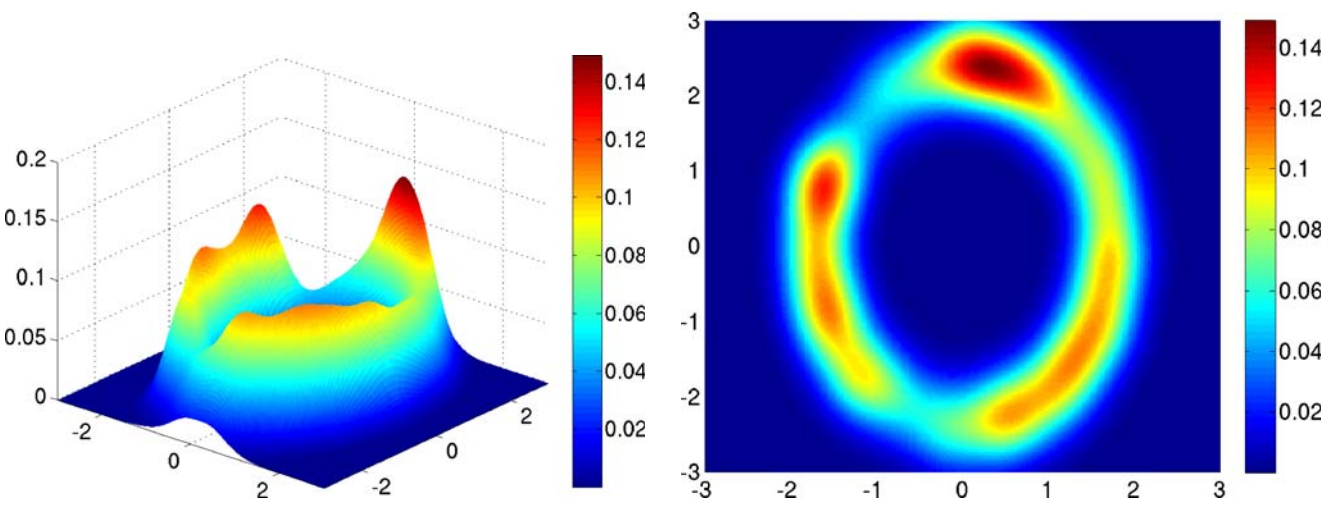
a. perspective view. *b. view from the top.*

Fig. 7 KDE of the posterior constructed on the states generated by the LMH algorithm (**a**, **b**)

can be proved (see [11] and [3] and [2] in a more general framework) that it has the correct ergodic properties. The idea of adaptive sampling is to improve the proposal efficiency, making it as close as possible to the posterior density. Practically, the parameters of the proposal are not updated at each iteration. We ran this algorithm with a first update of the parameters after 50 iterations and then an update every 10 iterations.

We ran all four algorithms for 10,000 iterations, which already represent an important number of iterations for history matching applications. We compare the results given by those four algorithms for the inverse problem defined above. To do this, we introduce three different criteria:

1. The kernel density estimate (KDE) of the posterior in Figs. 5, 6, 7, and 8, to be compared to Fig. 4. The latter represents the KDE of the posterior constructed from an i.i.d. sample of size 10,000 of the true posterior. In the whole paper, we consider the bandwidth of the KDE to be chosen according to the sample size.
2. The empirical acceptance rate in Fig. 9.



a. perspective view. *b. view from the top.*

Fig. 8 KDE of the posterior constructed on the states generated by the ASIMH algorithm (**a**, **b**)

3. The mean autocorrelations along the chain in Fig. 10.

In Figs. 5, 6, 7, and 8, we can see the KDE computed from the states generated by all four chains. It is of great interest to remark that the SIMH and the LMH only perform a local sampling of the posterior (Figs. 6 and 7), missing one of the two modes of the posterior. Indeed, their KDE do not exhibit the two modes of the posterior. Only the spannings performed by the IMH and the ASIMH are correct as they charge the entire ellipse. Particularly, the KDE constructed from the IMH sample is correct (Fig. 5). The one corresponding to the ASIMH sample, in Fig. 8, is really poor. This is probably due to the fact that we considered the entire sample. Indeed, it is possible that, as the first updates of the parameters occur, the algorithm has only spanned a small portion of the space and then it makes artefacts appear due to the use of *bad* parameters. Indeed, the parameter estimates are really slow to converge, and it conducts to poor sampling in the beginning of the algorithm. This behavior makes this algorithm improper for our application domain.

Furthermore, the ASIMH and the IMH exhibit the poorest final empirical acceptance rates (Fig. 9), around 0.1, meaning that only 10% of the moves are accepted for this relatively simple problem. Conversely, the SIMH and IMH show good performances regarding this criterion, especially the LMH, with a final empirical acceptance rate greater than 0.95. However, those two chains show poor performances with respect to autocorrelation length (defined as the first time beyond which the autocorrelation is less than 0.05, the dot line

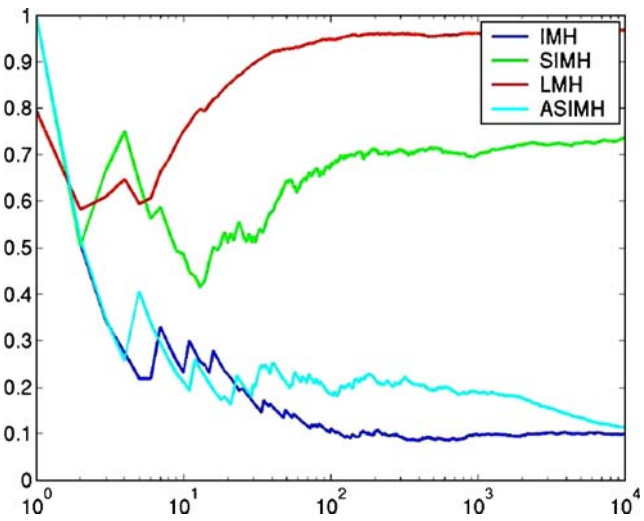


Fig. 9 Empirical acceptance rates along the chain for all four algorithms, logarithmic scale

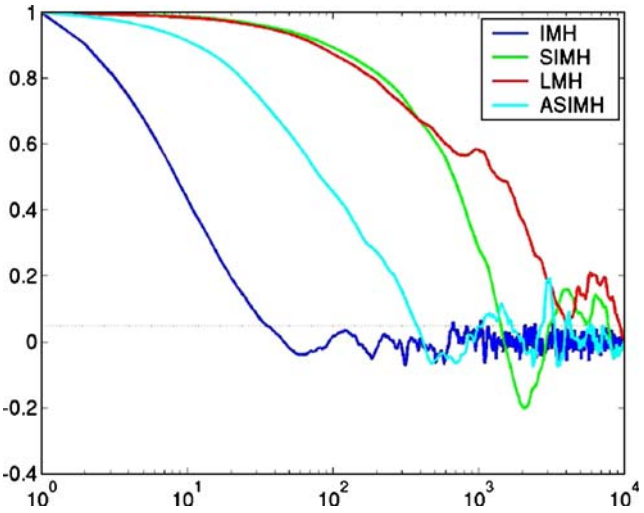


Fig. 10 Autocorrelations along the chain for all four algorithms, logarithmic scale

in Fig. 10), as they exhibit the longest autocorrelation length, unlike IMH and ASIMH.

By looking at these figures, the practitioner has to choose between good mixing properties with short autocorrelation but really poor acceptance rates and bad mixing with long autocorrelation and good acceptance rates. Furthermore, the effect of an increasing dimension is not addressed here, but it is known to worsen the properties of the chain. Especially, the IMH algorithm is known to fail in high dimension, although it performed relatively well here. In order to improve the mixing properties of the chains without increasing the computational cost of the algorithm, we now introduce the principle of parallel interacting Markov Chains.

4.3 Parallel Interacting Markov Chains

The principle of making interacting Markov Chains first appeared in [9] under the name parallel tempering (PT). It has been mostly applied in physicochemical simulations; see [7] and references therein. It is known in the literature under different names, such as exchange Monte Carlo and metropolis coupled-chain; see [14] for a review. The principle of PT is to simulate a number $(K+1)$ of replicas of the system of interest by MCMC, each at a different temperature, in the sense of the simulated annealing, and to allow the chains to exchange information, swapping their current state. The high-temperature systems are generally able to sample large volumes of state space, whereas low-temperature systems, while having precise sampling in a local region of state space, may become trapped in local energy minima during the timescale of a typical computer

simulation. PT achieves good sampling by allowing the systems at different temperatures to exchange their state. Thus, the inclusion of higher-temperature systems ensures that the lower-temperature systems can access a representative set of low-temperature regions of state space.

Simulation of $(K+1)$ replicas, rather than one, requires on the order of $(K+1)$ times more computational effort. This *extra expense* of PT is one of the reasons for the initially slow adoption of the method. Eventually, it became clear that a PT simulation is more than $(K+1)$ times more efficient than a standard, single-temperature Monte Carlo simulation. This increased efficiency derives from allowing the lower-temperature systems to sample regions of state space that they would not have been able to access, even if regular sampling had been conducted for a single-temperature simulation that was $(K+1)$ times as long. It is also worth noticing that PT can make efficient use of large CPU clusters, where different replicas can be run in parallel, unlike classical MCMC sampling that are sequential methods. An additional benefit of the PT method is the generation of results for a range of temperatures, which may also be of interest to the investigator. It is now widely appreciated that PT is a useful and powerful computational method.

More recently, some researchers in the statistical community gave attention to PT and, more generally, to interacting Markov Chains. They propose a general theoretical framework and new algorithms in order to improve the exchange information step addressed above. Two main algorithms drawn our attention: the equi-energy sampler (EES) of [16] and the population importance-resampling MCMC sampler (PIR) of [1], which allows us to go back in the history of the chain. More precisely, these last two algorithms are based on self-interacting approximations of nonlinear Markov kernels, defined by Andrieu et al. [1]. We now describe these methods in our context.

4.3.1 Description of the algorithms

We first define the family $\{\pi^{(l)}, l = 0..K\}$ of distributions we want to simulate from, such that:

$$\pi^{(l)}(x) \propto e^{-E_l(x)}, \tag{13}$$

where $E_l(x) = \frac{E(x)}{T_l}$ and $E(x) = \frac{1}{2}\|D^* - F(x)\|^2_{C_D^{-1}} + \frac{1}{2}\|x-\mu\|^2_{\Gamma^{-1}}$ is the energy of the system and T_l is the temperature at which it is considered. The T_l satisfy: $T_0 = 1 < T_1 < \ldots < T_K < +\infty$, so that $\pi^{(0)} = P(X|D^*)$. These distributions are a family of *tempered* versions of $P(X|D^*)$. We will also talk of tempered energies to denote the E_l. The parallel algorithms aim to simulate from:

$$\Pi(x) = \prod_{l=0}^{K} \pi^{(l)}(x), \tag{14}$$

allowing exchanges between states at different temperatures. Dilatate versions of $\pi^{(0)}$: $\pi^{(1)}, \ldots, \pi^{(K)}$ are easier to simulate. Thus, they can provide information on $\pi^{(0)}$.

Different strategies can be adopted to exchange information between chains at adjacent temperatures. We denote by $x = (x^{(0)}, \ldots, x^{(K)}) \in \mathcal{X}^{K+1}$ the current state of the chain. For $l = 0, \ldots, K-1$, we define the importance function:

$$r^{(l)}(x) = e^{-(E_l(x) - E_{l+1}(x))} \tag{15}$$

The method can be formalized by defining the following kernel P_n at time n, given all the previous states $x_{0:n-1} = (x_0, \ldots, x_{n-1})$ and for $A_0 \times \ldots \times A_K \in \mathcal{B}(\mathcal{X}^{K+1})$:

$$\begin{aligned} &P_n\ (x_{0:n-1}; A_0 \times \ldots \times A_K) \\ &\quad = P^{(K)}(x^{(K)}, A_K) \prod_{l=0}^{K-1} P^{(l)}_{x^{(l+1)}_{0:n-1}}(x^{(l)}, A_l), \end{aligned} \tag{16}$$

where, for $x^{(l+1)}_{0:n-1} \in \mathcal{X}^n$, $x^{(l)} \in \mathcal{X}$ and $A \in \mathcal{B}(\mathcal{X})$:

$$\begin{aligned} P^{(l)}_{x^{(l+1)}_{0:n-1}}(x^{(l)}; A) &= \theta\, P^{(l)}(x^{(l)}, A) \\ &\quad + (1-\theta) \int_{\mathcal{X}} \nu^{(l)}_{x^{(l+1)}_{0:n-1}}(x^{(l)}, dy) \\ &\quad \times T^{(l)}(y, x^{(l)}; A), \end{aligned} \tag{17}$$

and

$$\begin{aligned} &\nu^{(l)}_{x^{(l+1)}_{0:n-1}}(x^{(l)}, dy) \\ &\quad = \frac{\sum_{i=0}^{n-1} \omega^{(l)}_{n,i}(x^{(l)}, x^{(l+1)}_i)\delta_{x^{(l+1)}_i}(dy)}{\sum_{i=0}^{n-1} \omega^{(l)}_{n,i}(x^{(l)}, x^{(l+1)}_i)} \end{aligned} \tag{18}$$

and in the algorithms considered here, $T^{(l)}$ will take the following form:

$$\begin{aligned} T^{(l)}(y, x^{(l)}; A) &= \min\left(1, \frac{r^{(l)}(y)}{r^{(l)}(x^{(l)})}\right) \mathbb{1}_A(y) \\ &\quad + \left(1 - \min\left(1, \frac{r^{(l)}(y)}{r^{(l)}(x^{(l)})}\right)\right) \\ &\quad \times \mathbb{1}_A(x^{(l)}) \end{aligned} \tag{19}$$

In other words, at time step n, at temperature T_l, with probability θ, a classical MH move will be performed. Otherwise, with probability (1-θ), an exchange move will be proposed. It consists in choosing a state y among

$x_{0:n-1}^{(l+1)}$ with the weights $\omega_{n,i}^{(l)}(x^{(l)}, x_i^{(l+1)})$. This move is then accepted with probability $\min\left(1, \frac{r^{(l)}(y)}{r^{(l)}(x^{(l)})}\right)$. The new state is chosen according to $\nu^{(l)}_{x_{0:n-1}^{(l+1)}}$, that can be viewed as an importance sampling estimate of $\pi^{(l)}$ with the instrumental law $\pi^{(l+1)}$.

The three algorithms (PT, EES, and PIR) considered in this article can be written in this framework, and they differ only from the formulation of the weights $\omega_{n,i}^{(l)}$, for some $(y, z) \in \mathcal{X}^2$:

- For the PT algorithm, we have:

  $$\omega_{n,i}^{(l)}(y, z) = \mathbb{1}_{i=n-1},$$

  it is only possible to go to the current state of the chain at the adjacent higher temperature.
- For the EES algorithm, given a sequence of energy levels $E_0 < E_1 < \ldots < E_K < E_{K+1} = \infty$ defining a partition: $\mathcal{X} = \bigcup_{l=0}^{K} \mathcal{X}_l$ of energy rings: $\mathcal{X}_l = \{x \in \mathcal{X} : E_l < E(x) < E_{l+1}\}$, and the function $I(x) = l$ if $x \in \mathcal{X}_l$ the $\omega_{n,i}$ take the form:

  $$\omega_{n,i}^{(l)}(y, z) = \mathbb{1}_{\mathcal{X}_{I(y)}}(z),$$

  that is, the new state will be taken uniformly among the states $x_{0:n-1}^{(l+1)}$ of the chain at temperature T_{l+1} that are in the same energy ring as the current state.
- For the PIR algorithm, the weights $\omega_{n,i}$ take the following form:

  $$\omega_{n,i}^{(l)}(y, z) = r^{(l)}(z),$$

  that is, we obtain the new state by resampling from $x_{0:n-1}^{(l+1)}$ with the weights ω.

The main idea behind the two last algorithms is that the kernel defined in Eq. 17 will converge towards the following limiting kernel:

$$P^{(l)}_{x_{0:n-1}^{(l+1)}}(x^{(l)}; A) = \theta P^{(l)}(x^{(l)}, A) + (1-\theta) R^{(l)}(x^{(l)}, A), \tag{20}$$

where $R^{(l)}$ is a MH kernel, whose proposal distribution is given by:

- $Q^{(l)}_{PIR}(x, dy) = \pi^{(l)}(dy)$ for the PIR algorithm
- $Q^{(l)}_{EES}(x, dy) \propto \pi^{(l+1)}(y)\mathbb{1}_{\mathcal{X}_{I(x)}}(y)\lambda(dy)$ for the EES

Obviously, the convergence towards $R^{(l)}$ will not be achieved in the time of the simulation, but its approximation at time n will help to sample from the posterior, particularly to span a larger part of the state space.

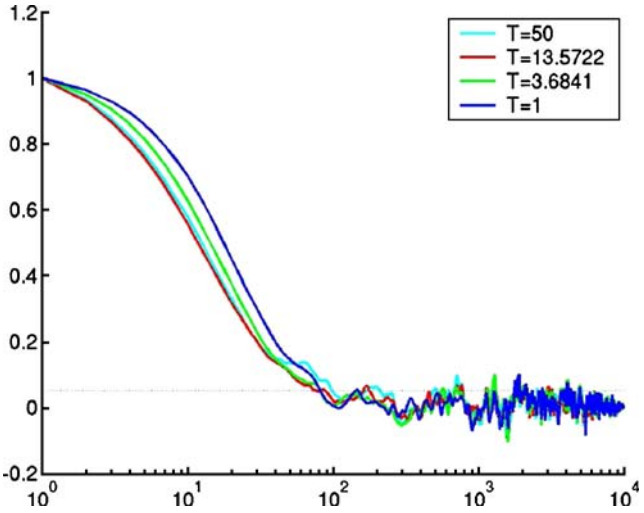


Fig. 11 Autocorrelations along the chain for the PT algorithm, logarithmic scale

Finally, it is worth noticing that, for all three algorithms, we can use the samples of all the chains, reweighting them by the following importance weights:

$$\eta^{(l)}(x) = e^{-(E_0(x) - E_l(x))}, \tag{21}$$

in order to compute estimates of $I_g = E_{\pi_0}\left[g(X)\right]$, for some g. Hence, the estimate $\hat{I}_g$, after N iterations of the algorithm, will take the form:

$$\hat{I}_g = \sum_{l=0}^{K} \frac{\sum_{i=0}^{N} \eta^{(l)}(x_i^{(l)}) g(x_i^{(l)})}{\sum_{i=0}^{N} \eta^{(l)}(x_i^{(l)})}. \tag{22}$$

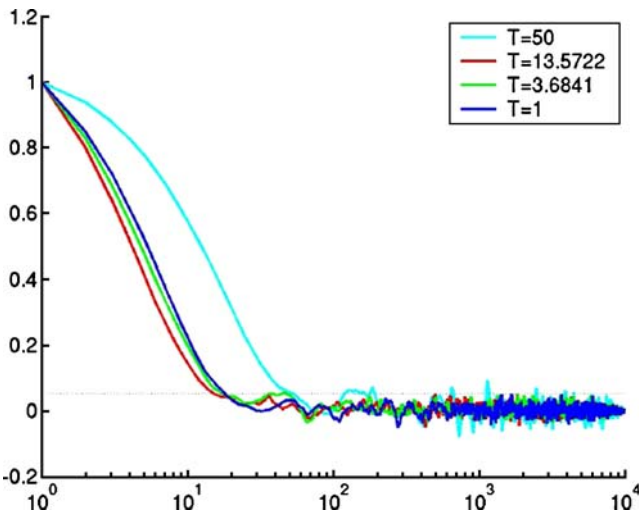


Fig. 12 Autocorrelations along the chain for the EES, logarithmic scale

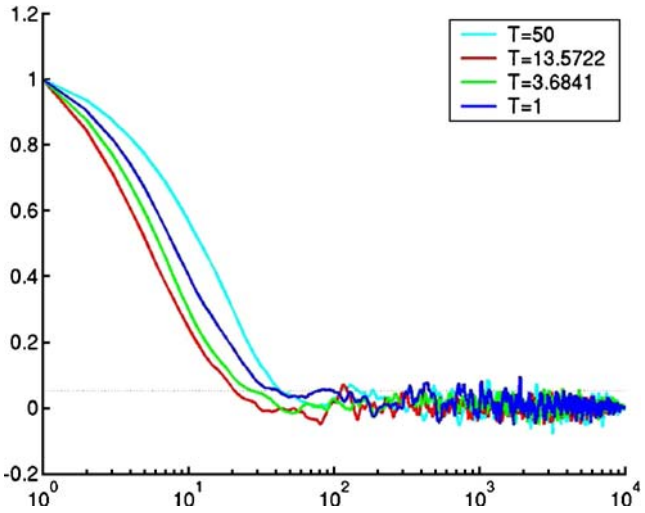


Fig. 13 Autocorrelations along the chain for the PIR algorithm, logarithmic scale

It has been shown numerically in [16] that using the reweighted entire sample will provide better estimates than using only $x_{0:N}^{(0)}$.

Concerning the choice of the parameters, some heuristic rules exist and are discussed in, e.g., [14] for the PT algorithm and in [16] for the EES. However, the choice depends mainly on the problem addressed and there is no general recipe to tune the parameters. We will explain how we choose them for the history matching application in Section 5.2. We now describe some numerical results for these three algorithms.

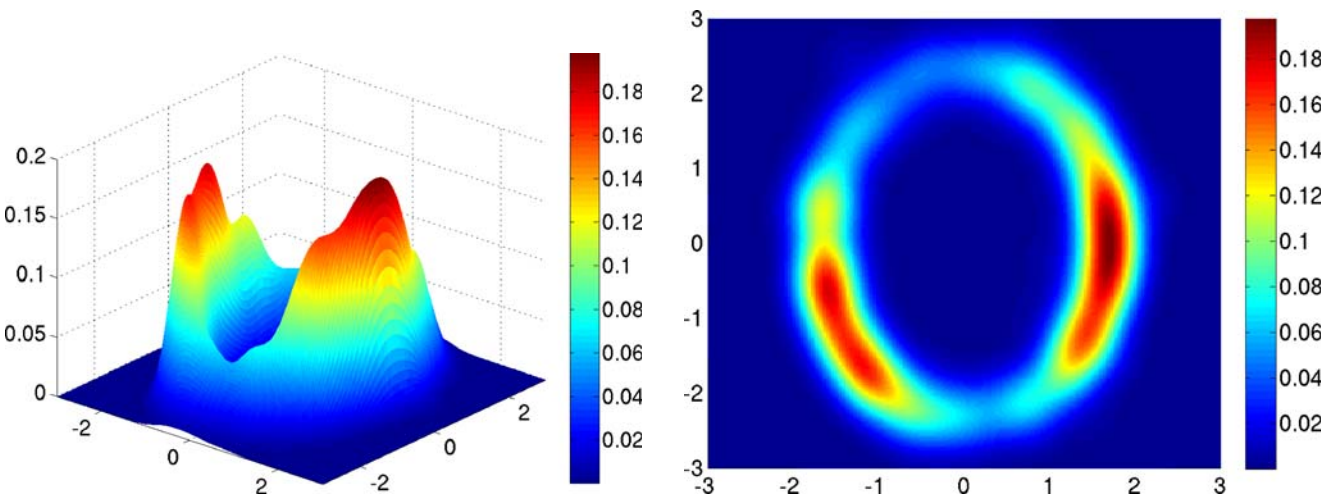
4.3.2 Numerical results

We compare the results given by these three algorithms on the problem defined in Section 4.2. We consider four different temperatures ($T_3 = 50$, $T_2 = 13.5722$, $T_1 = 3.6841$, $T_0 = 1$) and we set the probability of exchange θ at 0.2. At each temperature, we use a SIMH sampler with a variance varying according to the temperature, namely 0.1 at T_0 and $0.1\sqrt{T_l}$ at T_l, $0 < l \leq 3$.

We now compare the results given by those three algorithms for the inverse problem defined in Section 4.2. We do not reproduce here each of three different criteria we used previously. Indeed, in terms of empirical acceptance rate, for each chain, our three algorithms behave like the SIMH. We recall the two criteria, defined in Section 4.2, that we use there:

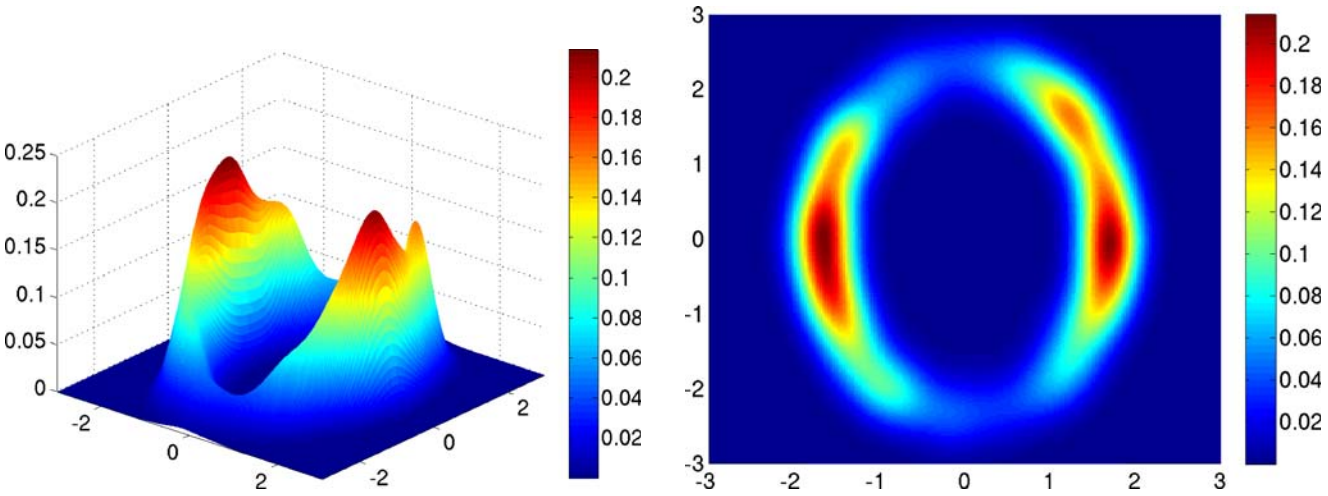
1. The mean autocorrelations for each parallel chain along the chain in Figs. 11, 12, and 13.
2. The KDE of the posterior in Figs. 14, 15, and 16 to be compared to Fig. 4 . Again, we used a bandwidth chosen according to the sample size. We constructed these estimates considering the whole sample, that is, including the states of all four chains, weighting them by the weights (Eq. 21).

When we look at Figs. 11, 12, and 13, we can see that these methods provide the major advantage of reducing autocorrelations along the chain with respect to single MCMC. For each of three algorithms, we represent the autocorrelations along each of four chains. We can see that the autocorrelations along the chain at T_0 are



a. perspective view. *b. view from the top.*

Fig. 14 KDE of the posterior constructed on the states generated by the PT algorithm (**a**, **b**)



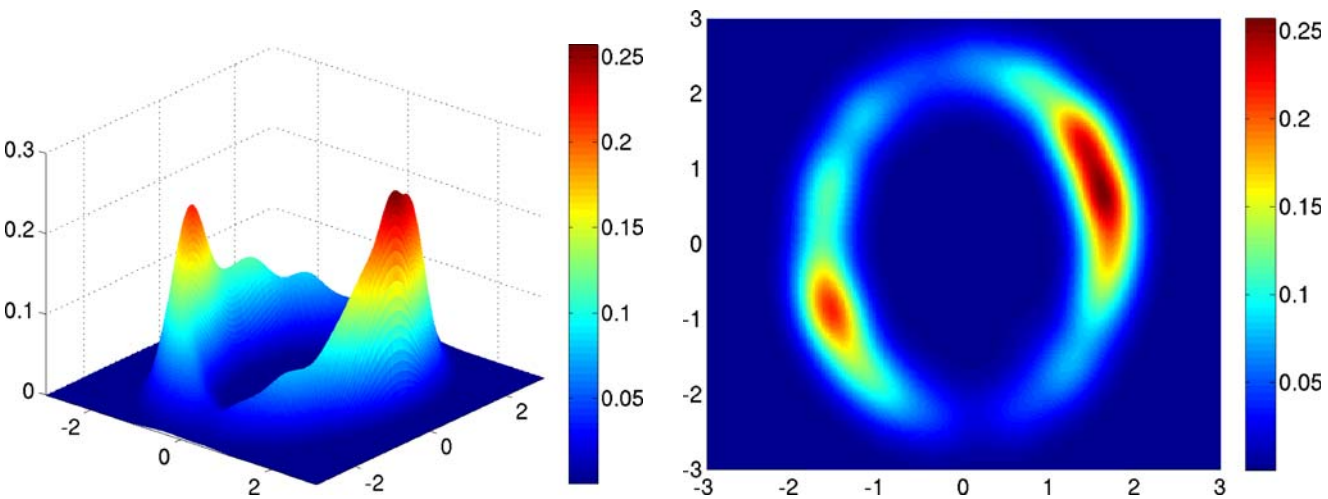
a. perspective view. *b. view from the top.*

Fig. 15 KDE of the posterior constructed on the states generated by the EES (**a**, **b**)

systematically reduced with respect to the classical SIMH algorithm (see Fig. 10), although the proposal variance has been chosen equally to be 0.1. This phenomenon is due to the exchange steps that allow regenerations of the chains. In other words, each piece of chain between two exchange steps is independent from one another, which induces the reduction of the autocorrelations.

In Figs. 14, 15, and 16, we represent the resulting KDE for all three algorithms. We can see the ability of these methods to efficiently sample multimodal distribution: the density estimates clearly exhibit the two modes of the posterior distribution.

Finally, we have shown in this toy example the advantage of using interacting Markov Chains: they improve the browsing of the support of the distribution while reducing the autocorrelations, comparatively to classical single MCMC sampling. These performances are realized without increasing the computational cost, as these algorithms are easily parallelizable. In fact, at a fixed iteration number, the parallel algorithms are even faster, as an exchange step counts for one iteration, without computing the forward operator. These methods are then particularly suited for solving inverse problems where no analytical expression for the forward operator exists, like the history matching problem.



a. perspective view. *b. view from the top.*

Fig. 16 KDE of the posterior constructed on the states generated by the PIR algorithm (**a**, **b**)

More precisely, the PIR and the EES algorithms are more powerful than the PT, as they allow exchanges with the past of the chains, accounting for all the information learned until the iteration n. In the next section, we apply the PIR algorithm to a reservoir characterization synthetic problem.

5 Application to history matching

In this section, we describe the results given by the PIR algorithm on a synthetic reservoir characterization problem. Indeed, we think it is the most suitable algorithm for our problem. Considering that the lower the temperature is, the slower will the chain enter the stationary regime; we can remark that the PIR does not need the chains to be in stationary regime before allowing exchanges, contrary to the EES algorithm. Indeed, in the EES, the exchange proposal is made in the same energy ring as the current state. Then, if the chain of interest has not reached the stationary regime and is still at high energy levels, the exchange proposal will be in the same energy ring as the current state. Therefore, it will not help to attain stationary regime. Conversely, the PIR proposes exchange proposals according to an importance sampling step, constructed on the states generated at the higher adjacent temperature. The proposals are then more likely at low energy levels. Therefore, we choose to apply here the PIR algorithm, described in Section 4.3.1.

5.1 Description of the case

The prior geostatistical model of the field for this problem and the parameters of the fluid-flow simulator are the same as in 3.4. Given a reference realization of the field X, computed with all its 2,498 components, and its water cut calculated on 3,000 days, we attempt to condition the geostatistical model X to the water cut D^*. In other words, we try to produce an i.i.d sample of Eq. 1 as explained in Section 2. However, we use a truncated KL expansion (Eq. 2) with $M = 50$ components to represent the field. Hence, the dimension of the inference problem is reduced. The formulation of the posterior is then the following:

$$P(X^{(M)}|D^*) \propto \mathrm{e}^{\left(-\frac{1}{2}\|D^*-F(X^{(M)})\|^2-\frac{1}{2}\|X^{(M)}-\mu\|^2_{\Gamma^{-1}_{(M)}}\right)}, \qquad (23)$$

where $\Gamma_{(M)} = \Phi_{(M)}\Lambda\Phi^t_{(M)}$,

$\Phi_{(M)}$ is the matrix $L \times M$ whose column vectors are the $\phi_i(x)$,

Λ is the diagonal $M \times M$ whose diagonal components are the λ_i.

Here, we consider the production data water cut. We recall here that the water cut is the proportion of water in the oil produced, in function of time. Hence, D^* and $F(X^{(M)})$ are both functions of time. The covariance of the measurement error on the water cut is assumed to be I, where I is the identity matrix. The exponent of the likelihood term in Eq. 23 is written:

$$\|D^* - F(X^{(M)})\|^2 = \sum_{t=0}^{t=3000} \left(D^*_t - F_t(X^{(M)})\right)^2$$

5.2 Methodology

In order to sample from Eq. 23, we implement the PIR algorithm. In this section, inspired by practical considerations found in [14] and [16] and our own experience, we explain how we choose the different parameters of the algorithm. Particularly, we will focus on the following four points:

1. The highest temperature to choose
2. The kernel to choose as a function of the temperature
3. The number of chains
4. The probability of proposing information exchange between chains

We first launch 200 preliminary runs in order to choose the maximum temperature we will use. We represent the histogram of energies in Fig. 17. In Fig. 18, we also show the corresponding curves of water cut together

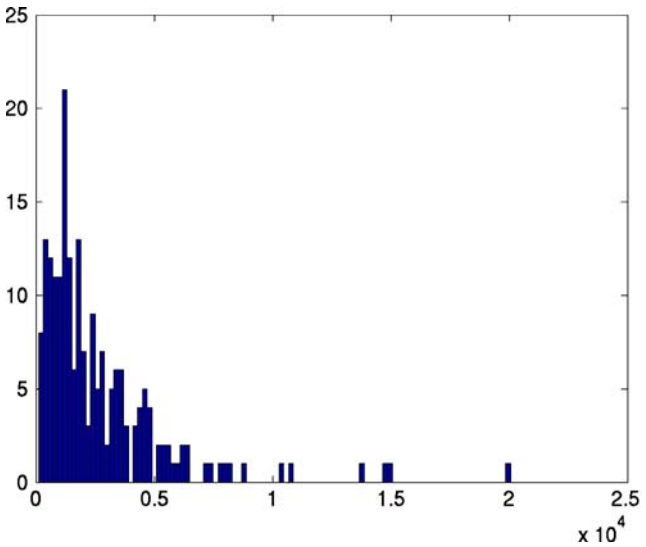


Fig. 17 Histogram of energies of 200 random realizations

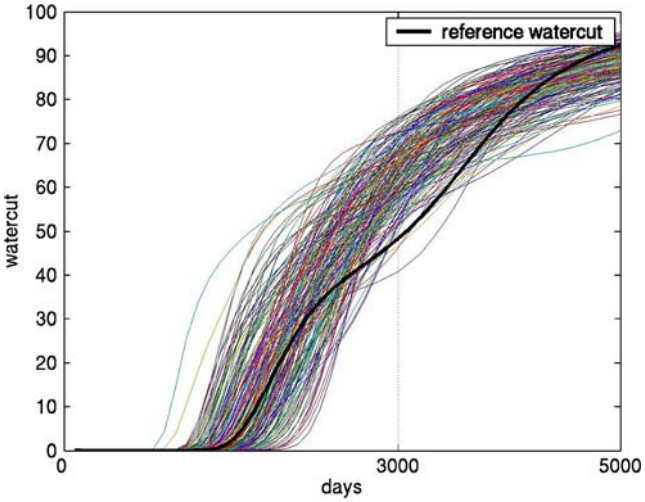


Fig. 18 Water cut curves of the sample

with the black thick curve representing the reference water cut. Moreover, we represent in this figure the 0-to-3,000-day period to be matched as well as the 3,000-to-5,000-day period. This last part of the curve is computed to assess the prediction ability of the sample. We can see in this figure that all the curves are far away from the reference water cut and are really poor for prediction purposes.

We can see in Fig. 17 that the energy is mainly distributed between 0 and 5,000, with an average of about 3,000 (the minimum energy of the sample being about 800). As proposed in [14], we choose first a maximal temperature. We consider the energy term in Eq. 23:

$$E(X) = \frac{1}{2}\|D^* - F(X^{(M)})\|^2 + \frac{1}{2}\|X^{(M)} - \mu\|^2_{\Gamma^{-1}_{(M)}}. \quad (24)$$

The mean of the second term of Eq. 24 is:

$$\mathbb{E}\left(\frac{1}{2}\|X^{(M)} - \mu\|^2_{\Gamma^{-1}_{(M)}}\right) = \mathbb{E}\left(\frac{1}{2}\sum_{i=1}^{M}\xi_i^2\right) = \frac{M}{2},$$

when the ξ_i, defined in Section 3.1, are considered i.i.d centered reduced Gaussian. Formally, the posterior distribution of the ξ_i is not Gaussian, as F is not linear. Nevertheless, we only try here to evaluate an order of magnitude of the mean energy of the posterior, in order to construct the temperature ladder. At the same time, the first term of Eq. 24 has to be near zero for a realization of Eq. 23. We deduce that the mean of Eq. 24 for the conditioned realizations will be about $M/2$, that is 25 here. As the average energy of the 200 Monte Carlo samples has been found to be 5,000, we choose the highest temperature to be 100 to ensure a sufficient transition acceptation rate for the chain at the highest temperature.

As already claimed, the idea of these methods is to improve the mixing of the chain. Then, according to the temperature, we have to choose kernels that will make this assumption effective. At the highest temperature, larger moves tend to be accepted, even though the energy level reached is not as low as the one finally aimed at. Thus, it is of great interest to use a fast mixing kernel that cannot be used at lower energy levels. For this application, a first run using a SIMH with very large moves at highest temperature, as proposed in [16], led to very poor results, generating only extreme values with very high energy levels at the highest temperatures. Thus, the exchange propositions were systematically rejected. We then chose an independent sampler at this temperature.

Conversely, at low temperatures, it is of interest to employ a kernel with good local properties like the Langevin sampler or a SIMH with small steps. This kind of kernel will perform a good local exploration of the posterior. The point is then to choose the intermediate kernels, between the highest and the lowest temperature levels. Increasing the step of a SIMH sampler according to the temperature, as proposed in [16], seemed to be a good idea. However, as said previously, it failed. The difficulty is then to choose kernels that progressively worsen their mixing properties, while increasing local properties, when descending the temperature ladder. In a high dimensional setting like ours, one can change the number of components that are affected at each iteration. For example, we can choose a kernel that modifies all components at each iteration at the highest temperature and a kernel that modifies only one at the lowest.

Furthermore, these considerations about kernels are closely related to the number of chains to use. Particularly, it is important not to employ too many chains. Indeed, using too many chains will slow the input of information from the highest temperature levels to the lowest. Conversely, the number of chains has to be large enough to allow them to exchange information at a good rate: the histograms of the tempered energies of two chains at adjacent temperatures need to possess a sufficient overlapping interval to allow exchanges.

Here, after some numerical experiments, we chose to use five different temperatures. We construct the temperature ladder distributing geometrically the temperatures between $T_0 = 1$ and $T_4 = 100$. The choice of a geometric distribution of the temperatures is a classical one in the PT literature; see, e.g., [7]. If the number of chains is sufficient, it generally allows a good overlapping of the histograms of the tempered energies.

Namely, we take $T_l = T_0 \left(\frac{T_4}{T_0}\right)^{l/4}$ for $l = 1, 2, 3$. Hence, we use the following temperature ladder:

$$T_0 = 1.000 < T_1 = 3.162 < T_2 = 10.000 < T_3 = 31.623 < T_4 = 100.000$$

Thus, we simulate five Markov chains $(X^{(l)})$ at the temperature $T^{(l)}$. At T_0, T_1, and T_2, we simulate from a SIMH algorithm with a step variance $0.1\sqrt{T_l}$, affecting, respectively, one, two, and five components. At T_3, we simulate from an independent sampler affecting 10 components. At T_4, we simulate from a global independent sampler. In other words, proportional to the temperature, we propose larger moves, using global samplers at the two highest temperatures. Simulating modifying fewer components at low temperature allows us to have better acceptance rates in our high dimensional space ($M = 50$) and to allow local exploration of the posterior. Moreover, the moves at the highest temperatures affect more components to improve the mixing of these chains, feeding the chains $(X^{(0)})$, $(X^{(1)})$, $(X^{(2)})$ with states that they could not have attained without the exchange steps.

Regarding the proposal rate of information exchange, there is again a balance to achieve between high and low rates. A high rate will encourage information exchange, but will slow local exploration. Conversely, a low rate will hamper the process of exchanging information.

After a few experiments, we allowed the chains to exchange information according to the PIR scheme just after the first iteration with a probability of 0.1. We ran this algorithm for 10,000 iterations, and we now describe its numerical results. The algorithm took 2 days to run on a desktop computer with a single processor AMD Opteron 146 2.0 GHz.

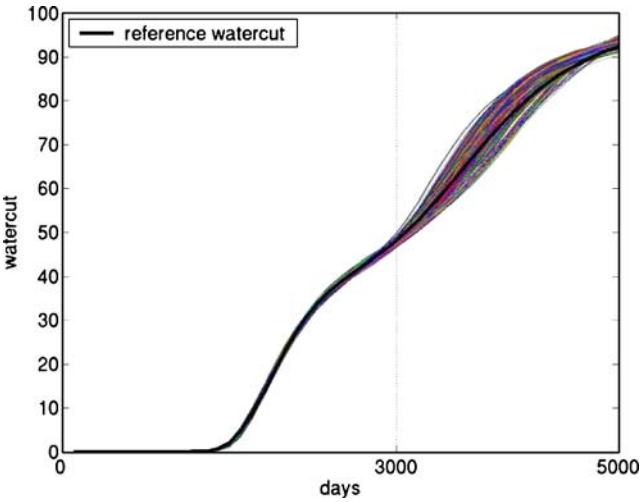


Fig. 20 Water cut curves of the sample and reference

5.3 Numerical results

We first show in Fig. 19 the energy of the states generated by the five chains used in the PIR for 10,000 iterations.

Figure 19 shows the energy of the states of the five chains in function of the number of the iteration. Looking at the lower curve, corresponding to T_0, we can see that it stabilizes after about 500 iterations, around

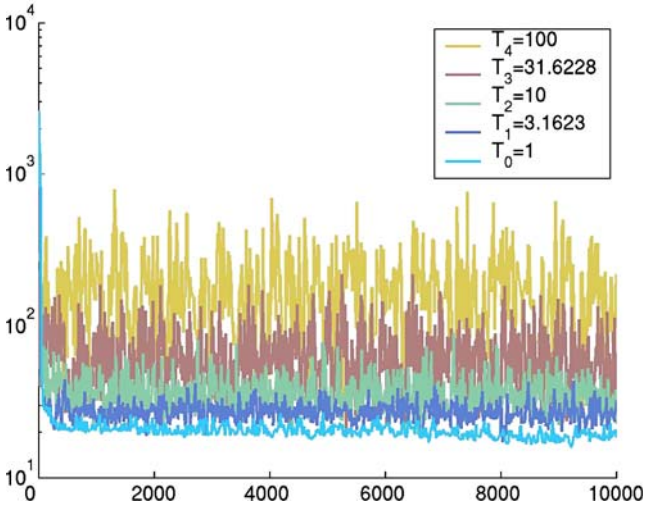


Fig. 19 Energy of the states of the five chains, generated by the PIR along the 10,000 iterations

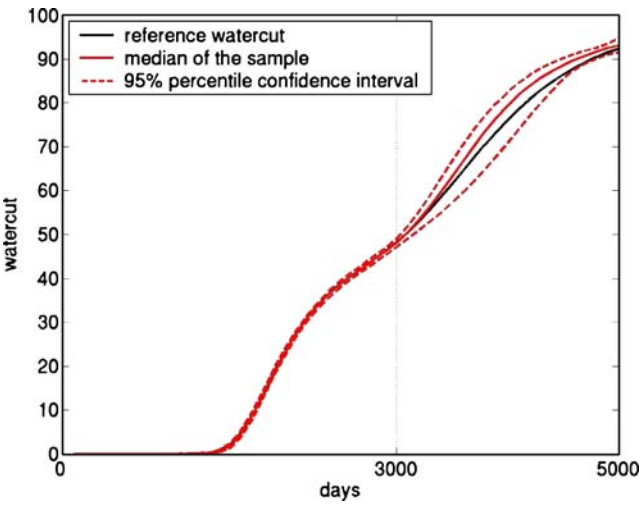


Fig. 21 Median, 95% percentile confidence interval, and reference water cut

levels of energy corresponding to the expected order of magnitude of the mean energy of the posterior. Indeed, allowing exchanges since the beginning helps to converge quickly. As all the other chains show a stabilized profile of energy after this number of iterations, we consider it as the end of the burn-in period, that is, we consider that each chain is in stationary regime beyond this number of iterations. Moreover, we can see that each couple of chains at adjacent temperatures show overlapping energy profiles, allowing the exchanges between the two chains. Indeed, the empirical exchange acceptance rate has been found between 0.6 and 0.8 for each couple of adjacent chains.

We present in Fig. 20 the water cut curves corresponding to the states generated by the chain at T_0 after the 500^{th} iteration. We plot the curves corresponding to this chain only because the weights (Eq. 21) of the states generated by the other chains are negligible in this application. We also represent the reference water cut as the thick line in black on this figure. In Fig. 21, the reference water cut in black, the median of the sample in red, and the 95% percentile interval as red dotted lines are represented. We use the median as estimator of the expectation, as it is more robust than the empirical mean, particularly to eventual extreme values.

We can see in Fig. 20 that the curves generated by our algorithm are well distributed around the reference one. In other words, the water cut data, up to 3,000 days, is correctly matched for the sample. Particularly, the breakthrough time, that is the time

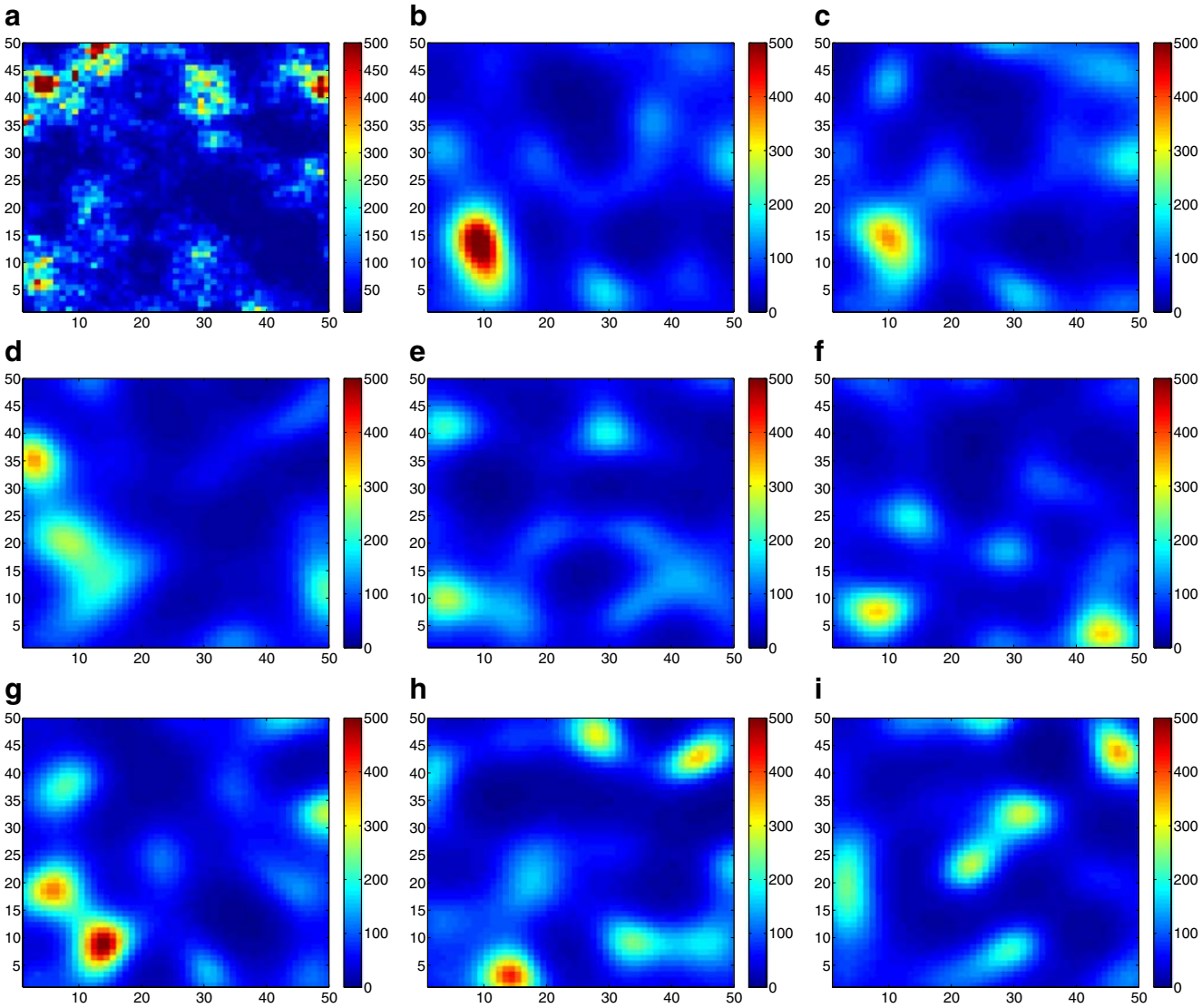


Fig. 22 Reference permeability field (**a**) and eight realizations from the posterior generated by the PIR algorithm (**b**–**i**)

when the water cut become greater than 0, is perfectly matched by the whole sample. Between 3,000 and 5,000, the curves are more widely spread around the reference, exploring the different possible prolongations of the curve. Comparing these results with Fig. 18 enlightens the quality of our results. In Fig. 21, we can also see that, for the period matched (up to 3,000 days), the median of the sample produced perfectly matched the reference. Moreover, the 95% confidence interval is extremely thin around the reference water cut until 3,000 days, when it widens according to the sample generated. We can see that, for the next 2,000 days, the reference water cut stays in the 95% confidence interval and is quite close to the median, validating our sample for prediction purposes. We can also remark that the confidence interval tends to shrink at the end of the period considered. This effect is due to the intrinsic nature of the water cut on this example. Indeed, it is a strictly increasing function with an asymptote at 100. This explains why the curves of the sample generated are getting closer at the end of the time period. Finally, we represent, in Fig. 22, the reference permeability field computed with all the components and a collection of eight realizations of the permeability field conditioned to the water cut data D^*, generated by the PIR algorithm.

We can first see in Fig. 22 that the aspect of the realizations (b, ..., i) generated by our algorithm is far smoother than the reference. This is due to the approximation made by using a truncated KL expansion with $M = 50$ components. It has been seen in Section 3.4 that such an approximation is able to reproduce the dynamical properties of the field. It is then possible to add the remaining components of the KL expansion to each realization, simulating the remaining (ξ_i), to obtain a permeability map that will have the same aspect as the reference one.

It can also be seen that the realizations we represent here are clearly different between each other. This illustrates the good exploration of the posterior (Eq. 23) carried out by the PIR algorithm, by improving the mixing properties with respect to classical single MCMC. A single Markov chain could not have produced such various maps within the same number of iterations; see, e.g., [6]. This ability is of great interest when addressing the problem of prediction uncertainties. Indeed, different permeability maps will lead to different prediction curves. Identifying numerous different permeability maps that are potential representations of the true permeability field will lead to better uncertainty assessment. That is illustrated here with the water cut curve prediction with its associated confidence interval in Fig. 21.

6 Conclusions

In this work, we have first described a way to reduce the dimension of the inverse problem in history matching by using a truncated KL expansion of the random field of interest. We have validated the method by a Monte Carlo study. Then, we have shown the deficiencies of classical MCMC methods when applied to multimodal ill-posed inverse problems. We have then proposed an innovative application of recent stochastic simulation methods, based on parallel interacting Markov Chains. Numerical results on a toy example have shown its interest in the resolution of inverse problems in the Bayesian framework. Finally, an application on a synthetic case of reservoir characterization has been performed, and its results are clearly satisfactory. Especially, the results have demonstrated the improved properties of the PIR algorithm in the sampling of the posterior. Moreover, we have produced with this algorithm a sample with good predictive properties. Finally, we want to recall here that our method has the same computational cost as classical MCMC methods.

Improvements can be made on the parameterization of the parallel algorithm. For instance, it could be really interesting to use a Langevin sampler at the lowest temperature, as this kernel is known to exhibit the best local performances among the different samplers. However, it requires computing the gradient of the fluid-flow simulator. This could be done by the use of a simulator with adjoint states or numerically. The dimension still being 50 in the case studied here, a numerical estimate of the gradient in each direction is intractable: it would require 50 additional simulations at each iteration to compute the local gradient estimate in single precision. Nevertheless, choosing the most important directions in which to compute the gradient should be of great interest. A preliminary sensitivity analysis could indeed be performed before running the algorithm in order to evaluate the influence of the components of the KL expansion for the considered response of the fluid-flow simulator. Then, a local numerical estimate of the gradient could be computed in the directions chosen. This task is also easily parallelizable. With this estimation, the Langevin sampler could be implemented without increasing the computational cost. Theoretically, it would make the local exploration of the posterior more accurate.

When new data are available, we have to integrate them. This problem of integrating new data could be easily addressed in the following way: we could use either the same method using the kernel given by the final estimation of Eq. 20 or an importance sampling resampling scheme [23] proposing a realization with

the weights given by Eq. 21, then reweighting them according to the new results.

It would also be of great interest to investigate the case of Gaussian based lithofacies geostatistical models, like truncated Gaussian model or pluri-Gaussian model. As Gaussian related, a KL expansion could be performed on these models in order to reduce their dimensionality. Then, the sampling techniques addressed here could be used to perform the history matching.

Acknowledgements The author is very grateful to J. Jacod and E. Moulines for helpful discussions. Comments made by L.Y. Hu and D. Busby were also very helpful.

References

1. Andrieu, C., Jasra, A., Doucet, A., Moral, P.D.: On nonlinear Markoc Chain Monte Carlo via self-interacting approximations. Tech. rep., University of Bristol (2007)
2. Andrieu, C., Moulines, E.: On the ergodicity properties of some adaptive MCMC algorithms. Ann. Appl. Probab. **16**, 1462–1505 (2003)
3. Andrieu, C., Robert, C.: Controlled MCMC for Optimal Sampling. Tech. rep., Cérémade, Université de PARIS - DAUPHINE (2001)
4. Caers, J., Hoffman, T.: The probability perturbation method: a new look at bayesian inverse modeling. Math. Geol. **38**(1), 81–100 (2006)
5. Chen, Y., Zhang, D.: Data assimilation for transient flow in geologic formations via ensemble Kalman filter. Adv. Water Res. **29**, 1107–1122 (2006)
6. Dostert, P., Efendiev, Y., Hou, T., Luo, W.: Coarse-gradient Langevin algorithms for dynamic data integration and uncertainty quantification. J. Comput. Phys. **217**(1), 123–142 (2006)
7. Earl, D., Deem, M.: Parallel tempering : theory, applications, and new perspectives. Phys. Chem. Chem. Phys. **7**, 3910–3916 (2005)
8. Gavalas, G., Shah, P., Seinfeld, J.: Reservoir history matchng by Bayesian estimation. SPE J. **16**(6), 337–350 (1976)
9. Geyer, C.: Markov chain Monte Carlo maximum likelihood. In: Computing Science and Statistics: Proceedings of 23rd Symposium on the Interface Foundation, p. 156. American Statistical Association, Fairfax Station, New York (1991)
10. Ghanem, R., Spanos, P.: Stochastic Finite Elements, a Spectral Approach. Springer, New York (1991)
11. Haario, H., Saksman, E., Tamminen, J.: An adaptive metropolis algorithm. Bernoulli **7**, 223–242 (2001)
12. Holden, L., Sannan, S., Soleng, H., Arntzen, O.: History Matching using Adaptive Chains. Tech. rep., Norwegian Computing Center (2002)
13. Hu, L.Y.: Gradual deformation and iterative calibration of Gaussian-Related stochastic models. Math. Geol. **32**(1), 87–108 (2000)
14. Iba, Y.: Extended ensemble Monte Carlo. Int. J. Modern Phys. C **12**(5), 623–656 (2001)
15. Kitanidis, P.: Quasi-linear geostatistical theory for inversing. Water Resour. Res. **31**(10), 2411–2419 (1995)
16. Kou, S., Zhou, Q., Wong, W.: Equi-energy sampler with applications in statistical inference and statistical mechanics. Ann. Stat. **34**(4), 1581–1619 (2006)
17. Lehoucq, R., Sorensen, D., Yang, C.: ARPACK Users' Guide: Solution of Large Scale Eigenvalue Problems with Implicitly Restarted Arnoldi Methods. SIAM, Philadelphia (1997)
18. Loève, M.: Probability Theory. Princeton University Press, Princeton (1955)
19. Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A.T.M.: Equations of state calculations by fast computing machines. J. Chem. Phys. **21**, 1087–1091 (1953)
20. Oliver, D.: On conditional simulation to inaccurate data. Math. Geol. **28**(6), 811–817 (1996)
21. Oliver, D., Reynolds, A., Bi, Z., Abacioglu, Y.: Integration of production data into reservoir models. Pet. Geosci. **7**(9), 65–73 (2001)
22. Oliver, D.S., Cunha, L.B., Reynolds, A.C.: Markov chain Monte Carlo methods for conditionning a permeability field to pressure data. Math. Geol. **29**(1), 61–91 (1997)
23. Robert, C., Casella, G.: Monte-Carlo Statistical Methods, 2nd edn. Springer, New York (2004)
24. Romary, T., Hu, L.: Assessing the dimensionality of random fields with Karhunen–Loève expansion. In: Petroleum Geostatistics 2007. EAGE, Cascais, 10–14 September 2007
25. Sarma, P., Durlofsky, L., Aziz, K., Chen, W.: Efficient real-time reservoir management using adjoint-based optimal control and model updating. Comput. Geosci. **10**(1), 3–36 (2006)
26. StreamSim Technologies, Inc.: 3dsl User Manual, Version 2.10 edn. (2003)
27. Tarantola, A.: Inverse Problem Theory and Model Parameter Estimation. SIAM, Philadelphia (2005)