Comput Geosci (2015) 19:1–15
DOI 10.1007/s10596-014-9444-9

ORIGINAL PAPER

Iterative Bayesian inversion with Gaussian mixtures: finite sample implementation and large sample asymptotics

Andreas S. Stordal

Received: 27 December 2013 / Accepted: 28 August 2014 / Published online: 17 September 2014

Abstract Approximate solutions for Bayesian estimation in large scale models is a topic under investigation in many scientific communities. We define an iterative method based on the adaptive Gaussian mixture filter with batch updates as a robust alternative to adaptive importance sampling. We prove asymptotic optimality under certain conditions, contrary to other methods discussed where the sample distribution depends on the nonlinearity and scaling of the model. The finite sample implementation of the method is compared to an ensemble smoother with multiple data assimilation and an ensemble-based randomized maximum likelihood approach on a synthetic 1D reservoir model.

Keywords Bayesian estimation · Ensemble methods · Gaussian mixtures · Iterative importance sampling · Data assimilation

Mathematics Subject Classification (2010) 62H · 86

1 Introduction

Bayesian estimation of parameters and initial conditions in high dimensions arise in many scientific areas such as reservoir engineering, hydrology, oceanography, and meteorology. Although many of these systems evolve in time and discrete time measurements arrive sequentially, they are often modeled as deterministic and hence completely described by their parameters and initial condition. In theory, the distribution to consider is a change of measure usually with respect to a Gaussian reference measure on a Hilbert space [40]. However, in practice, the initial conditions and parameters are typically discretized in space and one often seeks the corresponding finite-dimensional distribution. A typical example is reservoir engineering where the equations describing the fluid flow are determined by geological parameters, such as porosity and permeability, modeled as random fields that do not change in time. To solve the flow equations numerically on a grid, these fields are discretized in a procedure denoted as upscaling (e.g., [9]) where the value represents an average value of the corresponding grid cell. The aim is then to estimate the finite-dimensional distribution given the available data and prior geological information. Although the methodology we discuss here is applicable in many settings, our main focus is on large scale models where evaluation of the model and/or the measurement operator is extremely time-consuming.

Approximate solutions to the Bayesian inverse problem in terms of Monte Carlo methods such as Markov Chain Monte Carlo (MCMC), importance sampling, or sequential Monte Carlo (SMC) are in theory applicable since the posterior is assumed to be known up to a proportional constant. However, for large scale models, each evaluation of the likelihood requires the numerical solution of a large set of partial differential equations which are very time-consuming to solve. This puts an upper bound on the number of samples/iterations in MCMC and SMC methods and is in contrast to the curse of dimensionality in SMC methods as discussed in, e.g., [2], where it is suggested that the sample size should increase exponentially with the dimension of the problem. More recent publications provide analysis of stability of such methods in large dimension for finite sample size [3]. The method proposed, however, requires $O(Nd^3)$ computations, where N is the sample size and d is the dimension of the system. For large scale applications, d is often of the order $10^4 - 10^{12}$, and the

A. S. Stordal (✉)
IRIS, Bergen, Norway
e-mail: Andreas.S.Stordal@iris.no

computational cost of the proposed method is therefore far beyond feasible for large scale models anytime in the near future. A recent MCMC approach has been shown to speed up the convergence of old MCMC algorithms [11], although MCMC approaches are still considered too time-consuming for large scale applications (see, e.g., [32]). The consequence is that in many applications, simpler approximation methods, which are asymptotically biased, are applied. Perhaps, the most popular of these methods is the ensemble Kalman filter (EnKF) in different variants [18].

The EnKF can be viewed as a Monte Carlo version of the Kalman filter, where the first- and second-order moments are estimated from the sample itself. For linear Gaussian models, it converges to the optimal solution provided by the Kalman filter, whereas in nonlinear models, there is an asymptotic bias [28]. EnKF was introduced to tackle nonlinear dynamics better than, e.g., the extended Kalman filter (EKF) as it evaluates the fully nonlinear model instead of a tangent linear model at the current estimate. In the EnKF, the data are assimilated sequentially in time; however, there is also a smoother version (EnKS, [41]) that assimilates all data in a given time window at once. For parameters and initial conditions, as in history matching, we use the word smoother in the following to define any algorithm that updates all time instances of data at ones in contrast to a filter that incorporates data sequentially according to their time indices.

Since both EnKF and EnKS incorporate the measurements with an assumption of a Gauss-linear model, the bias may be severe in strongly nonlinear models. Here, we focus on methods that improve EnKS for these kind of problems.

By incorporating ideas from a Gauss-Newton scheme, a new EnKS approach with multiple linear update steps have recently been suggested for reservoir characterization ([16] see also [10]). To avoid overfitting the data, one has to inflate the data measurement error covariance matrix in each update step. This method is known as ensemble smoother with multiple data assimilation (ES-MDA) in the reservoir community, and it has shown promising results for nonlinear models where EnKS and EnKF provide poor solutions [16].

A second approach we consider here is quasi-linear inversion also known as randomized maximum likelihood [25, 33] denoted RML in the following. It is defined as a Monte Carlo method where each sample is obtained by minimizing a stochastic objective function. If the model under consideration is linear and the prior and likelihood are both Gaussian, RML coincides with sampling from the Gaussian posterior density. Numerical applications of RML require gradients, and although it is computationally faster than MCMC approaches, computing gradients (or gradient approximations) is still time-consuming in large scale models unless an adjoint code is available. Lately, a new implementation, known as ensemble randomized maximum likelihood (EnRML), was introduced to reduce the computation time of gradient approximations at the expense of precision of the gradient which is roughly estimated by an ensemble (see, e.g., [7, 8] and references therein). Here, we focus on gradient-free implementations since it resembles the ES-MDA approach.

Another recent data assimilation technique based on Gaussian mixtures was introduced in [38] as a bridge between EnKF and particle filters. For Bayesian inversion, [37] introduced an iterative version as an approximation of adaptive importance sampling [21]. Here, we extend the iterative version of the filter to a smoother one and show the similarity of finite sample implementation with that of EnRML and ES-MDA. The algorithm, denoted as the iterative adaptive Gaussian mixture smoother (IAGS), is then available for models where batch updating is preferred over sequential updating.

We show that the IAGS is very similar to both ES-MDA and EnRML in terms of implementation on large scale models; however, we also provide large sample asymptotics for general nonlinear models and show that the method is optimal, as the number of samples and iterations increase unlike the other two methods. The IAGS does not assume a Gaussian prior, which is implicitly assumed in the ES-MDA and EnRML. In addition, we show that IAGS can be implemented similarly to a version of EnRML and ES-MDA, but with additional importance weights. This leads us to believe that the new method performs at least as well as the existing methods for any given nonlinear problem.

In the next section, we discuss Bayesian parameter estimation and adaptive importance sampling and introduce and the IAGS algorithm. Asymptotic properties of IAGS are studied in Section 3 . The implementation of IAGS is compared to the implementation of ES-MDA and EnRML in Section 4 with applications including a 1D reservoir problem. Finally, the paper is summarized in Section 5.

2 Bayesian estimation

In the Bayesian framework, we assume that the random variable X, defined on an infinite dimensional Hilbert space $\mathcal{H}$, follows a prior probability measure $\mu(x)$. The measurements, Y, are noisy realizations of the nonlinear operator $\mathcal{G}(X)$. The posterior measure, $\pi(x)$, is described via the Radon-Nikodym derivative as

$$\frac{d\pi}{d\mu}(x) \propto \exp\{-\Psi(x)\}, \tag{1}$$

for some functional $\Psi(x)$. In practice, the X is discretized in space (or in time), and we will in the following deal with the finite-dimensional densities for a given discretization.

Two typical scenarios are that X either consists of parameters and initial conditions with prior probability measure $\mu(x)$ or $X = (X_1, \ldots, X_T)$, a time series over a finite time interval, $[1, T]$, with $\mu(x)$ the joint probability measure of $X_1, \ldots, X_T$. The measurements are assumed to be a random mapping of some or all elements of X. Our focus is on the first scenario, and we restrict ourselves to systems defined by

$$\begin{aligned} &X \sim \mu(x), \\ &Y = \mathcal{G}(X) + \eta, \end{aligned} \tag{2.2}$$

where $\mathcal{G}$ is the (nonlinear) measurement operator and η is typically a 0 mean Gaussian variable with covariance matrix R. Note that the measurement vector Y may contain measurements taken at different time instances. We define $g(x, y) \overset{\text{def}}{=} p_{Y|X}(y|x)$, and for $Y = y$ fixed, we define the likelihood function $g(x) \overset{\text{def}}{=} g(x, y)$.

Given a set of the measurements, $y = (y_1, \ldots, y_{n_y})$, the solution to the discretized Bayesian estimation problem is given by the posterior density

$$\pi(x) = p_{X|Y}(x|y) = \frac{\mu(x)g(x)}{\int \mu(v)g(v)\,dv}. \tag{2.3}$$

If $\mu(x) = \phi(x|m, \Sigma)$ and $\eta \sim \phi(\eta|0, R)$, where ϕ denotes a multivariate Gaussian density, then

$$\pi(x) \propto \exp\{-0.5\|x - m\|^2_\Sigma - 0.5\|y - \mathcal{G}(x)\|^2_R\}, \tag{2.4}$$

where $\|u\|^2_V = u^T V^{-1} u$.

Next, we describe one of the many Monte Carlo methods that reproduce the posterior, π, as the number of samples increase. The method, denoted iterative (or sometimes adaptive) importance sampling, is the starting point for the IAGS algorithm.

2.1 Iterative importance sampling in the Bayesian framework

Assume that we have a model described by Eq. 2.2 and that we want to draw a sample from the posterior distribution $\pi(x)$ given by Eq. 2.3. In this case, η can be an arbitrary random variable, admitting a known density function. Since we can evaluate π up to a normalizing constant, for a given x (even if $\mathcal{G}$ is a black box), any type of MCMC or Accept-Reject algorithm can be used. However, we focus on importance sampling methods in the following. For any density $q(x)$ with at least the same support as $\pi(x)$, we may compute, for a given function f integrable w.r.t π,

$$\begin{aligned} &I_N(f) = \textstyle\sum_{i=1}^N f(X_i) \left\{ \frac{W(X_i)}{\sum_{j=1}^N W(X_j)} \right\}, \\ &W(X_i) = \frac{\mu(X_i)g(X_i)}{q(X_i)}, \end{aligned}$$

where $\{X_i\}_{i=1}^N$ are i.i.d. samples from q. Independent of q, $I_N(f)$ converges to $\pi(f) = \mathbf{E}[f(x)|Y = y]$ with a typical convergence rate of $N^{-1/2}$ if the variance of the estimator is finite. However, for a finite sample size, the precision of $I_N(f)$ is highly dependent on q. As an example, suppose that π is Gaussian with mean m and unit variance, q is a Gaussian with mean 0 and unit variance and $f(x) = x$, then (without normalizing the weights W)

$$\text{Var}(I_N) = \frac{\exp(m^2)(4m^2 + 1) - m^2}{N}.$$

We see that $m = 0$ gives us the a variance of N^{-1} which is expected since we then sample from π. However, if we change the mean of q from 0 to τ, then

$$\text{Var}(I_N) = \frac{\exp((m - \tau)^2)((2m - \tau)^2 + 1) - m^2}{N}.$$

Thus, the variance is reduced when τ approaches m. In fact, $\text{Var}(I_N)$ converges to N^{-1} as $\tau \to m$. In the above example, all constants are known, so it is slightly different than the Bayesian framework; however, it should serve as a motivating example.

In a Bayesian setting, where the scale and shape of π is often difficult (or impossible) to know in advance, one is often left with the prior μ as the (initial) proposal q. This simple and naive direct importance sampling approach is extremely variance prone, particularly in large scale applications due to the limited sample size, and the variance depend on the quantities μ, π, and f and q. A simple way to address this problem (at least in theory) is to apply adaptive (or iterative) importance sampling (see, e.g., [21, 31] and references therein). The idea is to use the weighted sample from an importance sampling algorithm to construct a new proposal q and use this in the next iteration of the importance sampling algorithm. That is, one performs importance sampling iteratively, and at each iteration, the proposal q is updated. Given a weighted sample $\{W^i_{j-1}, X^i_{j-1}\}_{i=1}^N$ at iteration $j - 1$, the proposal at iteration j, q_j, is defined by smoothing the empirical measure $\pi^N_{j-1}(x) \overset{\text{def}}{=} \sum_{i=1}^N W^i_{j-1}\delta_{X^i_{j-1}}(x)$ with a symmetric, positive, and bounded kernel K,

$$q_j(x) = \pi^N_{j-1} * K(x) \overset{\text{def}}{=} \sum_{i=1}^N W^i_{j-1} K(x - X^i_{j-1}). \tag{2.5}$$

The kernel K is arbitrary, but one must always make sure that the support of q_j contains the support of π. Typically, the kernel is selected as $K_h(u) = h^{-d}K(u/h)$ where h is a bandwidth decreasing with the sample size. A new sample is then drawn independently from q_j and the importance weights are computed as

$$W^i_j = \frac{\pi(X^i_j)}{q_j(X^i_j)}, \; i = 1, \ldots, N.$$

Since W^i_j is known only up to a proportional constant, they have to be normalized. The methodology has been extended to adaptive multiscale importance functions in [5] and other adaptive techniques such as an MCMC approach combined with an annealing sequence to find an improved importance sampler [30]. More recent approaches involve the adaptive use of particle filters in an MCMC context [1] and adaptive approaches for Bayesian computing is found in, e.g., [12]. More general SMC methods to make parallel Markov chain Monte Carlo algorithms interact for global estimation are discussed in [14]. Convergence properties of adaptive importance sampling can be found in, e.g., [13].

The literature on adaptive and iterative Monte Carlo methods is vast, detailed, and include a lot more than what is cited above and in the introduction. However, the aim of this section is to give an introduction to the concept of iterative importance sampling as the IAGS presented in the next section is an approximation of this algorithm and not to study iterative importance sampling in itself. Before we describe the IAGS algorithm, we present an algorithm for the iterative importance sampling in Algorithm 1.

Algorithm 1 Iterative importance sampling

```
while $j \leq J$ do
    if $j = 0$ then
        Set $q_0 = \mu$
        for $i = 1 : N$ do
            Sample $X_0^i \sim q_0$
            Compute $W^i \propto \pi(X_0^i)/q_0(X_0^i)$
        end for
    else
        Compute $q_j^N(x) = \sum_{i=1}^N K_h(x - X_{j-1}^i)W_{j-1}^i$
    end if
    for $i = 1 : N$ do
        Sample $X_j^i \sim q_j^N$
        Compute $W_j^i \propto \pi(X_j^i)/q_j^N(X_j^i)$
    end for
    Approximate posterior $\pi$ with $\{W_j^i, X_j^i\}_{i=1}^N$
    $j \leftarrow j + 1$
end while
```

The importance sample may suffer from large variance at the first iteration, and it may take many iterations before one starts to sample from regions of high likelihood if the dimension of the problem is relatively large, especially if the prior has a large uncertainty. In addition, as mentioned before, the high dimension of the state space and/or measurements space in large scale models, such as reservoir engineering, would require a number of samples that is not in agreement with the computational power at hand which typically constrains us to a sample size of $\mathcal{O}(100)$. In order to construct a more robust estimator for large scale models, we turn to the theory of Gaussian mixture filters and iterative Gaussian mixture filters.

2.2 Iterative adaptive Gaussian mixture smoother

An iterative version of the adaptive Gaussian mixture filter (AGM) [38] was introduced in [37]. Since this version was defined sequentially in time, the computation time of the original AGM would be multiplied by the number of iterations. Here, we formulate an iterative version using batch updates, that is, we include all measurements in each update step. The IAGS updates the variables of interest, X, using the covariance with the simulated measurements, $\mathcal{G}(X)$, similar to EnKF, EnKS, and EnRML. Although it has been demonstrated that sequential updates are in general better than batch update for nonlinear models [19], algorithms using batch updates have some other advantages. For large scale models, they do not require stopping and restarting the numerical model at each observation time. This gives the batch update a computational advantage over sequential methods since it is usually much faster to solve the numerical equations over a long time period. Another advantage in reservoir engineering is that there is no need to update dynamical states, such as pressure and saturation variables. This reduces the computational burden as well as the storage space on the computer. Hence, iterative methods, which are required for strongly nonlinear systems, are in general computationally cheaper for large scale systems when batch updates are implemented. We are therefore motivated to define the IAGS as an extension of the IAGM.

The derivation of IAGS follows that of AGM for a deterministic system with one-time step in combination with iterative importance sampling algorithm. In other words, it may be derived in the same manner as Gaussian mixture filters (see, e.g., [6, 20, 23, 27]) with an additional weight reduction combined with a resampling and reweighting step between each iteration.

Before we define the algorithm, we introduce some notation. First, we augment the variable $X \in \mathbb{R}^d$ with $\mathcal{G}(X) \in \mathbb{R}^{n_y}$ to $Z = [X^T \; \mathcal{G}(X)^T]^T$ such that we may write (2.2) as

$$Y = HZ + \eta, \tag{2.6}$$

where H is a matrix consisting of zeros and ones. This is for notational convenience, and we stress that this should not be confused with a linear measurement operator for the original model.

Given a weighted sample $\{\widehat{X}_{j-1}^i, \; \widehat{W}_{j-1}^i\}_{i=1}^N$, approximating the posterior at iteration $j - 1$ by

$$\widehat{\Psi}_{j-1}^N = \sum_{i=1}^N \widehat{W}_{j-1}^i \delta_{\widehat{X}_{j-1}^i}.$$

We denote by $\phi(u|\mu, P)$ a Gaussian density with mean μ and covariance matrix P and define a density, $\widehat{\Psi}_{j-1}^N * \phi_h$

as the convolution between the Gaussian kernel $\phi_h(u) = \phi(u|0, h^2\widehat{\Sigma}^X_{j-1})$, where $\widehat{\Sigma}^X_{j-1}$ is a symmetric positive definite $d \times d$ matrix, and the empirical distribution defined by the weighted sample,

$$\widehat{\Psi}^N_{j-1} * \phi_h(x) = \sum_{i=1}^{N} \widehat{W}^i_{j-1}\phi\left(x|\widehat{X}^i_{j-1}, h^2\widehat{\Sigma}^X_{j-1}\right), \tag{2.7}$$

where h is the bandwidth parameter of the Gaussian kernel.

At iteration j, we start by sampling N particles, $\{X^i_j\}^N_{i=1}$ from Eq. 2.7. If $j = 1$, we set $\widehat{\Psi}^N_0 * \phi_h = \mu$. That is, initially, we start the algorithm by sampling from the prior. Then, we compute $Z^i_j = [(X^i_j)^T\ \mathcal{G}(X^i_j)^T]^T$, $i = 1, \ldots, N$ and approximate the prior density of Z with a weighted kernel estimator

$$\begin{aligned} P^N_{j,h}(z) &\overset{\text{def}}{=} \sum_{i=1}^{N} W^i_j\phi\left(z; Z^i_j, h^2\Sigma^Z_j\right), \\ W^i_j &\overset{\text{def}}{=} \frac{\mu(X^i_j)}{\widehat{\Psi}^N_{j-1}*\phi_h(X^i_j)}, \end{aligned} \tag{2.8}$$

where Σ^Z_j is a symmetric positive definite $(d+n_y)\times(d+n_y)$ matrix. The weights, W, are corrections for not using the prior as importance function. Next, we use that the measurement error is Gaussian so that the likelihood function $g(x)$ is given by as $g(x) = \phi(y|\mathcal{G}(x), R) = \phi(y|Hz, R)$ and inserting the empirical estimate (2.8) in Bayes' theorem, we get

$$P^N_{j,h}(z|y) \propto \left[\sum_{i=1}^{N} W^i_j\phi\left(z|Z^i_j, h^2\Sigma^Z_j\right)\right]\phi(y|Hz, R). \tag{2.9}$$

The Gaussian mixture in Eq. 2.9 can be rewritten as (see, e.g., [27])

$$P^N_j(z|y) \propto \sum_{i=1}^{N} \widehat{W}^i_j\phi\left(z; \widehat{Z}^i_j, \widehat{\Sigma}^Z_j\right),$$

where

$$\begin{aligned} \widehat{Z}^i_j &= Z^i_j + K_j(y - HZ^i_j), \\ K_j &= \Sigma^Z_j H^T(H\Sigma^Z_j H^T + h^{-2}R), \\ \widehat{\Sigma}^Z_j &= (I - K_jH)\Sigma^Z_j, \\ \widehat{W}^i_j &= W^i_j\phi(y; HZ^i_j, h^2H\Sigma^Z_jH^T + R). \end{aligned} \tag{2.10}$$

The expressions in Eq. 2.10 can be derived using the exact same calculations as in the classical Kalman filter. The proportionality in the weight equation in Eq. 2.10 simply means that the weights $\widehat{W}^i_j$, $i = 1, \ldots, N$ are normalized. If we ignore h for a moment, we see that the only difference with the classical Kalman filter is the weights which occur due to the fact that we have N and not one Gaussian components. This should not come as a surprise since each Gaussian kernel in Eq. 2.8 can be interpreted as an additional Gaussian model error. There is a strong link between the optimal SIR filter in a nonlinear-state space model with linear measurement operator and Gaussian model and measurement error. For such models, the (conditional) optimal importance function $p(x_t|y_t, x_{t-1})$ is a Gaussian density with parameters obtained in a similar manner as (2.10) (see, e.g., [34]).

Remark 1 The denominator in Eq. 2.8 requires $O(N^2)$ operations to evaluate, and it may be smarter to use a few clusters, instead of N, in the mixture when sampling before next iteration if N is large (see, e.g., [22]).

To avoid a weight collapse, the weights are shrunk towards uniform weights as in the AGM

$$\widehat{W}^{i,\alpha}_j = \alpha_j\widehat{W}^i_j + (1 - \alpha_j)N^{-1}, \tag{2.11}$$

where $0 < \alpha_j \leq 1$.

For any measurable function f, integrable w.r.t. π, we approximate $\pi(f) = \mathbf{E}_\pi[f(X)|Y = y]$ with

$$\widehat{\Psi}^N_j(f) \overset{\text{def}}{=} \sum_{i=1}^{N} \widehat{W}^{i,\alpha}_j f(\widehat{X}^i_j), \tag{2.12}$$

and to move to iteration $j + 1$, we use the kernel estimate

$$\widehat{\Psi}^N_j * \phi_h(x) = \sum_{i=1}^{N} \widehat{W}^{i,\alpha}_j\phi\left(x|\widehat{X}^i_j, h^2\widehat{\Sigma}^X_j\right), \tag{2.13}$$

as the new proposal at iteration $j+1$, where $\widehat{\Sigma}^X_j$ is the $d \times d$ upper left matrix of $\widehat{\Sigma}^Z_j$.

In practice, Σ^Z_j is often chosen as the sample covariance matrix of $\{Z^i_j\}^N_{i=1}$,

$$\Sigma^Z_j = (N - 1)^{-1}\sum_{i=1}^{N}(Z^i_j - \overline{Z}_j)(Z^i_j - \overline{Z}_j)^T,$$

where $\overline{Z_j} = N^{-1}\sum_{i=1}^{N} Z^i_j$. Hence, we see that with $\overline{\mathcal{G}}(X_j) = N^{-1}\sum_{i=1}^{N}\mathcal{G}(X^i_j)$, $\Sigma^Z_jH^T$ in the linear update in Eq. 2.10 is given by

$$\Sigma^Z_jH^T = (N - 1)^{-1}\sum_{i=1}^{N}(X^i_j - \overline{X}_j)(\mathcal{G}(X^i_j) - \overline{\mathcal{G}}(X_j))^T \tag{2.14}$$

which is the same sample cross-covariance also used in EnKS update; however, the update is scaled down due to the factor h^{-2} coming from the kernel ϕ_h. It is also worth pointing out that when d is so large that Σ^Z_j cannot be stored on a computer, we may factorize Σ^Z_j into $L^Z_j(L^Z_j)^T$ so that we only have to store the $d \times N$ matrix L_j. For this particular choice, line three in Eq. 2.10, although possible to solve for the updated ensemble [23], may be numerically unstable, and a standard implementation is to add $\eta^i \sim \phi(\eta|0, R)$ to y in Eq. 2.10 and let

$$\widehat{\Sigma}^z_j = (N-1)^{-1} \sum_{i=1}^{N} (\widehat{Z}^i_j - \overline{\widehat{Z}}_j)(\widehat{Z}^i_j - \overline{\widehat{Z}}_j)^T .$$

This ensures that (2.10) is asymptotically satisfied (see, e.g., [4]) while the expected value of each particle remains the same. Alternatively, a square root implementation can be used in the same way as in the square root EnKF (see, e.g., [35]). Also note that the truncated singular value decomposition (tsvd) used in [8] or the subspace tsvd from [17] can be applied to the inversion term in Eq. 2.10 when the number of data is large.

In theory, we can choose the weight shrinkage factor α_j arbitrarily. Often in practice, an adaptive α_j is used [38] computed as

$$\alpha^N_j = \frac{1}{N \sum_{k=1}^{N} \left(\widehat{W}^k_j \right)^2} . \tag{2.15}$$

With this weight shrinkage, the new estimated effective sample size [26] is always greater than 80 %. Also, since $0 < \alpha^N_j \leq 1$, we always take into account some of the nonlinear information contained in the weights. Obviously, the weight interpolation introduces bias in addition to the bias introduced by h; however, as we show in the next section, this bias is reduced at each iteration. We emphasize that it is fully possible to choose h differently in Eqs. 2.7 and 2.9 and to let h change at each iteration. The latter also applies to N.

Remark 2 If μ is Gaussian and the dimension of X is very large, an approximation of the mean and covariance from the ensemble based on a truncated SVD can be used. This will reduce the computational cost. If μ is not Gaussian, a further approximation can be made, either by ignoring the initial reweighting or by applying the weight reduction to W^i_j before multiplying by $\phi\left(z|\widehat{Z}^i_{j-1}, h^2_{j-1}\Sigma^Z_{j-1}\right)$ or as often is common for particle filter in high dimensions truncate the density somewhere in the tail so that all samples will have positive weight. This truncation should of course go to 0 as N goes to infinity as in [24].

Compared to the other iterative methods, as discussed in Section 4, the IAGS has the advantage of not having a Gaussian prior implicitly assumed in the methodology.

The IAGS algorithm is summarized in Algorithm 2.

Remark 3 If the sensitivity matrix G of $\mathcal{G}(x)$ w.r.t. x is available, then the Kalman gain K_j can be computed locally using $\Sigma^X_j G^T$ and $G\Sigma^X_j G^T$ in place of $\Sigma^Z_j H^T$ and $H\Sigma^Z_j H^T$. Other approaches in strongly nonlinear systems are suggested in [39] (and references therein).

Algorithm 2 IAGS algorithm

```
while $j \leq J$ do
    if $j = 1$ then
        Set $\widehat{\Psi}^N_{j-1} * \phi_h(x) = \mu(x)$
        Set $\widehat{W}^{i,\alpha}_{j-1} = N^{-1}$
    else
        Set $\widehat{\Psi}^N_{j-1} * \phi_h(x) = \sum_{i=1}^N \widehat{W}^{i,\alpha}_{j-1} \phi(x|\widehat{X}^i_{j-1}, h^2\widehat{\Sigma}^X_{j-1})$
    end if
    for $i = 1 : N$ do
        Sample $X^i_j \sim \widehat{\Psi}^N_{j-1} * \phi_h(x)$
        Set $Z^i_j = [(X^i_j)^T \, \mathcal{G}(X^i_j)^T]^T$
        Compute:
            $W^i_j = \frac{\mu(X^i_j)}{\widehat{\Psi}^N_{j-1} * \phi_h(X^i_j)}$
    end for
    Compute:
    $\overline{Z_j} = N^{-1} \sum_{i=1}^N Z^i_j$
    $\Sigma^Z_j = (N-1)^{-1} \sum_{i=1}^N (Z^i_j - \overline{Z}_j)(Z^i_j - \overline{Z}_j)^T$
    $K_j = \Sigma^Z_j H^T (H\Sigma^Z_j H^T + h^{-2}R)^{-1}$
    for $i = 1 : N$ do
        Update $\widehat{X}^i_j = X^i_j + K_j(y - HZ^i_j + \eta^i)$, where $\eta^i \sim$
        $\phi(0, R)$.
        Evaluate $\widehat{W}^i_j = \frac{W^i_j \phi(y|HZ^i_j, h^2 H\Sigma^Z_j H^T + R)}{\sum_{\ell=1}^N W^\ell_j \phi(y|HZ^\ell_j, h^2 H\Sigma^Z_j H^T + R)}$
    end for
    Compute:
    $\widehat{W}^{i,\alpha}_j = \alpha_j \widehat{W}^i_j + (1-\alpha_j)N^{-1}, \quad i = 1, \ldots, N$
    Approximate posterior $\pi$ with $\{\widehat{W}^{i,\alpha}_j, \ \widehat{X}^i_j\}^N_{i=1}$
    $j \leftarrow j + 1$
end while
```

We also notice that with $j = 1$, $h = 1$, and $\alpha = 0$, the IAGS algorithm is reduced to the EnKS, while if $h = 0$ in Eq. 2.9 and $\alpha = 1$, it is a version of the iterative importance sampling algorithm described in the previous section. Finally, we would like to point out that the kernels in Eqs. 2.7 and 2.9 may be selected differently including the bandwidth parameter h. We also stress that both h and the sample size N could change at each iteration. However, in the next section, we keep N fixed at each iteration and let h be a function of N when studying asymptotic properties of the algorithm.

3 Large sample asymptotics of IAGS

We study a weak type of convergence of a sequence of empirical measures $\{\widehat{\Psi}^N_j\}$ to π as N increases with a fixed (but possibly large) number of iterations J,

$$\lim_{N \to \infty} \mathbf{E} \left| \widehat{\Psi}^N_j(f) - \pi(f) \right|, \tag{31}$$

where $\widehat{\Psi}_j^N(f)$ is our empirical approximation of $\pi(f) = \mathbf{E}_\pi[f]$ for a certain type of functions. The expectation in Eq. 31 is w.r.t. the particle system.

The outline of the proof is as follows. At $j = 0$, we have the prior μ which is known. By defining $\widehat{\Psi}_0^N(f) = \mu(f)$, our initial error is simply $|\mu(f) - \pi(f)|$.

The key point of the proof is to consider two particle systems $\{X_j^i, W_j^i\}_{i=1}^N$ and $\{\widehat{X}_j^i, \widehat{W}_j^i\}_{i=1}^N$ giving rise to two different approximations of $\pi(f)$, namely $\widehat{\Psi}_j^N = \sum_i \widehat{W}_j^i f(\widehat{X}_j^i)$ and $\Psi_j^N = \sum_i W_j^i f(X_j^i)$. The former is a result of the IAGS algorithm at iteration j and the latter is a result of IAGS algorithm at iteration j with $h = 0$. Both, however, use the same proposal from the IAGS algorithm at iteration $j-1$. The reason for studying these two particle systems is the following: Since we know the initial error in terms of the prior and posterior, it is relatively simple to compute the error of an importance sampler with weight shrinkage. The first part of the estimator, using the a proportion of the correct weights, is asymptotically unbiased. The second part, using uniform weights, is simply a sample from the prior, and hence, the error can be quantified using the initial error. In order to quantify the additional error using linear updating, we simply couple this set of particles with a sample that is linearly updated. Since we can describe the difference between the updated sample and the original sample in terms of the bandwidth parameter h, we can then describe the error of the sample that is linearly updated. Since this sample is used to construct the next proposal, we can implement the same strategy again to compute the error of the nest iteration in terms of sampling error and the error from the previous iteration.

The particles $\{X_j^i\}_{i=1}^N$ are i.i.d. from $\widehat{\Psi}_{j-1}^N * K_h$, where $*$ is the convolution operator and K_h is a kernel with parameter h to be defined later. The weights $\{W_j^i\}_{i=1}^N$ are computed using the correct importance function. That is

$$g_j \overset{\text{def}}{=} \frac{\mu g}{\widehat{\Psi}_{j-1}^N * K_h}. \tag{3.2}$$

Hence, we can bound the error of $\Psi_j^N(f) \overset{\text{def}}{=} \alpha_j \sum_i W_j^i f(X_j^i) + (1-\alpha_j) N^{-1} \sum_i f(X_j^i)$ in terms of N and the error of $\widehat{\Psi}_{j-1}^N * K_h(f)$. The latter enters in the error since we are only using parts of the importance weights.

Then, $\{\widehat{X}_j^i\}_{i=1}^N$ is constructed from $\{X_j^i\}_{i=1}^N$ using the h-dependent linear interpolation from Eq. 2.10 on the augmented variables $Z_j^i = (X_j^i, \mathcal{H}(Z_j^i), i = 1, \ldots, N$. The variables $\{\widehat{X}_j^i\}_{i=1}^N$ are then simply the first components of the updated variables $\{\widehat{Z}_j^i\}_{i=1}^N$ In addition, the importance weights of $\{\widehat{W}_j^i\}_{i=1}^N$ are computed as in Eq. 3.2 with g replaced by $g * \tau_h$, where τ_h is a symmetric bounded kernel not necessarily the same as K_h. This gives rise to the approximation in Eq. 2.12

$$\widehat{\Psi}_j^N(f) = \alpha_j \sum_{i=1}^N \widehat{W}_j^i f(\widehat{X}_j^i) + (1-\alpha_j) N^{-1} \sum_{i=1}^N f(\widehat{X}_j^i). \tag{3.3}$$

If the kernels K_h and τ_h are selected as Gaussian and g is Gaussian, then this is same empirical approximation as in the IAGS algorithm. That is, if $K_h(u) = \phi(u|0, h^2\Sigma)$, $\widehat{\Psi}_{j-1}^N * K_h$ is a Gaussian mixture and if $\tau_h(u) = \phi(u|0, h^2 H\Sigma H^T)$, then $g * \tau_h(z) = \phi(y|Hz, h^2 H\Sigma H^T + R)$ when $g(z) = \phi(y|Hz, R)$.

For sufficiently small h, a Taylor expansion of both f and $g*\tau_h$ allows us to bound the difference between $\widehat{\Psi}_j^N(f)$ and $\Psi_j^N(f)$ in terms of h. Then, we use some triangle inequalities to bound the error of $\widehat{\Psi}_j^N(f)$ in terms of h, N and the error of $\widehat{\Psi}_{j-1}^N(f)$. Finally, letting $h = h(N) = N^{-1/4}$ and iterating on j, we end up with an error bound for $\widehat{\Psi}_j^N(f)$ in terms of N, j and the initial error $\mathbf{E}|\mu(f) - \pi(f)|$. Also note that for $h = 0$ in Eq. 2.9, $\{\widehat{X}_j^i\}_{i=1}^N = \{X_j^i\}_{i=1}^N$ and $\widehat{\Psi}_j^N(f) = \Psi_j^N(f)$.

We study the convergence for an arbitrary sequence $\{\alpha_j\}$ of weight shrinkage factors, although we believe that a similar result can be obtained with the adaptive choice in Eq. 2.15. To carry out the proof, we use the following assumptions. The proof itself is left to the Appendix.

3.1 Assumptions and definitions

A1 Let f be a bounded function $f : \mathbb{R}^d \to \mathbb{R}$ with continuous and bounded partial derivatives up to and including second order.

A2 For all $\nu_j \overset{\text{def}}{=} K_j(y - HZ_j)$ from Eq. 2.10, we have $\mathbf{E}|\nu_j|^2 < \infty$.

A3 Let K be a positive, symmetric, and bounded kernel such that

$$\int uK(u)\,du = 0, \qquad \int u^2 K(u)\,du < \infty$$

and define $K_h(u) = h^{-d} K(h^{-1}u)$

A4 For all j, we have almost surely w.r.t. π

$$\|g_j\| = \left\| \frac{\mu g}{\Psi_{j-1,h}^N * K_h} \right\| \le C_1^j, \qquad \left\| \frac{\mu D^{(2)} g}{\Psi_{j-1,h}^N * K_h} \right\| \le C_2^j,$$

where $D^{(2)} g$ is the second-order partial derivatives of g and $\|\cdot\|$ denotes the sup norm.

A5 For each j, we assume that $\eta_j^N(g_j) \overset{\text{def}}{=} N^{-1} \sum_{i=1}^N g_j(X_j^i) > \delta_j > 0$. With the above notation, $\eta_j^N(g_j f)$ is a number whereas $\eta_j^N g_j \overset{\text{def}}{=} N^{-1}\sum_{i=1} \delta_{X_j^i} g_j$ is an empirical measure.

A6 For fixed y, $||g|| \leq C_g$, for some finite constant C_g and where g is Borel measurable and twice differentiable with bounded partial derivatives.

The kernels in Eq. 3.1 are chosen as Gaussian in the IAGS algorithm; however, different kernels with heavier tails can be used when constructing the new importance function. Assumption Eq. 3.1 essentially says that the importance weights are bounded at each iteration and is similar to the assumption in [13] where it was also pointed out that ”a major finding of [5] is, however, that the dependence of the importance function on earlier proposals and realization does not jeopardize the fundamental importance sampling identity.” However, it is weaker than the uniform bound assumption in [31]. Note that Eq. 3.1 is satisfied if, e.g., both μ and g are Gaussian and that Eq. 3.1 is satisfied if the measurement error is additive and Gaussian.

Since $g(x)$ is the likelihood function given y and since the law of X is μ, we have that a version of $\pi(f)$ is given by

$$\pi(f) = \frac{\int \mu(x)g(x)f(x)\,dx}{\int \mu(x)g(x)\,dx} = \frac{\mu(gf)}{\mu(g)}. \tag{3.4}$$

3.2 Main results

We study two empirical estimates. The first, Ψ_j^N, is a weighted sum of dirac measures using an interpolation of correct importance weights g_j and uniform weights. The second, $\widehat{\Psi}_j^N$, is a weighted sum of dirac measures after each sample has been linearly updated. The weight function has also changed via a convolution. Ψ_j^N and $\widehat{\Psi}_j^N$ are constructed from a sample, $\{X_j^i\}_{i=1}^N$, at iteration j drawn from $\widehat{\Psi}_{j-1}^N * K_h$.

Lemma 1 *Under the assumptions* (3.1)–(3.1)*, the empirical measure* Ψ_j^N *satisfies*

$$\mathbf{E}\left|\Psi_j^N(f) - \pi(f)\right| \leq \frac{\|f\|}{\sqrt{N}} C_1^j (2C^{-1} + 1)$$
$$+\beta_j\left(\mathbf{E}\left|\widehat{\Psi}_{j-1}^N(f) - \pi(f)\right| + h^2 C_f\right),$$

for some constants C_1, C, C_f, *and* $\beta_j < 1$.

The error of Lemma 1 is easily interpreted. The first part of the r.h.s. is essentially the typical N convergence we get from importance sampling (this empirical measure is obtained with $h = 0$ at iteration j). The second part of the r.h.s originates from the weight shrinkage. As soon as $\alpha_j < 1$, a part of our estimate is simply an empirical measure obtained by sampling from a smoothed version of the empirical measure at the previous iteration. That is why h appears as well as $\widehat{\Psi}_{j-1}^N$.

Lemma 2 *Under the assumptions* (3.1)–(3.1)*, the empirical measures* Ψ_j^N *and* $\widehat{\Psi}_j^N$ *satisfy*

$$\mathbf{E}\left|\widehat{\Psi}_j^N(f) - \Psi_j^N(f)\right| \leq \|f\| h^2 F_j,$$

for some constant F_j *depending on* f *and* g.

Lemma 2 simply states that the difference between the empirical measures of IAGS starting at iteration j with $h = 0$ and $h > 0$ can be expressed as a function of h. Combining Lemma 1 and Lemma 2, we can prove Theorem 1.

Theorem 1 *Under the assumptions* (3.1)–(3.1) *and with the bandwidth selection* $h = h(N) = N^{-1/4}$*, the empirical measure* $\widehat{\Psi}_j^N(f)$ *defined in* Eq. 3.3 *from IAGS satisfies asymptotically for* $1 \leq j \leq J$

$$\mathbf{E}\left|\widehat{\Psi}_j^N(f) - \pi(f)\right| \leq \frac{\|f\| B}{\sqrt{N}}\left(1 + \sum_{\ell=1}^{j-1}\prod_{k=\ell+1}^{j}\beta_k\right)$$
$$+\left|\mu(f) - \pi(f)\right|\left(\prod_{k=1}^{j}\beta_k\right),$$

as N *goes to infinity with* B *being a constant and* $\beta_k = 1 - \alpha_k < 1$ *for all* k.

Theorem 1 gives us an indication on how the IAGS error is reduced as a function of the sample size and the number of iterations. Further, let $\beta = \sup \beta_k$, $k = 1, \ldots J$. Since J is finite, we have $\beta < 1$ and we may deduce that

$$\mathbf{E}\left|\widehat{\Psi}_j^N(f) - \pi(f)\right| \leq \frac{\|f\|}{\sqrt{N}}\frac{B}{1-\beta} + \left|\mu(f) - \pi(f)\right|\beta^j,$$

which makes it even easier to interpret the error. We know from both theory and experience that $\alpha = 1$ is not an option, as it would lead to a particle collapse; however, from Theorem 1, we see that we have the possibility to get very close to the Bayesian solution without ever selecting any $\alpha_k = 1$. This is very important in terms of robustness of the algorithm.

Although asymptotic theory is nice in itself, the IAGS algorithm is mainly proposed for high dimensional problems with limited sample size. Whereas the effect of the linear Kalman type update vanishes asymptotically, it cannot be stressed enough how important it is in applications with a small sample size. We therefore turn our focus to applications and compare the practical implementation of IAGS with two other algorithms that are well known in the petroleum reservoir community.

4 Implementation and comparison with ES-MDA and EnRML

In this section, we discuss the similarities and differences between IAGS, EnRML, and ES-MDA. Initially, the three algorithms are the same, starting with an i.i.d. sample from the prior μ, which we assume is Gaussian in the following. We follow the same notation as in Section 2 and augments each particle with the measurement operator to get $Z_=[X^T \quad \mathcal{G}(X)^T]^T$. All three methods discussed here update each augmented particle at iteration j with a perturbation $\delta_{Z^i_j}$. Hence,

$$\widehat{Z}^i_j = Z^i_j + \delta_{z^i_j}.$$

The quantity of interest, $\widehat{X}^i_j$, is then simply the first part of $\widehat{Z}^i_j$. For the IAGS update in Eq, 2.10, $\delta_{Z^i_j} = K_j(y - HZ^i_j)$, where K^i_j is a function of h given by $\Sigma^z_j H^T (H\Sigma^z_j H^T + h_j^{-2}R)^{-1}$. We use the full expression for K_j hereafter.

The EnRML has several implementations [8]; however, they are all ensemble-based approximations of the original RML which is defined to minimize the stochastic objective function

$$J(x)_{\xi,\eta} = \frac{1}{2}(x - m - \xi))^T C_p^{-1}(x - m - \xi) \qquad (4.1)$$
$$+\frac{1}{2}(y + \eta - \mathcal{G}(x))^T R^{-1}(y + \eta - \mathcal{G}(x)),$$

where m is the prior mean and C_p is the prior covariance matrix. A sample $\{X_i\}_{i=1}^N$ is obtained by minimizing, for each i, $J(x)_{\xi_i,\eta_i}$, where $\xi_i \sim \phi(\xi; 0, C_p)$ and $\eta_i \sim \phi(\eta; 0, R)$. Note that $J(x)_{0,0}$ is equal to $-\log(\pi(x))$ in Eq. 2.4 up to an additive constant; hence, RML has an implicit assumption of Gaussian prior and likelihood. Minimizing Eq. 4.1 results in a random variable distributed according to the posterior if $\mathcal{G}$ is linear in addition to the mentioned Gaussian assumptions. If we look at the Levenberg-Marquardt version of EnRML without the model mismatch term (denoted EnRMLapprox in [8]), the update step at iteration j is

$$\delta^{\text{EnRML}}_{Z^i_j} = \Sigma^z_j H^T [H\Sigma^z_j H^T + (1 + \lambda_j)R]^{-1} \qquad (4.2)$$
$$(y - HZ^i_j + \eta^i),\ i = 1, \ldots, N$$

where $\eta^i \sim \phi(\eta; 0, R)$. The number of iterations depends on the values $|J(\widehat{X}^i_j)_{\xi_i,\eta_i} - J(\widehat{X}^i_{j-1})_{\xi_i,\eta_i}|$ and $\|\widehat{X}^i_j - \widehat{X}^i_{j-1}\|$, and the algorithm is stopped when either of the two is below a predefined threshold.

For IAGS, we have seen that the update is given

$$\delta^{\text{IAGS}}_{Z^i_j} = \Sigma^z_j H^T [H\Sigma^z_j H^T + h_j^{-2}R]^{-1} \qquad (4.3)$$
$$(y - HZ^i_j + \eta^i),\ i = 1, \ldots, N$$

which is equivalent to the EnRML-approx update if $h_j^{-2} = 1 + \lambda_j$. Note that in EnRML-approx, the model mismatch term is dropped (to avoid inverting an $N_x \times N_x$ matrix), whereas in the IAGS, the model mismatch term never appears. At each iteration, we sample from the new importance function and correct the weights. This will preserve the prior model (avoid overfitting the data) at least for moderate sized problems. From this comparison, it is not surprising that IAGS performs at least as well as EnRML in large scale models and better for small scale applications.

The ES-MDA is an iterative smoother that also uses the sample covariance to update the ensemble. Each iteration is equivalent to an ordinary EnKS run where the data covariance matrix has been inflated by a factor k_j at iteration j such that after the last iteration, J, we have $\sum_{i=1}^J \frac{1}{k_i} = 1$. The update can be formulated as

$$\delta^{\text{ES-MDA}}_{Z^i_j} = \Sigma^Z_j H^T [H\Sigma^z_j H^T + k_j R]^{-1}(y - HZ^i_j + \eta^i_j), \qquad (4.4)$$

where $\eta_j \sim \phi(\eta|0, k_j R)$.

Remark 4 Note that the only difference in the update formula for the ES-MDA is the sample η^i_j. In EnRML and IAGS, $\eta^i \sim \phi(\eta|0, R)$ is kept fixed for all j, while in ES-MDA, $\eta^i_j \sim \Phi(\eta|0, k_j R)$ changes with each iteration.

In a linear Gaussian model, ES-MDA is equivalent (up to sampling error) to the EnKS, but ES-MDA can handle nonlinear models better [16].

A second difference with EnRML is that the ES-MDA does not evaluate the objective function after each iteration, and it is therefore not necessary to decrease the step size and rerun if the objective function has increased. EnRML and ES-MDA may in practice have very similar implementation (actually give almost identical results) if one relates the tuning parameter of EnRML to the inflation factor in ES-MDA. Similarly, the h parameter in IAGS can be tuned in the exact same way as λ, and the method only differs from the EnRML by the importance weights and the fact that we, at least at the final iteration, sample from a smoothed density and compute weights according to the prior in order to keep the theoretical relationship to the rigorous approach of iterative importance sampling. We also note that unlike EnRML and ES-MDA, the IAGS algorithm can be stopped after any iteration, that is, we can, in theory, iterate as many times as we want.

Although the number of iterations in ES-MDA is predefined, the following scheme gives ES-MDA a similar convergence criteria as EnRML.

- Select λ_1 as in EnRML and let the first inflation factor, k_1, satisfy

$$\frac{1}{k_1} = (1+\lambda_1)^{-1}.$$

- Iterate and tune λ_j as in EnRML until convergence has been reached then do one more iteration where k_{last} satisfy

$$\frac{1}{k_{\text{last}}} = 1 - \sum_{\ell=1}^{\text{last}-1} \frac{1}{k_\ell}$$

  - Stop at iteration j if $\sum_{\ell=1}^{j} \frac{1}{k_\ell} > 1$
  - Let k_j satisfy

$$\frac{1}{k_j} = 1 - \sum_{\ell=1}^{j-1} \frac{1}{k_\ell}.$$

If convergence is achieved, then the final iteration should not influence the results. There are however scenarios were one might expect the constrained $\sum_{\ell=1}^{j} \frac{1}{k_\ell} = 1$ to be violated, e.g., if the objective function $J_{\xi_i,\eta_i}(x)$ has a long valley with a minimum far from the current position, then having a value of λ_j close to 0 (k_j close to one) for many iterations would improve the result.

In practice, one does not have to resample and recompute weights after each update in the IAGS. One may simply run each iteration equivalent to EnRML and then, once convergence has been achieved, one may resample, recompute weights, and run again with a small h to get as close to importance sampling as possible.

Remark 5 For multimodal posterior distributions, only the IAGS can capture several modes in theory since it uses importance weights; however, in practice, they will all have problems since each method uses the global covariance as a gradient approximation, and in multimodal systems, this may be close to 0 [39].

Based on the above discussion, it is natural to believe that for large scale models, there is little that separates the results obtained with the three aforementioned methods for large scale models. For lower dimensional problems, however, we show in the next section that including importance weights may improve upon the result based solely on linear updates.

4.1 Comparison on a simple reservoir model

In this section, we study a 1D reservoir model [15] also studied in [8]. The model consists of 31 gridblocks (see Fig. 1), and the model parameters are gridblock log-permeabilities. The prior distribution is Gaussian with mean $\mu_p = 5$ and covariance matrix, C_p, that has ones on the diagonal and the off diagonal is computed with an exponential covariance function with practical range corresponding to 10 grid cells. We use the same ten initial ensembles, each of size 100, drawn independently from the prior, as in [15].

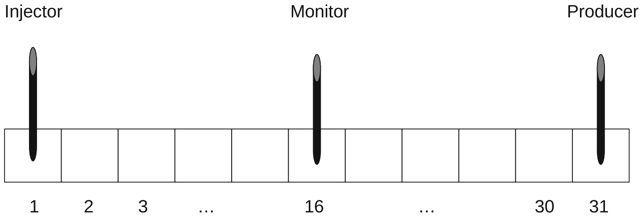


Fig. 1 1D model with 31 grid cells

There is a water injection well, operated at a constant bottom hole pressure of 4000 psi, in the first gridblock. In the last gridblock, there is a producer operated at a constant bottomhole pressure of 3000 psi. The measurements are given by the gridblock pressure at gridblock 16, denoted $\mathcal{G}(x)$, every 30 days over a period of 360 days. During this period, there is a water breakthrough at the monitor, but not at the production well. A natural result is that there is less information in the data about the permeability between the monitor and producer than between the injector and the monitor. The measurement noise is a 0 mean Gaussian random variable with variance of 1 psi, which is extremely small compared to the reservoir pressure.

We follow [8] and use a different reservoir simulator than [15]. The measurements are therefore regenerated to avoid bias from the numerical model. However, we compare the results to the MCMC results in [15] and the EnRML and ES-MDA reported in [8]. Since the measurements are almost perfect, the difference in the measurement noise will not change the results of the estimation significantly. To confirm this, the IAGS was run several times, each time with a different realization of the measurement error. The difference in the results were not visible for the permeability estimates, and the difference in the median of the normalized objective function was in the second or third decimal.

In [8], it was reported an average of 15 iterations until convergence; however, the ES-MDA were run with ten iteration, so the EnRML results were also reported after ten iterations. We therefore implement the IAGS with both ten and 15 iterations. The h value is initially set to $h = 0.05$ and then increased by 0.05 at each iteration until iteration 15 (or 10) where h is set to 0.1. The selection of h is ad hoc and needs further investigation.

In Table 1, we show the median of the normalized objective function, $J(x)/n_y$, where $n_y = 30$ is the number of measurements and

$$J(x) = (x-m)^T C_p^{-1}(x-m) + (y-\mathcal{G}(x))^T(y-\mathcal{G}(x)).$$

We see that neither the ES-MDA with ten (equally weighted) iterations [8] nor ES-MDA with ten different scaling factors [15] can compete with the EnRML or the

Table 1 Results from normalized objective function

Method	Median of $J(x)/N_d$
ES-MDA 10 (Emerick)	10.6
ES-MDA 10 (Chen)	10.7
EnRML 10 (Chen)	4.7
EnRML 15 (Chen)	2.5
IAGS 10	3.1
IAGS 15	1.8
MCMC (Emerick)	1.8

IAGS with ten (or 15) iterations. It is not discussed in [8] if the improved results of EnRML compared to ES-MDA is due to the selection of the scaling factor or due to the measurement error with increased noise that is added at each iteration. Based on the discussion in the previous section, it is reasonable to believe that it is possible to obtain results with ES-MDA that are much closer to the results of EnRML than the ones presented here.

As seen in Table 1, the median of the objective function is closer to MCMC for IAGS than the other two methods for both ten and 15 iterations. We believe this is due to the importance weights of the IAGS. In Fig. 2, the true log permeability is plotted along with the 0.02, 0.25, 0.5, 0.75, and 0.98 quantiles from the posterior ensemble obtained from IAGS and MCMC. We see that IAGS gives results not too different from the MCMC results; the absolute difference is shown in Fig. 3, with a lot fewer samples and severely less computation time (20 million model runs for the MCMC versus 10 times 1500 or 1000 for IAGS). The median of the IAGM is always within 10 % of the MCMC median. There are of course some differences especially in the region beyond the pressure observation where there is very little sensitivity of permeability to the observations. In this region, the 50 % marginal confidence intervals from the MCMC covers the true permeability field more often than the corresponding confidence interval for IAGS. We also note that there is only one point (grid cell 30) where the 96 % marginal confidence interval of IAGS fails to cover the truth.

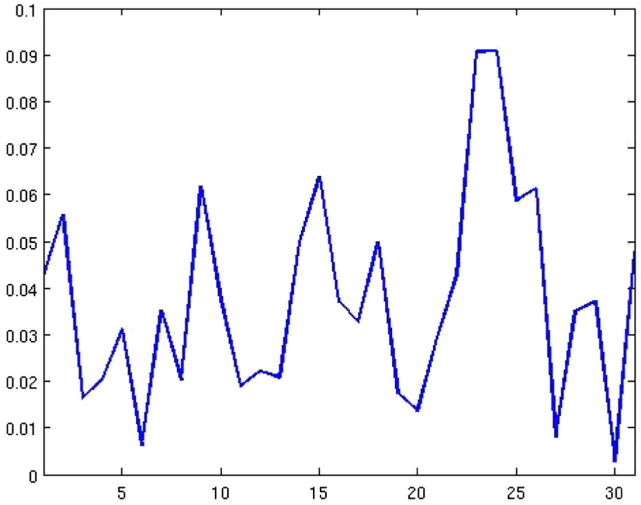
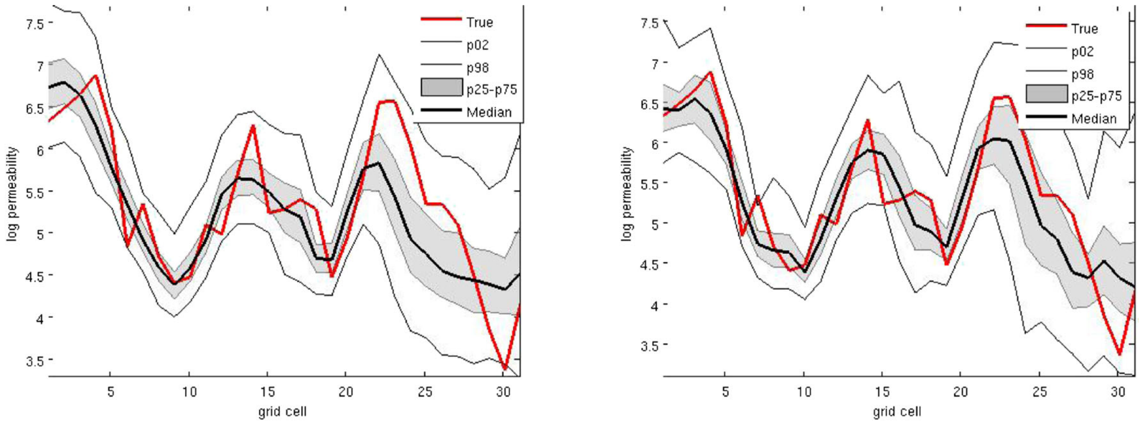


Fig. 3 Absolute difference in logPerm median from IAGS and MCMC

We also implemented an importance sampler with 15,000 particles. Due to the size of the problem and the extremely small measurement error, the largest of the 15,000 importance weights was 1 as expected. As a curiosity, the normalized objective function of this particle was 200.61. The same happened after each iteration with an adaptive approach, with the normalized objective function decreasing insignificantly. We believe that this demonstrates the potential of IAGS over adaptive importance sampling in high dimension.



(a) IAGS

(b) MCMC

Fig. 2 Estimated logPerm from IAGS and MCMC

5 Summary and Conclusions

In this paper, we have discussed the practical similarities and theoretical differences between different iterative Bayesian approximation methods which are applicable in large scale models. We first introduced the iterative adaptive Gaussian mixture smoother (IAGS) as an approximation of adaptive importance sampling in Bayesian parameter estimation.

We showed the similarities between IAGS and ensemble smoother with multiple data assimilation (ES-MDA) and Levenberg-Marquardt formulation of the ensemble-randomized maximum likelihood (EnRML) and tested IAGS on a synthetic 1D problem with 31 unknown parameters and extremely small measurement error. With 15 iterations, the results from IAGS was very similar to the ones obtained from an MCMC run. By construction, we saw that IAGS can be implemented as EnRML and ES-MDA, but with additional importance weights. We therefore argued that IAGS behaves similarly to EnRML, but with potential improvement if the reduced importance weights contain information. A significant improvement is expected for medium sized problems.

It is natural to believe that for the IAGS with a fixed number of total simulations, the best results are obtained with N and α increasing for each iteration. In addition, comparing the algorithm performance with N samples and j iterations instead of Nj samples is of interest for future investigation.

Acknowledgments The author would like to thank Hans A. Karlsen (University of Bergen) for the very helpful suggestions and comments and Geir Nævdal(IRIS) and Remus Hanea (Statoil) for the valuable discussions. The author also acknowledge industrial funding through the Petromaks project "Integrated work flow and realistic geology" and Schlumberger for providing scientific licenses for the Eclipse reservoir simulator.

Appendix Proof of Lemmas and Theorem 1

Proof of Lemma 1

Proof Since $g(x)$ is the likelihood function given $Y = y$ and since the law of X is μ, we have that a version of $\pi(f)$ is given by

$$\pi(f) = \frac{\int \mu(x)g(x)f(x)\,dx}{\int \mu(x)g(x)\,dx} = \frac{\mu(gf)}{\mu(g)}. \tag{A.1}$$

We start by stating simple and a well-known result on Martingale inequalities (see, e.g., [29]). Suppose that $\eta_1, \ldots, \eta_N$ are random variables which are independent and centered conditioned on a σ-field $\mathcal{H}$. Then, if $\mathbf{E}\left[|\eta_i|^2|\mathcal{H}\right] \leq A$, we have

$$\mathbb{E}\left|N^{-1}\sum_{i=1}^{N}\eta_i\right|^2 \leq \frac{A}{N}. \tag{A.2}$$

The same inequality holds if we define $\eta_i = \nu_i - \mathbb{E}\left[\nu_i|\mathcal{H}\right]$ and $\mathbf{E}\left[|\nu_i|^2|\mathcal{H}\right] \leq A$.

In order to use the above inequality, we define the following measure

$$\begin{aligned} \pi_j^N(x) &\overset{\text{def}}{=} \alpha_j \frac{\widehat{\Psi}_{j-1}^N * K_h g_j(x)}{\widehat{\Psi}_{j-1}^N * K_h(g_j)} + (1-\alpha_j)\widehat{\Psi}_{j-1}^N * K_h(x) \\ \pi_0^N(x) &\overset{\text{def}}{=} \mu(x) \end{aligned} \tag{A.3}$$

where $g_j(x)$ is defined in Eq. 3.2. Note that π_j^N is N dependent, but it is not an empirical measure, i.e., it cannot be computed. The introduction of these measures is simply to help us study the convergence of the empirical measures of interest. It is also worth noticing that

$$\frac{\widehat{\Psi}_{j-1}^N * K_h(g_j f)}{\widehat{\Psi}_{j-1}^N * K_h(g_j)} = \pi(f) \tag{A.4}$$

by definition.

Given a sample, $\{X_j^i\}_{i=1}^N$, from $\widehat{\Psi}_{j-1} * K_h$, we have

$$\eta_j^N(g_j f) = N^{-1}\sum_{i=1}^{N} f(X_j^i)g_j(X_j^i).$$

Then, our empirical estimate of $\pi(f)$ is

$$\Psi_j^N(f) = \alpha_j \frac{\eta_j^N(g_j f)}{\eta_j^N(g_j)} + (1-\alpha_j)\eta_j^N(f).$$

From (A.2) with the sigma field, $\mathcal{H}_{j-1}$, generated by $\{\widehat{X}_{j-1}^i\}_{i=1}^N$, we have

$$\begin{aligned} &\mathbf{E}\left|\eta_j^N g_j(f) - \widehat{\Psi}_{j-1}^N * K_h(g_j f)\right|^2 \leq \frac{\|f\|^2(C_1^j)^2}{N}, \\ &\mathbf{E}\left|\eta_j^N(f) - \widehat{\Psi}_{j-1}^N * K_h(f)\right|^2 \leq \frac{\|f\|^2}{N}. \end{aligned} \tag{A.5}$$

The inequalities in (A.5) simply states the typical $N^{-1/2}$ convergence in i.i.d. Monte Carlo sampling and relates the ensemble of predictions of the IAGS algorithm at iteration j with the importance function they are sampled from.

Using the triangle inequality, we have

$$\begin{aligned} &\left|\Psi_j^N(f) - \pi_j^N(f)\right| \leq \alpha_j \left|\frac{\eta_j^N(g_j f)}{\eta_j^N(g_j)} - \frac{\widehat{\Psi}_{j-1}^N * K_h(g_j f)}{\widehat{\Psi}_{j-1}^N * K_h(g_j)}\right| \\ &+(1-\alpha_j)\left|\eta_j^N(f) - \widehat{\Psi}_{j-1}^N * K_h(f)\right|, \end{aligned} \tag{A.6}$$

which states that the error in the IAGS algorithm can be split in two since the empirical measure can be studied as α_j times a weighted sample plus $1 - \alpha_j$ times an unweighted sample. We also have for all N and j: $\widehat{\Psi}_{j-1}^N * K_h(g_j) = C$

(it is the normalizing constant of π) by construction. Hence, for the first term of (A.6) we have, since $\alpha_j \leq 1$,

$$\alpha_j \left| \frac{\eta_j^N(g_j f)}{\eta_j^N(g_j)} - \frac{\widehat{\Psi}_{j-1}^N * K_h(g_j f)}{\widehat{\Psi}_{j-1}^N * K_h(g_j)} \right| \leq \left| \frac{\eta_j^N(g_j f)}{\eta_j^N(g_j)} - \frac{\widehat{\Psi}_{j-1}^N * K_h(g_j f)}{\widehat{\Psi}_{j-1}^N * K_h(g_j)} \right|$$
$$= C^{-1} \left| \frac{\eta_j^N(g_j f)}{\eta_j^N(g_j)} \left(\widehat{\Psi}_{j-1}^N * K_h(g_j) - \eta_j^N(g_j) \right) \right.$$
$$\left. + \left(\eta_j^N(g_j f) - \widehat{\Psi}_{j-1}^N * K_h(g_j f) \right) \right| .$$

Obviously, we have $\eta_j^N(g_j f) \leq \|f\| \eta_j^N(g_j)$.

Next, we define $L_1 f = \eta_j^N(g_j f) - \widehat{\Psi}_{j-1}^N * K_h(g_j f)$ we get by using the triangle inequality

$$\alpha_j \left| \frac{\eta_j^N(g_j f)}{\eta_j^N(g_j)} - \frac{\widehat{\Psi}_{j-1}^N * K_h(g_j f)}{\widehat{\Psi}_{j-1}^N * K_h(g_j)} \right| \tag{A.7}$$
$$\leq C^{-1} \left(\|f\| |L_1 1| + |L_1 f| \right) .$$

Note that (A.7) simply describes the expected difference between a normalized weighted sample and its theoretical counterpart. The second term of (A.6) is simpler; if we define $L_2 f = \eta_j^N(f) - \widehat{\Psi}_{j-1}^N * K_h(f)$, we get

$$(1 - \alpha_j) |\eta_j^N(f) - \widehat{\Psi}_{j-1}^N * K_h(f)| \leq |D^2 f| . \tag{A.8}$$

Then using (A.5), we see that

$$\mathbf{E}|L^1 f| \leq \frac{\|f\| C_1^j}{\sqrt{N}}, \quad \mathbf{E}|L^1 1| \leq \frac{C_1^j}{\sqrt{N}}, \quad \mathbf{E}|L^2 f| \leq \frac{\|f\|}{\sqrt{N}}, \tag{A.9}$$

hence,

$$\mathbf{E}|\Psi_j^N(f) - \pi_j^N(f)| \leq (2C^{-1} C_1^j + 1) \frac{\|f\|}{\sqrt{N}} . \tag{A.10}$$

This gives us the sampling error at iteration j as a function of N. However, we are interested in the error between (at first) $\Psi_j^N(f)$ and $\pi(f)$ since we will study the error between $\widehat{\Psi}_j^N(f)$ and $\Psi_j^N(f)$ later. Thus, if we can find an upper bound for the error between $\pi_j^N(f)$ and $\pi(f)$, we can use the triangle inequality to bound the quantity of interest.

We use the following decomposition combined with (A.4)

$$\begin{aligned} \pi_j^N(f) - \pi(f) &= \alpha_j \frac{\widehat{\Psi}_{j-1}^N * K_h(g_j f)}{\widehat{\Psi}_{j-1}^N * K_h(g_j)} \\ &\quad + (1 - \alpha_j) \widehat{\Psi}_{j-1}^N * K_h(f) - \pi(f) \\ &= (1 - \alpha_j) \left(\widehat{\Psi}_{j-1}^N * K_h(f) - \pi(f) \right) \\ &= \beta_j \left(\widehat{\Psi}_{j-1}^N * K_h(f) - \pi(f) \right) , \end{aligned}$$

to obtain

$$\mathbf{E} \left| \pi_j^N(f) - \pi(f) \right| = \mathbf{E} \beta_j \left| \widehat{\Psi}_{j-1}^N * K_h(f) - \pi(f) \right| .$$

Since K_h is a bounded symmetric kernel with finite second-order moments and since f is twice differentiable with bounded partial derivatives, we have for small h (using a Taylor expansion; see, e.g., [36])

$$\left| \widehat{\Psi}_{j-1}^N * K_h(f) - \widehat{\Psi}_{j-1}^N(f) \right| \leq h^2 C_f \tag{A.11}$$

for a constant C_f depending on f where we have used (A.1) and (A.3). It then follows that

$$\mathbf{E} \left| \pi_j^N(f) - \pi(f) \right| \leq \beta_j \left(h^2 C_f + \mathbf{E} \left| \widehat{\Psi}_{j-1}^N(f) - \pi(f) \right| \right) . \tag{A.12}$$

We then get by using the triangle inequality and combining (A.10), (A.11), and (A.12)

$$\mathbf{E} \left| \Psi_j^N(f) - \pi(f) \right| \leq \frac{\|f\|}{\sqrt{N}} C_1^j (2C^{-1} + 1) \tag{A.13}$$
$$+ \beta_j \left(\mathbf{E} \left| \widehat{\Psi}_{j-1}^N(f) - \pi(f) \right| + h^2 C_f \right) .$$

□

Proof of Lemma 2

Proof Next, we look at what happens when we include the linear integration step and approximative weights at each iteration. By splitting g_j into $g_j^1 g_j^2$ where

$$\begin{aligned} g_j^1 &= \frac{\mu}{\widehat{\Psi}_{j-1}^N * K_h}, \\ g_j^2 &= g, \end{aligned} \tag{A.14}$$

the approximation of $\pi(f)$ from the IAGS after each iteration can be written as

$$\widehat{\Psi}_j^N(f) = \alpha_j \frac{\widehat{\eta}_j^N(\widehat{g}_j f)}{\widehat{\eta}_j^N(\widehat{g}_j)} + (1 - \alpha_j) \widehat{\eta}_j^N(f), \tag{A.15}$$

where from Eq. 2.10

$$\widehat{\eta}_j^N(f) \overset{\text{def}}{=} N^{-1} \sum_{i=1}^{N} f(X_j^i + h^2 \delta_j^i),$$

$$\delta_j^i = \Sigma_j^{X, \mathcal{G}(X)} (h^2 \Sigma_j^{\mathcal{G}(x)} + R)^{-1} \left(y - \mathcal{G}(X_j^i) \right) , \tag{A.16}$$

$$\widehat{g}_j(X_j^i) = g_j^1 (g_j^2 * \tau_h)(X_j^i), \tag{A.17}$$

where τ_h is a symmetric-bounded kernel with finite second-order moments and $*$ denotes the convolution operator. With the Gaussian likelihood function described in Section 3 and with $\tau_h(u) = \phi(u; 0, h^2 H \Sigma^z H^T)$ (a 0 mean Gaussian density with covariance matrix $h^2 H P H^T$), we have

$$\widehat{g}_j(X_j^i) = g_j^1(X_j^i) \phi(y | H Z_j^i, h^2 H \Sigma_j^z H^T + R) \tag{A.18}$$

since $g(x) = \phi(y | H z, R)$ in this case. This is just another way of writing the update equations of the Gaussian mixture (2.10). Note that the theory presented here does not require τ_h to be Gaussian; however, (2.10) is a direct result of a Gaussian measurement error and it would be less rigorous to apply the same update for a non Gaussian likelihood function.

For small h, a Taylor expansion of f around X_j^i gives

$$f(X_j^i+h^2\delta_j^i)=f(X_j^i)+h^2\delta_j^i D^{(1)}f(X_j^i)+h^4(\delta_j^i)^2R_2(X_j^i),$$

where R_2 is the reminder term in the Taylor expansion and $D^{(1)}f$ denotes the first-order partial derivatives of f. Since the second-order derivatives of f is assumed to be uniformly bounded, we get

$$\left|f(X_j^i+h^2\delta_j^i)-f(X_j^i)\right|\leq h^2|\delta_j^i|C_1^f+h^4|\delta_j^i|^2C_2^f \tag{A.19}$$

for some constants C_1^f and C_2^f depending on f.

Similarly, for small h, we have

$$\begin{aligned} g*\tau_h &= \int g(u)h^{-1}\tau(h^{-1}(u-x))\,du \\ &= \int g(x+hz)\tau(z)\,dz = g(x)+h^2\int z^2R_2(x)\tau(z)\,dz \end{aligned}$$

where we have used the substitution $z=h^{-1}(x-u)$ and symmetry of τ. Together with Eq. 3.1, we get

$$\left|\widehat{g}_j(x)-g_j(x)\right|=\left|g_j^1[(g*\tau_h)(x)-g(x)]\right|=\leq h^2C^{g,j}, \tag{A.20}$$

for some constant $C^{g,j}$ depending on g and C_2^j. From (A.19) and (A.20), we see that

$$\begin{aligned}\mathbf{E}\left|\widehat{\eta}_j^N(\widehat{g}_jf)-\eta_j^N(g_jf)\right| &\leq \mathbf{E}\left|\widehat{\eta}_j^N(\widehat{g}_jf)-\widehat{\eta}_j^N(g_jf)\right| \\ &+\mathbf{E}|\widehat{\eta}_j^N(g_jf)-\eta_j^N(g_jf)\,|\leq h^2C^{g,f,j} \end{aligned}\tag{A.21}$$

for some constant $C^{g,f,j}$ depending on f and g and where we have used that $\mathbf{E}\|\delta_j^i\|<\infty$ by Eq. 3.1 and (A.17). In the same way, we deduce that

$$\mathbf{E}\left|\widehat{\eta}_j^N(\widehat{g}_j)-\eta_j^N(g_j)\right|\leq h^2C^{g,j}, \tag{A.22}$$

and

$$\mathbf{E}\left|\widehat{\eta}_j^N(f)-\eta_j^N(f)\right|\leq h^2C_f, \tag{A.23}$$

for sufficiently small h.

Again, we use the triangle inequality to get

$$\begin{aligned}\mathbf{E}\left|\widehat{\Psi}_j^N(f)-\Psi_j^N(f)\right| &\leq \mathbf{E}\,\alpha_j\left|\frac{\widehat{\eta}_j^N(\widehat{g}_jf)}{\widehat{\eta}_j^N(\widehat{g}_j)}-\frac{\eta_j^N(g_jf)}{\eta_j^N(g_j)}\right| \\ &+\mathbf{E}\,\beta_j\left|\widehat{\eta}_j^N(f)-\eta_j^N(f)\right|. \end{aligned}\tag{A.24}$$

For the first part, we use that $\eta_j^N(g_j)>\delta_j$ from Eq. 3.1 and the inequalities in (A.21) and (A.22) to get

$$\begin{aligned} &\mathbf{E}\alpha_j\left|\frac{\widehat{\eta}_j^N(\widehat{g}_jf)}{\widehat{\eta}_j^N(\widehat{g}_j)}-\frac{\eta_j^N(g_jf)}{\eta_j^N(g_j)}\right| \\ &=\mathbf{E}\alpha_j\left|\frac{1}{\eta_j^N(g_j)}\left(\frac{\widehat{\eta}_j^N(\widehat{g}_jf)}{\widehat{\eta}_j^N(\widehat{g}_j)}(\eta_j^N(g_j)-\widehat{\eta}_j^N(\widehat{g}_j))+(\widehat{\eta}_j^N(\widehat{g}_jf)-\eta_j^N(g_jf))\right)\right| \\ &\leq \|f\|\frac{h^2}{\delta_j}C^{f,g,j}, \end{aligned}$$

for some constant $C^{f,g,j}$ using (A.21), (A.22) and 3.1.

From (A.23), we have

$$\mathbf{E}\,\beta_j\left|\widehat{\eta}_j^N(f)-\eta_j^N(f)\right|\leq\beta_jh^2C^f,$$

which gives us

$$\mathbf{E}\left|\widehat{\Psi}_j^N(f)-\Psi_j^N(f)\right|\leq h^2\|f\|F_j \tag{A.25}$$

where $F_j=\beta_jC^f+C^{f,g,j}\delta_j^{-1}\|f\|^{-1}$. □

Proof of Theorem 1

Proof Combining Lemma 1 and Lemma 2 with the triangle inequality, we see that

$$\begin{aligned}\mathbf{E}\left|\widehat{\Psi}_j^N(f)-\pi(f)\right| &\leq \mathbf{E}\left|\widehat{\Psi}_j^N(f)-\Psi_j^N(f)\right|+\mathbf{E}\left|\Psi_j^N(f)-\pi(f)\right| \\ &\leq h^2\|f\|D_j+\tfrac{\|f\|}{\sqrt{N}}C_1^j(2C^{-1}+1) \\ &\quad+\beta_j\,\mathbf{E}\left|\widehat{\Psi}_{j-1}^N(f)-\pi(f)\right|, \end{aligned}\tag{26}$$

where $D_j=F_j+h^2\beta_jC^f\|f\|^{-1}$.

Letting $h=N^{-\frac{1}{4}}$, we may rewrite the above as

$$\mathbf{E}\left|\widehat{\Psi}_j^N(f)-\pi(f)\right|\leq\frac{\|f\|}{\sqrt{N}}B_j+\beta_j\,\mathbf{E}\left|\widehat{\Psi}_{j-1}^N(f)-\pi(f)\right|,$$

for sufficiently large N, where $B_j=C_1^j(2C^{-1}+1)+D_j$

Next, we define $B=sup\,B_k$, $k=1,\ldots J$, where J is the maximum number of iterations. Then, by iterating backwards on j, we finally obtain

$$\begin{aligned}\mathbf{E}\left|\widehat{\Psi}_j^N(f)-\pi(f)\right| &\leq \tfrac{\|f\|B}{\sqrt{N}}\left(1+\textstyle\sum_{\ell=1}^{j-1}\prod_{k=\ell+1}^{j}\beta_k\right) \\ &+|\mu(f)-\pi(f)|\left(\textstyle\prod_{k=1}^{j}\beta_k\right). \end{aligned}$$

□

References

1. Andrieu, C., Doucet, A., Holenstein, R.: Particle Markov chain Monte Carlo methods. J. R. Stat. Soc.: Ser. B (Stat. Methodol.) **72**(3), 269–342 (2010)
2. Bengtsson, T., Bickel, P., Li, B.: Curse-of-dimensionality revisited: collapse of particle filter in very large scale systems. Probab. Stat. **2**, 316–334 (2008)
3. Beskos, A., Crisan, D., Jasra, A.: On the stability of sequential Monte Carlo methods in high dimensions arXiv preprint (2011). arXiv:1103.3965
4. Burgers, G., van Leeuwen, P., Evensen, G.: Analysis scheme in the ensemble Kalman filter. Mon. Weather Rev. **126**(6), 1719–1724 (1998)
5. Cappé, O., Guillin, A., Marin, J.-M., Robert, C.P.: Population Monte Carlo. J. Comput. Graph. Stat. **13**(4) (2004)
6. Chen, R., Liu, J.S.: Mixture Kalman filters. J. R. Stat. Soc. Ser. B-Stat. Methodol. **60**, 493–508 (2000)

7. Chen, Y., Oliver, D.S.: Ensemble randomized maximum likelihood method as an iterative ensemble smoother. Math. Geosci. **44**(1), 1–26 (2012)
8. Chen, Y., Oliver, D.S.: Levenberg-Marquardt forms of the iterative ensemble smoother for efficient history matching and uncertainty quantification. Comput. Geosci., 1–15 (2013)
9. Christie, M., Blunt, M.: Tenth SPE comparative solution project: a comparison of upscaling techniques. SPE Reserv. Eval. Eng. **4**(4), 308–317 (2001)
10. Cotter, C.J., Reich, S.: Ensemble filter techniques for intermittent data assimilation-a survey arXiv preprint (2012). arXiv:1208.6572
11. Cotter, S., Roberts, G., Stuart, A., White, D.: MCMC methods for functions modifying old algorithms to make them faster. Statistical Science (2012)
12. Del Moral, P., Doucet, A., Jasra, A.: An adaptive sequential Monte Carlo method for approximate Bayesian computation. Stat. Comput. **22**(5), 1009–1020 (2012)
13. Douc, R., Guillin, A., Marin, J.-M., Robert, C.P.: Convergence of adaptive mixtures of importance sampling schemes. Ann. Stat. **35**(1), 420–448 (2007)
14. Doucet, A., Del Moral, P., Jasra, A.: Sequential Monte Carlo samplers. J. R. Stat. Soc. B **68**(3), 411–436 (2006)
15. Emerick, A.: History matching and uncertainty characterization using ensemble-based methods. University of Tulsa, PhD thesis (2012)
16. Emerick, A., Reynolds, A.: Ensemble smoother with multiple data assimilation.Computers & Geosciences (Available online 17 March (2012)
17. Evensen, G.: Sampling strategies and square root analysis schemes for the EnKF. Ocean Dyn. **54**(6), 539–560 (2004)
18. G: Evensen. Data Assimilation: the ensemble Kalman filter. Springer (2007)
19. Fossum, K., Mannseth, T., Oliver, D., Skaug, H.: Numerical comparison of ensemble Kalman filter and randomized maximum likelihood. In ECMOR XIII-13th European Conference on the Mathematics of Oil Recovery (2012)
20. Frei, M., Künsch, H.R.: Mixture ensemble Kalman filters. Computational statistics and data analysis (2011)
21. Givens, G., Raftery, A.: Local adaptive importance sampling for multivariate densities with strong nonlinear relationships. J. Am. Stat. Assoc. **91**(433), 132–141 (1996)
22. Gray, A.G., Moore, A.W.: N-Body'problems in statistical learning. Adv. neural Inf. Process. Syst., 521–527 (2001)
23. Hoteit, I., Pham, D.T., Triantafyllou, G., Korres, G.: A new approximate solution of the optimal nonlinear filter for data assimilation in meteorology and oceanography. Mon. Weather Rev. **136**(1), 317–334. doi:10.1175/2007MWR1927.1. JAN 2008. ISSN 0027-0644
24. Ionides, E.L.: Truncated importance sampling, vol. 17 (2008)
25. Kitanidis, P.K.: Quasi-linear geostatistical theory for inversing. Water Resour. Res. **31**(10), 2411–2419 (1995)
26. Kong, A., Liu, J., Wong, W.: Sequential imputations and Bayesian missing data problems. J. Am. Stat. Assoc. **89**(425), 278–288 (1994)
27. Kotecha, J.H., Djurić, P.M.: Gaussian sum particle filtering. IEEE **51**(10), 2602–2612 (October 2003)
28. Le Gland, F., Monbet, V., Tran, V.-D., et al.: Large sample asymptotics for the ensemble Kalman filter. Oxf. Handb. Nonlinear Filtering, 598–631 (2011)
29. Ledoux, M., Talagrand, M.: Probability in Banach Spaces, Isoperimetry and Processes.23. Springer (1991)
30. Neal, R.M.: Annealed importance sampling. Stat. Comput. **11**(2), 125–139 (2001)
31. Oh, M.-S., Berger, J.O.: Adaptive importance sampling in monte carlo integration. J. Stat. Comput. Simul. **41**(3-4), 143–168 (1992)
32. Oliver, D.S., Chen, Y.: Recent progress on reservoir history matching: a review. Comput. Geosci. **15**(1), 185–221 (2011)
33. Oliver, D.S., Reynolds, A.C., Liu, N.: Inverse theory for petroleum reservoir characterization and history matching.Cambridge (2008)
34. Ristic, B., Arulampalam, S., Gordon, N.: Beyond the Kalman filter, pages 45–47.Artech House (2004)
35. Sakov, P., Oke, P.R.: Implications of the form of the ensemble transformation in the ensemble square root filters. Mon. Wea. Rev. **136**, 1042–1053 (2008)
36. Silverman, B.W.: Density estimation for statistics and data analysis.Chapman and Hall (1986)
37. Stordal, A., Lorentzen, R.: An iterative version of the adaptive Gaussian mixture filter. Comput. Geosci. (2014). doi:10.1007/s10596-014-9402-6
38. Stordal, A., Karlsen, H., Nævdal, G., Skaug, H., Vallès, B.: Bridging the ensemble Kalman filter and particle filters. Comput. Geosci. **15**(2), 293–305 (2011)
39. Stordal, A., Karlsen, H., Nævdal, G., Oliver, D., Skaug, H.: Filtering with state space localized Kalman gain. Phys. D **241**(13), 1123–1135 (2012)
40. Stuart, A.M.: Inverse problems: a Bayesian perspective. Acta Numerica **19**(1), 451–559 (2010)
41. van Leeuwen, P.J., Evensen, G.: Data assimilation and inverse methods in terms of a probabalistic formulation. Mon. Weather Rev. **124**, 2898–2913 (1996)