Bayesian Calibration of Hydrocarbon Reservoir Models Using an Approximate Reservoir Simulator in the Prior Specification

Ole Petter Lødøen and Håkon Tjelmeland

Department of Mathematical Sciences,
Norwegian University of Science and Technology,
Trondheim, Norway

Abstract

We consider prediction and uncertainty analysis for the 'history matching' problem in petroleum reservoir evaluation. The unknown reservoir properties are represented on a fine three dimensional lattice. A 'reservoir simulator', solving a set of partial differential equations, takes the reservoir properties as input and gives the production properties as output. The history matching problem is to infer the reservoir properties from an observed production history. To run the reservoir simulator on the lattice size of interest is a computer intensive task, and this severely limits the number of runs that can be made.

We formulate the problem in a Bayesian setting and, following previous suggestions in the statistical literature, consider the reservoir simulator as an unknown function. We propose a new and more realistic prior formulation for this function, combining a (faster) version of the reservoir simulator on a coarse lattice with parameters correcting for the bias introduced by the coarser lattice. We simulate from the resulting posterior by Markov chain Monte Carlo (MCMC). We present a case study inspired by the Troll field in the North Sea. Convergence and mixing properties of the MCMC algorithm are good. The case study demonstrates how the observed production history provide information both about the reservoir properties and about the bias correcting parameters used in the prior specification.

Keywords: Approximate reservoir simulation; Bayesian statistics; Markov chain Monte Carlo; Parameter estimation; Production conditioning

1 Introduction

In the petroleum industry fluid flow simulations are regularly used to learn about potential future production from a reservoir. This is usually done via fluid flow

simulations on a set of potential scenarios for the reservoir properties. The simulation results are used to study uncertainties, and ultimately are basis for decisions concerning production strategies.

The fluid flow model is given as a set of partial differential equations. These are much too complex to be solved analytically and numerical solutions are found using 'reservoir simulators'. This is complex computer programs that take as input properties of the petroleum reservoir and the operating conditions, including spatial distribution of permeability and porosity and location of wells. The output is a description of the resulting production, including time series of pressure and produced volumes of oil, water and gas in each well.

The reservoir properties that are needed as input to the reservoir simulator are in practice largely unknown. In contrast, after the reservoir has been in operation for some time, the first part of the production time series are observed. It is therefore of great interest to use the observed production properties to obtain information about the input reservoir properties. In practice this implies that one needs to find simulator input values that produce simulator output values consistent with the observed production. In the petroleum industry this is called 'history matching' and is typically done via an iterative and partly manual procedure. As one run of the fluid flow simulator may take several hours or even days to run on a computer, this is a very time demanding process. One is therefore often satisfied when one has obtained one set of history matched input values. The input values obtained from this process is of some interest in itself, but the main objective is to get better predictions of future production properties for a variety of (future) operating conditions. This can then be used to tune the operating conditions to optimise future production from the reservoir.

History matching is one example of a wider class of problems involving complex computer models. The last years several articles have appeared in the statistical literature that formulate this in a Bayesian setting, both in a general form and for history matching in particular. In contrast to the standard procedures used in the petroleum industry, focus is here also on quantifying the uncertainty of the reservoir variables and the corresponding predictions. A thorough general discussion can be found in Kennedy and O'Hagan (2001), see also Oakley and O'Hagan (2004). Craig et al. (1996, 1997, 2001) consider the history matching problem, but limit the attention to Bayes linear models. A key point in all these articles is that they consider the reservoir simulator as an unknown function, to which a prior distribution is assigned. The reservoir simulator is a deterministic function of the input variables and it is known in the sense that it can be run for any set of input values. In practice, however, the computer resources required by each run severely limits the number of runs that can be made. It is therefore reasonable to regard the simulator as an unknown function.

In this paper we adopt a simplified version of the general framework of Kennedy

and O'Hagan (2001). A difficult part in this approach is the specification of the prior distribution for the unknown reservoir simulator function. To specify the prior we adopt ideas from Omre and Lødøen (2004). The prior is a very complex function going from a high dimensional input space to another high dimensional output space. We consider it unreasonable to be able to describe anything like this with, for example, a simple Gaussian random function. After all, the complexity of this function is the reason why the complex computer code was implemented in the first place. However, a reservoir simulator can be run for different grid sizes and we propose to use a coarse grid simulator as part of the prior specification. It is well known that the reservoir simulator output is biased when run on a coarse grid and this is also incorporated into our prior specification. The coarse scale simulator then becomes a part of the resulting posterior distribution. Thus, the only viable alternative for evaluation of the posterior properties is via Markov chain Monte Carlo (MCMC) simulation.

The paper is organised as follows. In Section 2 our Bayesian model is specified and in Section 3 we use this formulation in a case study. Section 3.1 defines the case study set-up, in Section 3.2 we define the exact prior and likelihood models for the problem, in Section 3.3 we discuss the choice of MCMC algorithm and in Section 3.4 we present and discuss the simulation results. Finally, Section 4 provides conclusions.

2 Model formulation

In this section we formulate the history matching problem in a Bayesian setting. We first formulate a quite general model. A more detailed definition of the model components is given in Section 3, where we also apply the model to a case study.

2.1 Notation

Several reservoir properties are important for fluid flow. Important examples are porosity, permeability and initial fluid saturations. These quantities vary spatially in the reservoir and is commonly represented on a three dimensional lattice. Let $x \in \Re^n$ denote a vector with these properties in each node. Thus, n equals the number of lattice nodes times the number of reservoir properties considered. Let $\omega(x)$ denote the reservoir simulator. In addition to being a function of x, the simulation output also depends on the operating conditions, like well positioning and the production strategy used. However, as the operating conditions are known we suppress this dependency in our notation. The reservoir simulator output is really a number of time series, representing pressures in the wells, production rates of oil, water and gas, and potentially injection rate of water or gas. We limit ourselves to observing these quantities at a number of discrete times, so that $y = \omega(x) \in \Re^m$, where m

is the number of production variables times the number of time points used. The vector $y = \omega(x)$ is naturally divided into two parts, components related to the past and components related to future production. Let $D^p \in \Re^{m_p \times m}$ and $D^f \in \Re^{m_f \times m}$, where $m_p + m_f = m$, be matrices that picks out from y quantities related to the past and the future, respectively, and write $y^p = D^p y$ and $y^f = D^f y$. In particular, we let y_0^p denote the vector of observed quantities from the reservoir under study and let y^f denote the corresponding future production that we want to predict.

Adopting the strategy in Kennedy and O'Hagan (2001), we consider $\omega(\cdot)$ as an unknown function, for which a prior should be defined. We run the simulator for K values $x_1, \ldots, x_K$ and let $y_k = (y_k^p, y_k^f) = \omega(x_k), k = 1, \ldots, K$ denote the corresponding simulation results. As $\omega(\cdot)$ is an unknown function, $y_1, \ldots, y_K$ are observed data and a corresponding likelihood function needs to be defined. An important design of experiment problem is the choice of the input values $x_1, \ldots, x_K$. We will not discuss this problem here, and so consider $x_1, \ldots, x_K$ as given and fixed. However, see the discussion in Omre and Lødøen (2004). In the case discussed in Section 3 we generate the x_k's simply by sampling independently k times from the prior for x.

As discussed above we also use a coarse grid version of the simulator, which we denote by $\tilde{\omega}(x)$. The operator $\tilde{\omega}(\cdot)$ really consists of two parts. First the fine scale representation of the reservoir properties in x is upscaled to a corresponding coarse scale representation, and thereafter the reservoir simulator is run on the coarse scale. For porosity and initial fluid saturations the upscaling process is just to take arithmetic averages, whereas for permeability the upscaling is a highly non-linear operation. For reviews of various types of upscaling see Farmer (2000) and Durlofsky (2003). In the following we assume the coarse grid to be sufficiently small so that $\tilde{\omega}(x)$ can be evaluated reasonably fast. In our case study discussed in Section 3 evaluations of $\omega(x)$ and $\tilde{\omega}(x)$ take about one hour and twenty seconds, respectively. One should note that the same computer code, with the same operating conditions, is run for both $\omega(\cdot)$ and $\tilde{\omega}(\cdot)$. Thus, the number of and the types of output variables are the same, so $\tilde{\omega}(x) \in \Re^m$.

2.2 Stochastic model

We specify a prior distribution for the reservoir properties x, denoted $\pi(x)$. Further, we specify a prior for the unknown function $\omega(x)$ by setting

$$\omega(x) = A\tilde{\omega}(x) + b + \varepsilon(x), \tag{1}$$

where $A \in \Re^{m \times m}$ is a diagonal matrix, $b \in \Re^m$ and $\varepsilon(x) \in \Re^m$ is a multivariate stochastic process defined for $x \in \Re^n$. The matrix A and the vector b are introduced to model the bias in coarse scale simulator runs, and can be thought of as slopes and intercepts in a multivariate linear regression setting. Independent prior distributions are specified for A and b. We write $A \sim \pi(A)$ and $b \sim \pi(b)$.

The specification of the residual process $\varepsilon(x)$ is a challenging task. We discuss details of this for the case study in Section 3. At the general level discussed here, just note that the residual process is a multivariate spatial time series ensuring that y_0^p and y_k, $k = 1, \ldots, K$, are exactly reproduced. We adopt a hierarchical specification: Given a hyper-parameter vector θ we assume $\varepsilon(x)$ to be a zero mean Gaussian process. To θ we assign a prior distribution $\pi(\theta)$.

The available data is y_0^p and $y_1, \ldots, y_K$. We will consider the fine scale reservoir simulator $\omega(\cdot)$ to be a perfect model for the fluid flow process in the reservoir, and we assume the observations to be without observation error. Thus, if x is the (unknown) true reservoir properties, we have $y_0^p = D^p\omega(x)$. Clearly, as discussed in Kennedy and O'Hagan (2001), these two assumptions are unrealistic. However, we consider this to be a reasonable first pass. When filling in all the details, as we do in Section 3, our current model is already quite complex. If including also modelling and observation error terms, identifiability issues must also be carefully considered.

The likelihood function for the available data becomes

$$\left[\begin{array}{c} y_0^p \\ y_1 \\ \vdots \\ y_K \end{array}\right] \Bigg| x, A, b, \theta \sim \mathrm{N}\left(\left[\begin{array}{c} D^p(A\tilde{\omega}(x) + b) \\ A\tilde{\omega}(x_1) + b \\ \vdots \\ A\tilde{\omega}(x_K) + b \end{array}\right], \Sigma_\theta(x, x_1, \ldots, x_K) \right), \tag{2}$$

where the covariance matrix $\Sigma_\theta(x, x_1, \ldots, x_K)$ follows from the covariance structure assumed for $\varepsilon(x)|\theta$.

The above fully specifies the Bayesian model. The posterior distribution of interest is $\pi(x, y^f | y_0^p, y_1, \ldots, y_K)$. This distribution is clearly analytically intractable, both because it is given by a high dimensional integral and because it includes the coarse grid simulator $\tilde{\omega}(\cdot)$. The first problem can be solved by considering $\pi(x, y^f, A, b, \theta | y_0^p, y_1, \ldots, y_K)$ instead. The standard solution to the latter is to explore the distribution via a Metropolis–Hastings algorithm (Metropolis et al., 1953; Hastings, 1970).

2.3 Posterior simulation

Our task is to evaluate the properties of $\pi(x, y^f, A, b, \theta | y_0^p, y_1, \ldots, y_K)$. This distribution is naturally split into two parts,

$$\pi(x, A, b, \theta | y_0^p, y_1, \ldots, y_K) \quad \text{and} \quad \pi(y^f | x, A, b, \theta, y_0^p, y_1, \ldots, y_K). \tag{3}$$

The latter is a Gaussian distribution and is thereby straightforward to evaluate. The only viable alternative for the first is to use the Metropolis–Hastings algorithm. Nice introductions to MCMC and the Metropolis–Hastings algorithm can be found in Dellaportas and Roberts (2003) and in Gamerman (1997). As the distribution function for $\pi(x, A, b, \theta | y_0^p, y_1, \ldots, y_K)$ includes the coarse scale simulator, the function $\tilde{\omega}(\cdot)$ must be evaluated in each Metropolis–Hastings iteration that updates x.

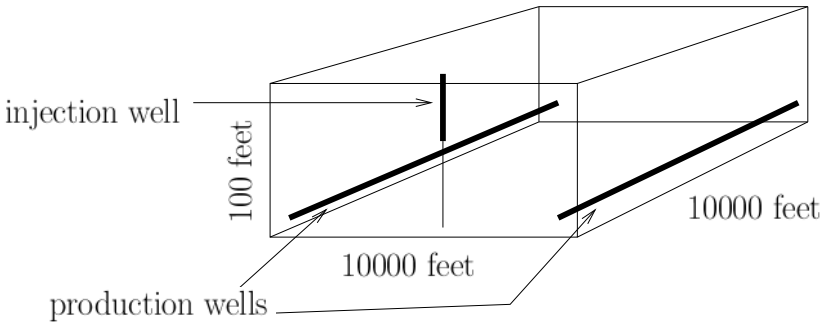

Figure 1: Outline of the reservoir. The thick lines indicate where the wells are perforated.

Even if the evaluation of $\tilde{\omega}(\cdot)$ is very fast compared to the evaluation of the fine scale simulator $\omega(\cdot)$, this still puts severe restrictions on the number of iterations we can run in a Metropolis–Hastings algorithm. It is thereby essential to use an algorithm with good convergence and mixing properties. For the case study discussed in Section 3 we use an algorithm where each of x, (A, b) and θ are updated separately using Metropolis updates. Again we give more details in the next section.

3 Case study

In this section we give a more detailed description of the model outlined in Section 2, related to a specific case study. This case study was first introduced in Hegstad and Omre (2001), and later used in Omre and Lødøen (2004). The reservoir under study covers a domain of size $10^4 \times 10^4 \times 10^2$ feet^3, and is discretised onto a lattice of size $50 \times 50 \times 15$. Hence, the lattice has 37,500 nodes. The reservoir is initially fully saturated with oil, and it has one vertical injection well and two horizontal production wells, as shown in Figure 1. The injection well is perforated in the upper five layers of the reservoir, which means that these are the only layers where we have injection of fluids into the reservoir, in this case gas. The production wells are perforated along the entire well trace, and they produce both oil and gas. The wells operate under a set of constraints, which we discuss below.

Hegstad and Omre (2001) define a Bayesian model to evaluate the production forecast conditioned on static data (seismic data and well observations) and dynamic data (observed production history) through a brute-force rejection sampling algorithm. They also use $\tilde{\omega}(\cdot)$ as an approximation to $\omega(\cdot)$ without considering the bias thereby introduced. Omre and Lødøen (2004) evaluate the production forecast conditioned on the same data, but also account for the biases and changed error structures introduced by the coarse scale reservoir simulator $\tilde{\omega}(\cdot)$. They use the observations $y_k = \omega(x_k)$, $k = 1, \ldots, K$, together with the corresponding coarse scale simulation results $\tilde{y}_k = \tilde{\omega}(x_k)$, $k = 1, \ldots, K$, to estimate the coefficients (A, b) and

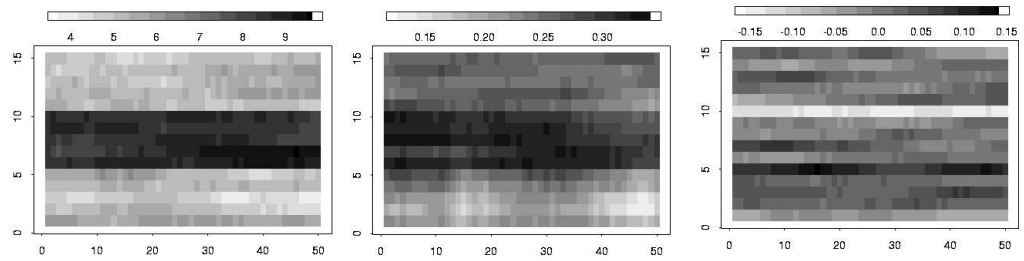

Figure 2: Vertical cross sections of the true reservoir properties, x^{true}. From left to right; log-permeability, porosity and seismic reflection coefficient.

the covariance matrix in (1) through linear regression, independently at each point in time. Hence, they do not consider correlation in time. In the history matching, each pair (x, y) is given weight according to a likelihood function involving the observed production in an importance sampling algorithm. In the current work, we estimate the parameters (A, b) and the covariance matrix in (1) through the Metropolis-Hastings algorithm, and sample history matched realisations (x, y) from the posterior. We include parameters to model correlation both between reservoirs and in time.

3.1 Problem setting

As described above, the reservoir covers a domain of size $10^4 \times 10^4 \times 10^2$ feet^3, with spatially varying reservoir properties represented on a three dimensional lattice of size $50 \times 50 \times 15$. A reference reservoir x^{true} containing the true values, in each node on the lattice, for seismic reflection coefficients, porosities and permeabilities, is constructed inspired by the Troll Field in the North Sea offshore Norway. Hence, $n = 112,500$. Figure 2 shows a vertical cross section of the true reservoir properties. The reservoir consists of three distinct layers, where the middle layer has higher permeability and porosity values than the other two. Each reservoir variable in each layer is constructed from a realisation of a Gaussian random field where the mean and covariance structure is chosen to get the layered structure seen in Figure 2. For the seismic reflection coefficient, high negative and positive values are added to the interfaces between the layers to mimic seismic reflectors. This makes the reflection coefficient non-Gaussian. Details on the construction of the reference reservoir is given in Hegstad and Omre (2001).

The coarse grid representation of the reservoir is defined on a lattice of size $10 \times 10 \times 15$. The mapping from the fine scale to the coarse scale is defined by taking averages of regions of size $5 \times 5 \times 1$ on the fine scale lattice. Porosities and saturations are mapped by simple arithmetic averages, while permeabilities are mapped by harmonic averages. Permeability is the most difficult property to upscale, and a variety of algorithms with varying degrees of accuracy exists, see Farmer (2000) and Durlofsky (2003). Since all algorithms inevitably lead to some bias in the predictions,

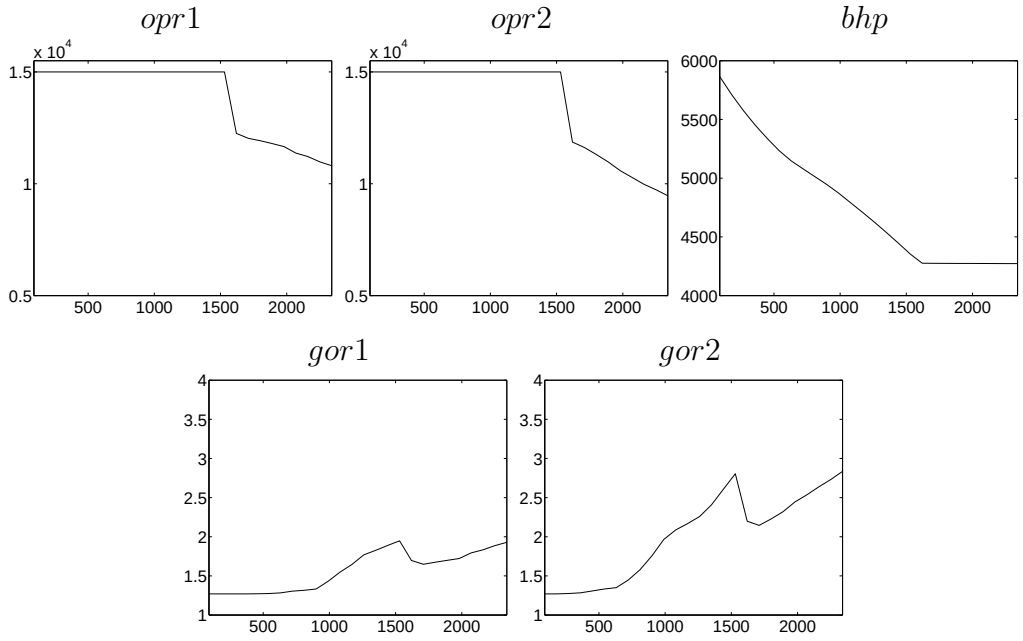

Figure 3: The production y^{true} from the reference reservoir as function of days.

we choose to apply one of the simplest available.

The reservoir simulator requires a number of input parameters to be defined. We assume that properties such as mobility ratios, rock compressibilities and relative permeability curves are known. These are of course important parameters for fluid flow calculations, but not the scope of our study. Further, the fact that the reservoir is initially fully saturated with oil gives the initial fluid saturations.

In addition to the reservoir properties, the well positions and the operating conditions need to be given as input. The vertical injection well operates at a constant rate of gas injection, which happens at a rate of 65,000 mscf/day (1 mscf $=$ $1,000$ feet^3 at 60° F and 14.7 psi). The two production wells initially produce at a constant rate of oil production, which happens at a rate of 15,000 stb/day (1 stb $\approx$ 159 l) in each well. In each of the production wells, this rate of production is maintained until the pressure drops below 4,100 psi, then this well switch to production at a constant pressure of 4,100 psi.

When the input variables and the operating conditions are fully defined, the reservoir simulator can be run to give time series for selected flow responses as output at required times. In this work, we use the Eclipse 300 version 2004a reservoir simulator, see GeoQuest (2004). We choose to evaluate the pressure in the injection well (*bhp*), the gas/oil ratios in the two production wells (*gor*1 and *gor*2), and the oil production rates in the two production wells (*opr*1 and *opr*2), all at a quarterly basis for six and a half years, hence $m = 130$. Figure 3 shows the selected flow responses $y^{true} = \omega(x^{true})$ from the reference reservoir under the given operating conditions.

Notice that when the injected gas starts to reach the production well we get a sharp increase in the gas/oil ratio. This can been seen, both for $gor1$ and $gor2$, in Figure 3 at approximately 800 days. Further, notice the sharp decline in the oil production rates when the production wells switch from constant rate control to constant pressure control. This happens at approximately 1,500 days in both production wells. The bottom hole pressure in the injection well does not reach a constant level, even if it appears so in Figure 3, since it is the pressure difference between the injection well and the production wells that drives the production.

We find these sudden changes hard to model by a Gaussian function, and they also lead to numerical instabilities, so we choose to re-parametrise the flow response time series. Let $y = \omega(x)$ be the flow response as before. Now y contains five separate time series, one for each of the flow responses considered. For the four flow responses that initially stay at a constant level, we define δ_i, $i \in \{opr1, opr2, gor1, gor2\}$, to be the number of times/quarters it takes before the flow response leaves the constant level. This allows us to define a vector $\Delta = [\delta_{opr1} \;\; \delta_{opr2} \;\; \delta_{gor1} \;\; \delta_{gor2}]^T$. Further, we define $z = [z_{opr1} \;\; z_{opr2} \;\; z_{gor1} \;\; z_{gor2} \;\; z_{bhp}]^T$ to be the remainder of the production after the different flow responses leave the constant level. Let $D(\cdot)$ be an operator that picks out the time until the different flow responses leave a constant level, and also returns the remainder of the flow responses after leaving a constant level. Hence, $(\Delta, z) = D(y) = (D^{\Delta}(y), D^z(y))$. Note that z have varying dimension.

3.2 Model formulation

What we refer to as the prior model $\pi(x)$ for x is in fact the posterior model for the reservoir properties conditioned on static data (seismic data and well observations). Since our interest is mainly the conditioning on production data, we leave the conditioning on the static data out of the notation. The prior model for the reservoir properties is defined hierarchically, where the prior for porosity and permeability depend on the seismic reflection coefficient. A complete definition of the prior model is given in Hegstad and Omre (2001). The model parameters, including means, correlation lengths and standard deviations, are the same as the ones used by Omre and Lødøen (2004). The prior model is easy to sample from, in the sense that samples can be generated relatively fast.

As described in Section 2.2 we want to model the unknown function $\omega(\cdot)$ through a bias corrected coarse scale grid version $\tilde{\omega}(\cdot)$ of $\omega(\cdot)$. We assume independence between the re-parametrised variables, and assign prior distributions of the same form as (1),

$$\Delta = D^{\Delta}(\omega(x)) = A_{\Delta} D^{\Delta}(\tilde{\omega}(x)) + b_{\Delta} + \varepsilon_{\Delta}(x), \tag{4}$$

$$z = D^z(\omega(x)) = A_z D^z(\tilde{\omega}(x)) + b_z + \varepsilon_z(x), \tag{5}$$

where $A_{\Delta} = \mathrm{diag}(a_{\Delta})$ and $A_z = \mathrm{diag}(a_z)$. The vectors $a_{\Delta} \in \Re^4$ and $b_{\Delta} \in \Re^4$ both have one element related to each of the flow responses that initially stay at

a constant level. The dimension of the vectors a_z and b_z is varying in different realisations. The parts of a_z and b_z relating to each flow response are assumed to have constant values. Therefore, a_z and b_z are both fully determined by only five values, one for each flow response.

Since the vectors a_Δ and b_Δ can be seen as the slope and intercept in a linear regression setting, natural choices for prior distributions are

$$a_\Delta \sim \mathrm{N}(\mathbb{1}, \Sigma_{a_\Delta}) \quad \text{and} \quad b_\Delta \sim \mathrm{N}(0, \Sigma_{b_\Delta}), \tag{6}$$

with independence entailing $\Sigma_{a_\Delta} = \mathrm{diag}(\sigma^2_{a_\Delta})$ and $\Sigma_{b_\Delta} = \mathrm{diag}(\sigma^2_{b_\Delta})$. Similar arguments yields

$$a_z \sim \mathrm{N}(\mathbb{1}, \Sigma_{a_z}) \quad \text{and} \quad b_z \sim \mathrm{N}(0, \Sigma_{b_z}) \tag{7}$$

as priors for a_z and b_z. Because of assumed independence between production variables and correlation in time, which is discussed below, Σ_{a_z} and Σ_{b_z} are block diagonal matrices, where the blocks are defined by $\sigma^2_{a_z}$ and $\sigma^2_{b_z}$. The parameters $\sigma^2_{a_\Delta}$, $\sigma^2_{b_\Delta}$, $\sigma^2_{a_z}$ and $\sigma^2_{b_z}$ are chosen so that (6) and (7) are vague priors compared to the scale on which the corresponding flow responses vary. The chosen values are listen in Table 1.

parameter	$opr1$	$opr2$	$gor1$	$gor2$	bhp
$\sigma^2_{a_\Delta}$	2^2	2^2	2^2	2^2	—
$\sigma^2_{b_\Delta}$	40^2	40^2	40^2	40^2	—
$\sigma^2_{a_z}$	2^2	2^2	2^2	2^2	2^2
$\sigma^2_{b_z}$	10000^2	10000^2	20^2	20^2	5000^2

Table 1: The parameters defining the priors for a_Δ, b_Δ, a_z and b_z.

Specifying the covariance structures for the residual processes $\varepsilon_\Delta(x)$ and $\varepsilon_z(x)$ in (4) and (5) is more complicated. If we think of x as a "location" in the space of possible realisations of the reservoir properties, $\varepsilon_\Delta(x)$ is a multivariate spatial process, and $\varepsilon_z(x)$ is a multivariate spatio-temporal process. By assuming separability and independence between variables, the covariance structure of $\varepsilon_\Delta(x)$ can be expressed through a spatial correlation function and a multivariate covariance matrix $\Sigma_\Delta = \mathrm{diag}(\sigma^2_\Delta) \in \Re^{4\times 4}$. We assume that, given a hyper-parameter r_x, the correlation is only dependent on the distance between the two locations x and x'. We choose an exponential correlation function on the form

$$\rho_x(x, x'|r_x) = \exp\left\{-3\frac{||x - x'||}{r_x}\right\}, \tag{8}$$

where $||x - x'||$ is the Euclidean distance between x and x', and r_x is the effective correlation length. The above fully specifies the covariance structure of the residual

process $\varepsilon_\Delta(x)$ through the hyper-parameters r_x and σ_Δ^2. To both these parameters we must assign prior distributions $\pi(r_x)$ and $\pi(\sigma_\Delta)$, which we specify below.

The same separability assumption allows us to express the covariance structure of $\varepsilon_z(x)$ through the same spatial correlation function $\rho_x(\cdot,\cdot|\cdot)$ as above, a multivariate covariance matrix $\Sigma_z = \text{diag}(\sigma_z^2)$, and a temporal correlation function. We choose to use a temporal correlation function on the form

$$\rho_t(z(s), z(t)|r_t, \nu) = \exp\left\{-3\left(\frac{|t-s|}{r_t}\right)^\nu\right\}, \tag{9}$$

where $|t-s|$ is the distance in time between t and s, r_t is the effective correlation length, and $\nu \in [0,2]$. This fully specifies the covariance structure of the residual process $\varepsilon_z(x)$ through the hyper-parameters r_x, r_t, ν and σ_z.

By collecting all the hyper-parameters previously defined in this section in one hyper-parameter vector θ, we get

$$\theta = (\sigma_\Delta, \sigma_z, r_x, r_t, \nu). \tag{10}$$

The prior distribution $\pi(\theta)$ is defined as follows. For the standard deviations and correlation lengths we choose Gamma priors, while the parameter ν is a priori set to be uniformly distributed on the interval $[0,2]$. The parameters defining the Gamma priors, are listed in Table 2. These values are again chosen to give vague priors. This completes the definition of the stochastic model.

parameter	σ_Δ				σ_z					r_x	r_t
	*opr*1	*opr*2	*gor*1	*gor*2	*opr*1	*opr*2	*gor*1	*gor*2	*bhp*		
α	5	5	5	5	1.5	1.5	1.5	1.5	1.5	5	1.5
β	10	10	10	10	5000	5000	2.5	2.5	2500	100	100

Table 2: The parameters α and β defining the Gamma priors for the hyper-parameter vector θ.

3.3 Metropolis–Hastings algorithm

The Metropolis-Hastings algorithm simulates a reversible Markov chain with limiting distribution identical to our specified target distribution, $\pi(x, A, b, \theta|y_0^p, y_1, \ldots, y_K)$. First, new states for one of $x, (A,b)$ or θ are proposed, according to some proposal distribution. Then, the proposed states are accepted as new states for the Markov chain according to an acceptance probability, otherwise the old states are kept. These two steps combined are what we refer to as a Metropolis update.

Our algorithm works in the following way: First the initial states of the Markov chain are drawn from the prior distributions for $x, (A,b)$ and θ as defined in Section 3.2. Then new states for the Markov chain are found using Metropolis updates.

The Metropolis updates can be divided into two types; (i) updates of the reservoir properties x, and (ii) updates of the parameters in the likelihood function. Updates of type (i) are the most resource requiring, since they involve running the reservoir simulator $\tilde{\omega}(\cdot)$ to evaluate the acceptance probability. The updates of type (ii) are very fast. This makes the algorithm feasible for parallel computing. While one (or more) processor(s) are performing reservoir simulations to do updates of type (i), other processors can perform a (large) number of updates of type (ii). When a reservoir simulation is finished, the last accepted states of (A, b) and θ can be used together with the simulation result $\tilde{\omega}(x)$ from the reservoir simulation, to perform updates of type (i). We take advantage of this, and define one iteration in our algorithm to be one update of type (i) and 50 updates of type (ii). The simulations are performed using two processors, and the number of updates of type (ii) is chosen so that the idle time for either processor is minimal.

We have three types of reservoir properties; permeability, porosity and seismic reflection coefficients. In updates of type (i) we propose new values in all grid nodes for one of these properties. Letting ϕ_{old} denote a vector of the current values for the chosen reservoir property, we generate potential new values, ϕ_{new}, by sampling a vector ϕ from the prior and setting

$$\phi_{new} = \mu + \alpha(\phi - \mu) + \sqrt{1 - \alpha^2}(\phi_{old} - \mu), \tag{11}$$

where μ is the corresponding prior mean vector and $\alpha \in [0, 1]$. By choosing α close to zero, a small change is proposed, while $\alpha = 1$ gives independent proposals. We tuned the value of α to get an acceptance rate around 0.25. This resulted in $\alpha = 0.6$. Note that changing the reflection coefficients also changes the conditional means for porosity and permeability, because of the way $\pi(x)$ is modelled hierarchically. Changes in porosity and permeability only affects these variables.

The updates of type (ii) are performed by Gaussian random-walk proposals. This means proposing a new state from a Gaussian distribution centred at the current value. The slope and intercept in linear regression are negatively correlated. Experience have shown that the mixing and convergence properties of the algorithm are improved if we take advantage of this, so for each flow response, we propose a change in a_Δ and b_Δ or a_z and b_z simultaneously, with a correlation of -0.8. All the other parameters in the likelihood function are updated independently, one at a time. This means that we have a total of 21 different alternatives for updates of type (ii). In one iteration of the algorithm, each of the 50 updates of type (ii) are chosen independently at random among the 21 alternatives. The standard deviations in the Gaussian proposal distributions are tuned to give acceptance probabilities in the interval $[0.2, 0.4]$.

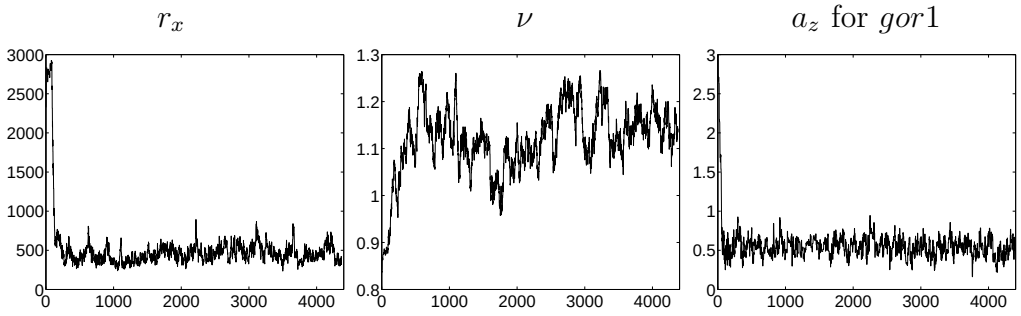

Figure 4: First scenario: Plots of r_t, r_x and ν as functions of iteration number.

3.4 Results

We consider prediction of future production and evaluation of the corresponding uncertainties for the reservoir simulation problem described in Section 3.1. We consider two different scenarios, where the only difference is the length of the observed production history. For the first scenario, we choose to use the production from the reference reservoir (y^{true}) up to 1,800 days as the observed production history y_0^p. For the second scenario, the observed production history is chosen to be y^{true} up to 1,350 days. This means that for the first scenario, both the gas-breakthrough times and the times where the production controls switch from constant rate to constant pressure are observed in both production wells. For the second scenario, only the gas-breakthrough times are observed. We sample $K = 10$ independent realisations from the prior $\pi(x)$ to get x_k, $k = 1, \ldots, 10$, and run the full fluid flow simulator $\omega(\cdot)$ to get $y_k = \omega(x_k)$, $k = 1, \ldots, 10$, considered to be observations from the function $\omega(\cdot)$. These observations are the same in both scenarios.

First scenario

For the first scenario, we run the Metropolis-Hastings algorithm described in Section 3.3 for 10,000 iterations to generate samples from $\pi(x, A, b, \theta | y_0^p, y_1, \ldots, y_K)$. Trace plots of the first 4,000 iterations for ν, r_x and a_z for $gor1$ are shown in Figure 4. Visual inspection suggests that the chain converges after about 500 iterations. Similar or better mixing and convergence are observed for the other parameters.

We use the simulated values, after convergence, to estimate the posterior distributions for the parameters x, (A, b) and θ. Figure 5 shows density estimates of the posterior for each of a_z, a_Δ, b_z, b_Δ, σ_z and σ_Δ for $opr1$ and $gor1$, in addition to the parameters r_x, r_t and ν in the correlation functions. It is clear from the figures that the coarse scale production properties are biased compared to the respective fine scale production properties. The estimated mode for a_z for $opr1$ is about 0.7, meaning that the simulated values on the coarse scale are too high. Further, the estimated mode for a_Δ for $gor1$ is about 0.6, meaning that gas-breakthrough in this well happens too late in the coarse scale simulation runs. Note that $a_z = 1$ for $opr1$

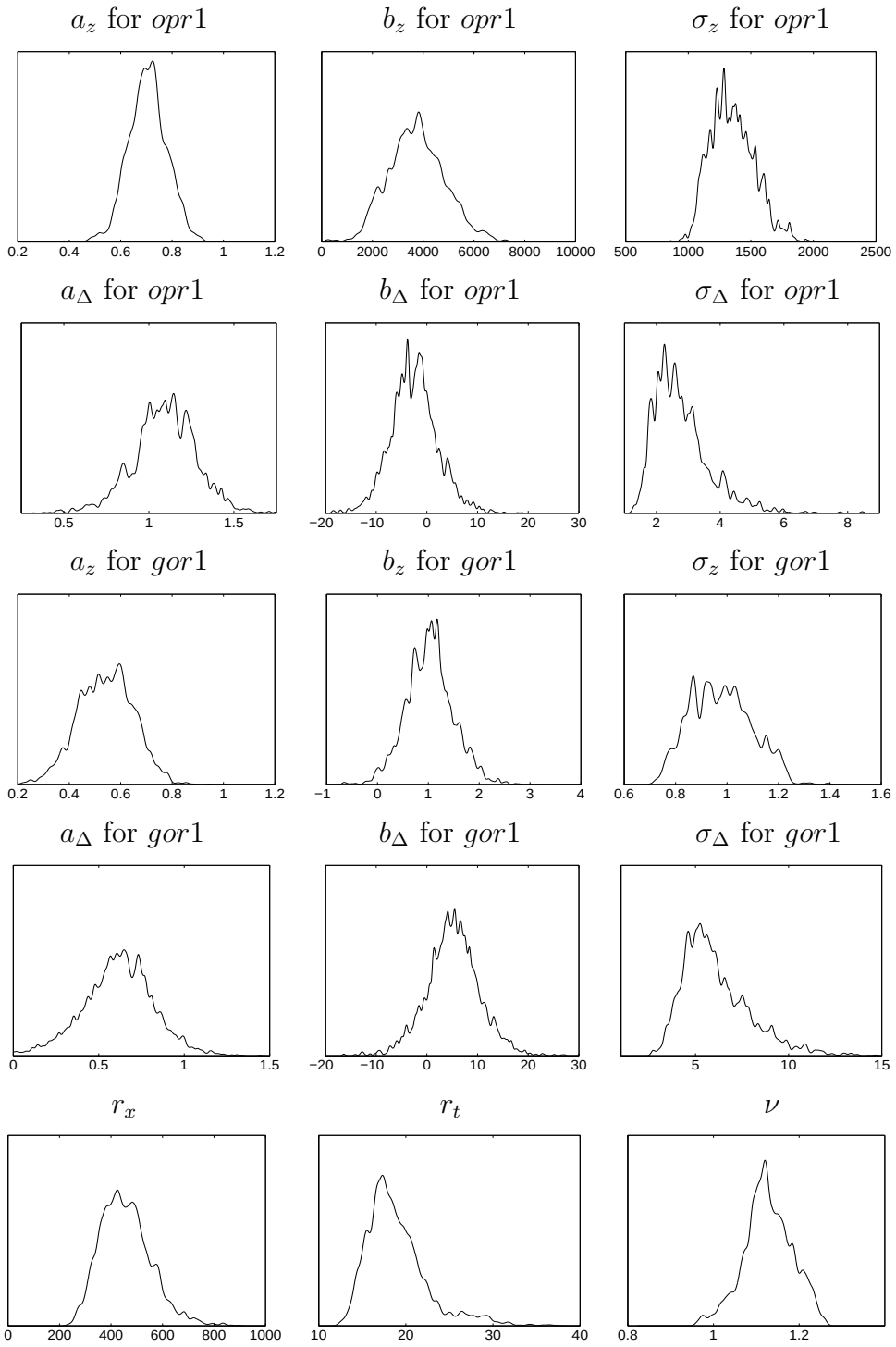

Figure 5: First scenario: Gaussian kernel density estimates for a selection of the parameters we sample in the Metropolis-Hastings algorithm.

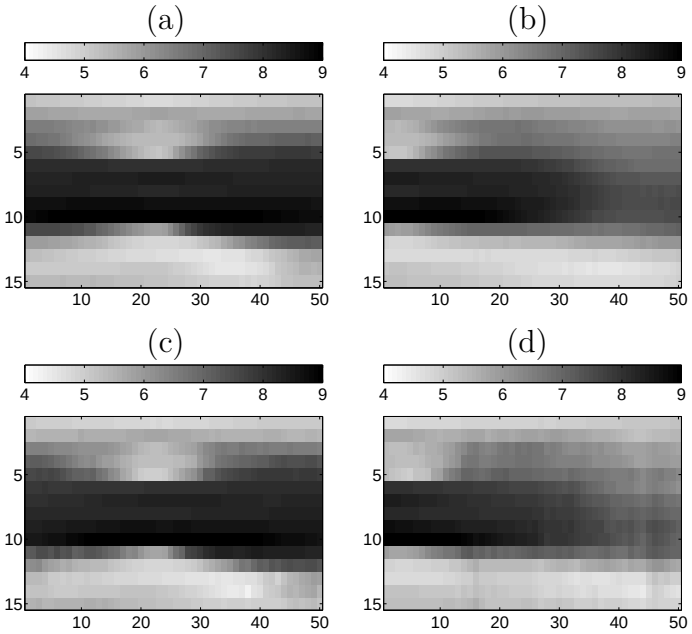

Figure 6: First scenario: Vertical slices of the estimated prior mean log-permeability, (a) and (b), and the corresponding vertical slices of the estimated posterior mean log-permeability, (c) and (d).

and $a_\Delta = 1$ for $gor1$ are highly unlikely values. This demonstrates that when we use $\tilde{\omega}(\cdot)$ as a prior for the unknown function $\omega(\cdot)$, we must, to make the prior realistic, also include parameters to correct for the bias introduced by the upscaling.

Figures 6(c) and (d) show the estimated mean posterior log-permeability for two vertical slices of the reservoir. For comparison, Figures 6(a) and (c) contain corresponding prior mean values. The first slice ((a) and (b)) is parallel to the production wells while the second slice ((c) and (d)) is orthogonal to the production wells. Both slices are located two grid blocks away from the injection well. In the areas close to the injection well, which is located in the leftmost area of (a) and (c), and in the middle area of (b) and (d), the estimate of the posterior mean is somewhat closer to the distinct layered structure in the reference reservoir. This is the region of the reservoir where the fluid saturations change the most during the time window we are considering, since the oil initially in place is replaced by the injected gas. Hence, it is also the region which has the largest impact on the resulting production properties. It is thereby natural that the observed production history is most informative about the reservoir properties in this part of the reservoir. Figure 7 shows estimates of the prior and posterior standard deviation for the log-permeability for the first vertical slice. The prior uncertainty is low near all the wells because we condition on well observations. The uncertainty is higher at the interfaces between the high and low permeable layers because of bimodality in the prior, due to the uncertainty in the location of the high permeable region. The

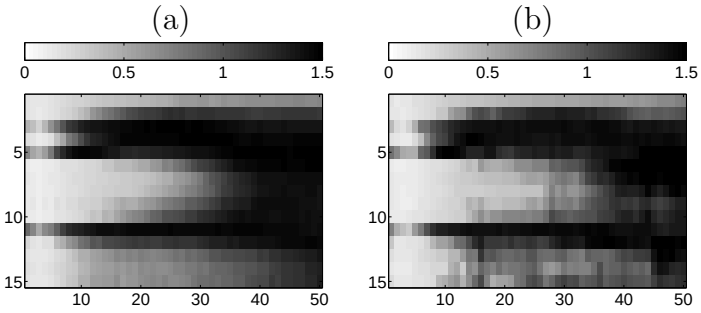

Figure 7: First scenario: Vertical slice of the estimated standard deviation for the prior log-permeability, (a), and the corresponding vertical slice of the estimated standard deviation for the posterior log-permeability, (b).

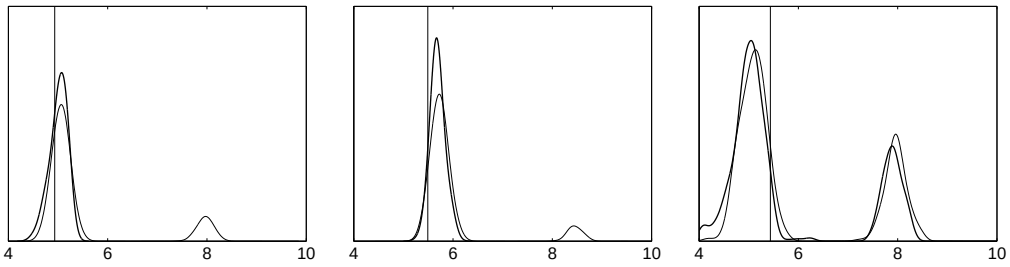

Figure 8: First scenario: Estimates of the prior log-permeability (thin lines) and the posterior log-permeability (thick lines) for three grid blocks in the reservoir. The vertical lines indicate the true value for the log-permeability.

posterior standard deviation is lower, especially in the area close to the injection well.

The more distinct layered structure in the posterior close to the injection well can also be seen by comparing the prior and posterior estimates for single grid blocks in this region. Figure 8 shows estimates of the prior and posterior log-permeability for single grid blocks, in the area right above and right below the high permeable middle layer, compared to the true value from the reference reservoir. The prior is bimodal, meaning that the location of the high permeable middle layer is uncertain. In the area close to the injection well (the two leftmost figures), the posterior estimate has only one mode. Further away from the wells (the figure to the right), the prior and posterior estimates look quite similar, meaning that the observed production history does not provide much information about the reservoir properties here.

The fact that the observed production history carry information about the reservoir properties can also be seen if we compare coarse scale simulation runs when sampling the reservoir properties from the prior and when sampling from the posterior. Figure 9 shows a comparison of the mean and 90% confidence interval for $\tilde{y} = \tilde{\omega}(x)$ when x is sampled from the prior and the posterior. The effect of the observed production history is clear.

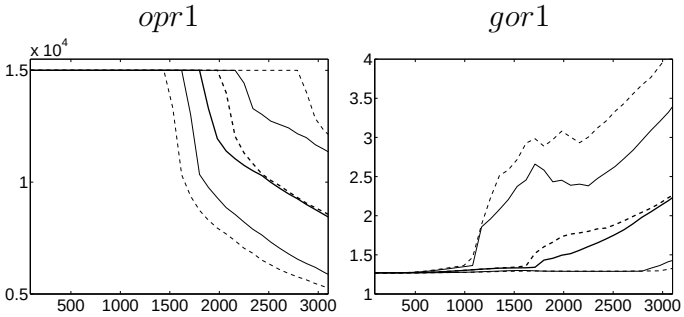

Figure 9: First scenario: Comparison of the mean and 90% confidence interval for $\tilde{y} = \tilde{\omega}(x)$ when x is drawn from the prior (dashed lines) and when x is drawn from the posterior (whole lines).

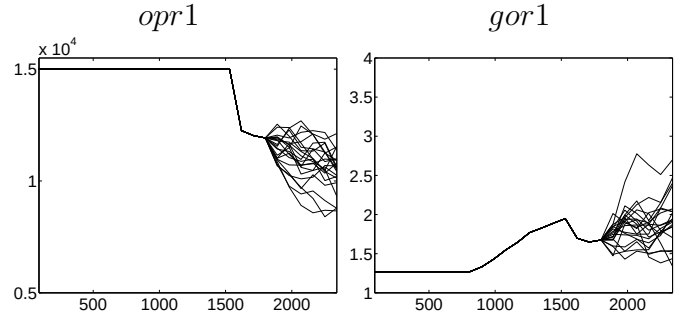

Figure 10: First scenario: Independent posterior realisations of $y = (y_0^p, y^f)$.

For each iteration of the Metropolis-Hastings algorithm we also predict the production until $2,340$ days, by sampling y^f from $\pi(y^f|x, A, b, \theta, y_0^p, y_1, \ldots, y_{10})$. Figure 10 shows 20 realisations of $y = (y_0^p, y^f)$. These realisations are each 200 iterations apart, so that they are essentially independent realisations from the posterior. Figure 11 shows the corresponding mean and 90% confidence interval, based on all the predicted y^f's compared to y^{true}. The predicted mean of the future production is close to the true production from the reference reservoir.

Second scenario

For the second scenario, which is conditioning on the shorter production history, we again run the Metropolis-Hastings algorithm for 10,000 iterations to generate samples from the posterior distribution $\pi(x, A, b, \theta|y_0^p, y_1, \ldots, y_K)$. The trace plots and density estimates of all the parameters in (A, b) and θ look very similar to those obtained when conditioning on the longer production history. The reason for this is that the main factors in the estimation of these parameters are the observations $y_k, k = 1, \ldots, 10$, which remain the same in both cases.

Figure 12 shows the estimated mean posterior log-permeability and the corresponding estimate of the posterior standard deviation for the vertical slice parallel

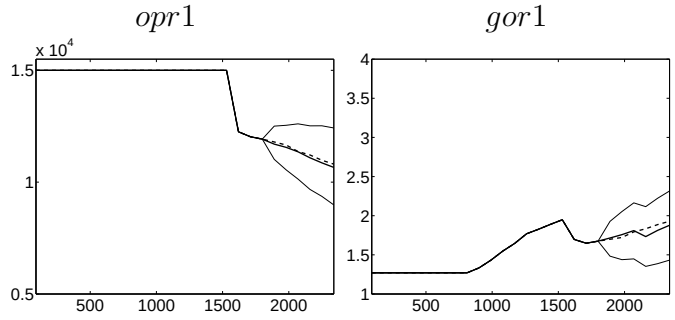

Figure 11: First scenario: The predicted mean (thick lines) and 90% confidence interval (thin lines) compared to the true production (dashed lines).

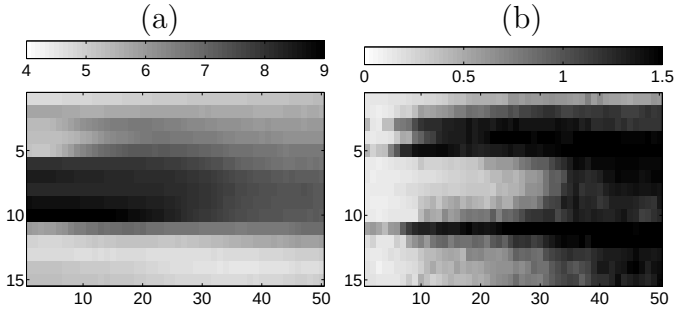

Figure 12: Second scenario: Vertical slice of the estimated standard deviation for the prior log-permeability, (a), and the corresponding vertical slice of the estimated standard deviation for the posterior log-permeability, (b).

to the production wells. The main difference between the prior (shown in Figures 6(b) and 7(a)) and posterior is seen in the estimate of the standard deviation. The posterior estimate is lower, which means that the location of the high permeable middle layer is more certain in the posterior. In the area near the injection well, the layered structure is not as distinct as when we condition on the longer production history (shown in Figure 6(b)). This can also be seen if we compare the prior and posterior estimates for single grid blocks in the area right above and right below the high-permeable middle layer, shown in Figure 13. Both the prior and posterior estimates are bimodal. However, for the grid blocks close to the injection well (the two leftmost figures) the probability mass in the mode to the right is less in the posterior. Further away from the injection well, the prior and posterior estimates are very similar. The grid blocks are the same as for the estimates shown in Figure 8, when conditioning on the longer production history, where the posterior estimates close to the injection well have only one mode.

For each iteration of the Metropolis-Hastings algorithm, we again predict the production until 2,340 days, by sampling y^f from $\pi(y^f|x, A, b, \theta, y_0^p, y_1, \ldots, y_K)$. Figure 14 shows the mean and 90% confidence interval based on all the predicted y^f's. We see that right after the production control switches, at approximately 1,600 days, the

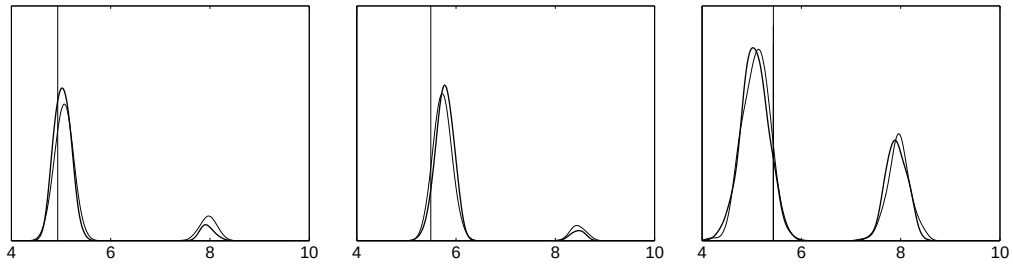

Figure 13: Second scenario: Estimates of the prior log-permeability (thin lines) and the posterior log-permeability (thick lines) for different grid blocks in the reservoir. The vertical lines indicate the true value for the log-permeability.

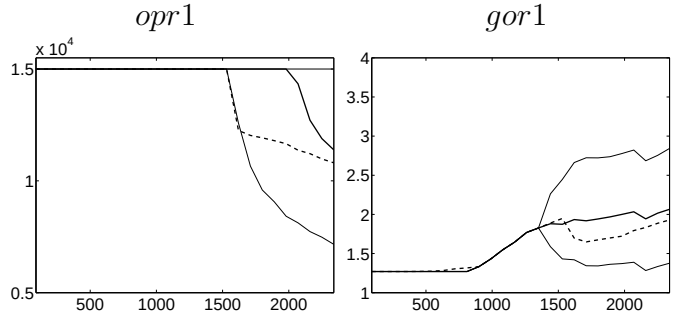

Figure 14: Second scenario: The predicted mean (thick lines) and 90% confidence interval (thin lines) compared to the true production (dashed lines).

true oil production rate is at the edge of the 90% confidence interval. One possible explanation for this is that the prior is too optimistic. A comparison between the observations $y_1, \ldots, y_K$, which come from the prior, and the true production shows that this may be the case here. For the gas/oil ratio, the predicted mean is close to the true production.

4 Closing remarks

Prediction and uncertainty analysis for the history matching problem in petroleum reservoir evaluation can only be assessed by sampling based on a fluid flow simulator. When sampling conditioned to an observed production history, evaluation of the acceptance probability in the Metropolis-Hastings algorithm involves running a fluid flow simulator. In most cases, the computational demands of the reservoir simulator prohibits repeated simulations, needed for an efficient MCMC algorithm.

In the present paper we adopt the Bayesian framework from Kennedy and O'Hagan (2001). Using ideas from Omre and Lødøen (2004) we define a new and more realistic prior formulation by including a coarse lattice reservoir simulator in the prior specification. We use a Metropolis-Hastings algorithm to generate samples from the resulting posterior.

We applied the model to two different scenarios in a case study inspired by the Troll field, and showed how the observed production history provides information both about the bias correction parameters in the prior, and the reservoir properties. The result is a better characterisation of the reservoir properties, and thereby better predictions of the future production. The changes inferred to the reservoir properties are mostly restricted to the areas of the reservoir where the fluid saturations change during the time covered by the observed production history.

The study has shown that there are some topics within the experimental design that should be further investigated. Among these topics is the choice of prior distribution. The second scenario shows some evidence that the true production is in the tail of the prior distribution, which means that the prior is too optimistic. Moreover, the selection of the input values $x_1, \ldots, x_K$ is a challenging task. Recall that this selection must be made prior to performing any fine scale fluid flow simulations. Ideally, these observations should span the posterior distribution, which is practically impossible to achieve. We can, however, base the selection on a set of coarse scale fluid flow simulations, as in Omre and Lødøen (2004).

Acknowledgements

The study is partly supported by the Uncertainty in Reservoir Evaluation (URE) program at the Norwegian University of Science and Technology.

References

Craig, P., Goldstein, M., Rougier, J. and Seheult, A. (2001). Bayesian forecasting for complex systems using computer simulators, *J. Am. Statist. Ass.* **96**: 717–729.

Craig, P., Goldstein, M., Seheult, A. and Smith, J. (1996). Bayes linear strategies for matching hydrocarbon reservoir history, *in* J. Barnardo, J. Berger, A. Dawid and A. Smith (eds), *Bayesian Statistics 5*, Oxford University Press, pp. 69–95.

Craig, P., Goldstein, M., Seheult, A. and Smith, J. (1997). Pressure matching for hydrocarbon reservoirs: A case study of Bayes linear strategies for large computer experiments, *in* C. Gatsonis, J. Hodges, R. Kass, R. McCulloch, P. Rossi and N. Singpurwalla (eds), *Case Studies in Bayesian Statistics, III*, Vol. 121 of *Lecture Notes in Statistics*, Springer, New York, pp. 37–93.

Dellaportas, P. and Roberts, G. O. (2003). An introduction to MCMC, *in* J. Møller (ed.), *Spatial Statistics and Computational Methods*, number 173 in *Lecture Notes in Statistics*, Springer, Berlin, pp. 1–41.

Durlofsky, L. (2003). Upscaling of geocellular models for reservoir flow simulation: A review of recent progress. Presented at 7th International Forum on Reservoir Simulation, Bühl/Baden-Baden, Germany, June 23-27, 2003.

Farmer, C. (2000). Upscaling: A review, *International Journal for Numerical Methods in Fluids* **40**: 63–78.

Gamerman, D. (1997). *Markov chain Monte Carlo: Stochastic simulation for Bayesian inference*, Chapman & Hall, London.

GeoQuest (2004). *Eclipse 2004a, reference manual*, GeoQuest Reservoir Technologies, Schlumberger.

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

Hegstad, B. and Omre, H. (2001). Uncertianty in production forecasts based on well observations, seismic data, and production history, *SPE Journal* (June 2001): 409–424.

Kennedy, M. and O'Hagan, A. (2001). Bayesian calibration of computer models (with discussion), *J. R. Statist. Soc. B* **63**: 425–464.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. and Teller, E. (1953). Equation of state calculations by fast computing machines, *Journal of Chemical Physics* **21**: 1087–1092.

Oakley, J. and O'Hagan, A. (2004). Probabilistic sensitivity analysis of complex models: a Bayesian approach, *J. R. Statist. Soc. B* **66**: 751–769.

Omre, H. and Lødøen, O. (2004). Improved production forecasts and history matching using approximate fluid-flow simulators, *SPE Journal* (September 2004): 339–351.