Math Geol (2007) 39: 383–398
DOI 10.1007/s11004-007-9102-z

Well Conditioning in Object Models

Ragnar Hauge · Lars Holden · Anne Randi Syversveen

Received: 19 December 2005 / Accepted: 31 January 2007 / Published online: 10 August 2007

Abstract This paper presents two object models with corresponding simulation algorithms, which aim to condition well data correctly while still converging in reasonable time. The first model is devoted to fluvial channels and the second one is mainly intended for smaller objects. To verify the conditioning, a method for validating well conditioning algorithms for object models is given. The purpose is to determine the extent to which the well conditioning introduces a bias in the models. To do this, we check that the double expectation of a parameter conditioned to wells is equal to the unconditional expectation. This method is applied to two different object models. Both the conditioning algorithms presented here give good results using this test.

Keywords Stochastic modelling · Object model · Well conditioning · Metropolis–Hastings

Introduction

Modelling of heterogeneous reservoirs is often done by including a facies model, and then adding different petrophysical properties for the different facies types. Different facies will typically have very different levels of porosity and permeability. The idea is that the facies model captures the dominating heterogeneity, so simple and smooth models can be used for the petrophysics within each facies. A common tool for modelling of facies is an object model where one facies is set as the background facies and objects of other facies are added on the top of this.

R. Hauge (✉) · L. Holden · A.R. Syversveen
Norwegian Computing Center, P.O. Box 114, 0314 Oslo, Norway
e-mail: ragnar.hauge@nr.no

L. Holden
e-mail: lars.holden@nr.no

A.R. Syversveen
e-mail: anne.randi.syversveen@nr.no

Traditionally, the main concern with conditioning of object models to well observations has been to get objects to match the observations. Correct conditioning of the well data is important since it has a major impact on flow properties. By exact conditioning, we mean not only that the facies is correct, but also that other properties like connectivity and size distributions for observed objects are preserved.

In full 3D, where wells are taken to be continuous, the approaches which assume that a randomly placed object has a positive probability of fitting a well observation will not work. The reason is that the edge is an observation of a continuous distribution, and the probability of drawing the exact location is zero. This means that algorithms like the ones described in Lantéjoul (1997) cannot be used directly. To a certain degree, this can be overcome by discretizing the reservoir and wells into a grid, as in Deutsch and Wang (1996), but correct conditioning is still very time consuming if the grid resolution used is not very rough.

We discuss two object models with the corresponding Metropolis–Hastings based algorithms where we specifically place objects at the well observations. A similar approach was used in Syversveen and Omre (1997). Whereas this solves the problem of perfect matching of well observations, a bias may be introduced in other aspects of the model. This bias may be introduced either by the approximations made in the algorithm or by the lack of convergence. The latter will occur if the algorithm converges too slowly for practical use. The idea of placing objects directly in the wells was also used non-iteratively in Viseur et al. (1998), but the algorithm used there does not focus on the correct conditioning.

We also suggest a simple test for bias in critical observators. This test relies on the double expectancy and may be applied to any object-based algorithm. When checking for the bias in this paper, we particularly consider three observators which are important for the flow in the reservoirs:

(1) The number of well penetrations of an object. Note that even if the object is completely eroded in the well location, the well is still considered to penetrate the object.
(2) The mean size of an object as a function of the number of penetrations.
(3) The global net/gross distribution, which is taken here as the volume ratio of objects to background.

The first point on this list is an obvious aspect to check, as it relates directly to the well conditioning and has major influence on the flow properties of the reservoir. It is also obvious that the size distribution of the penetrated objects has influence on the flow. Since large objects are more likely to be penetrated, the mean size should increase with the number of penetrations. Finally, if there is something wrong with the conditioning algorithm, this will often show up in the net/gross distribution as too much or too little sand in the vicinity of the wells. Although the true values for these parameters are not known in the real world, we can find them in our synthetic test case and use to check the validity of the algorithms.

Models and Geometry

Two models are considered in this paper. One is a fluvial channel model, while the other is a more general object model. Mathematically, they are very similar. The only difference is that the fluvial model has a net/gross indicator term instead of the traditional intensity term in point processes. Except for the inclusion of an interaction term, the general object model is a Poisson point process. The general model is typically used to generate smaller objects. This difference leads to different conditioning algorithms used for the two models.

In a typical facies modelling setting, we have well data w given as paths through the reservoir along which the facies is known. We also assume that if objects of the same facies are stacked, the border between them is known, but have no information whether observations in different wells belong to the same object. The well observations are taken to be exact. In addition, we have net/gross information γ for the fluvial model, which is given as a range that the net/gross must be within. The general object model does not use net/gross directly, but requires an intensity estimated from the net/gross. Finally, we may have inverted seismic data, which are correlated to the local facies probability. The model for a realisation r in the fluvial model is then

$$\pi(r \mid s, w, \gamma) = c f_{\mathrm{M}}(r) f_{\mathrm{S}}(s \mid r) f_{\mathrm{I}}(r) f_W(r) I(r \mid w, \gamma), \tag{1}$$

whereas the general object model with intensity λ is given by

$$\pi(r \mid s, w, \lambda) = c f_{\mathrm{M}}(r) f_{\mathrm{S}}(s \mid r) f_{\mathrm{I}}(r) f_W(r) I(r \mid w) f_N(r \mid \lambda). \tag{2}$$

Each f-term is a likelihood which describes a certain aspect of the model. The term f_{M} describes the object geometry, f_{S} is the seismic likelihood and f_{I} covers the interaction between objects. One object may condition several well observations, and there may be a prior distribution on the probability of this happening, given by $f_W(r)$. For the general object model, the term f_N is the distribution for the number of objects. Finally, I is an indicator function, which in the fluvial model is 1 if the net/gross is within desired limits and all wells are correctly conditioned, and 0 otherwise. In the general model, I is only a well indicator since the net/gross is handled through the intensity.

The geometry of the reservoir is described by the term f_{M}. This is the distribution for object shapes and sizes, and it is included the main geological part of the model. Interaction between the objects, handled by f_{I}, is also based on the geological input. The rest of the terms relate to data and describe how likely the well data, seismic data, well coupling interpretations and observed net/gross is, given the current realisation. For more details on these terms and the models, see Holden et al. (1998) for the fluvial model, and Lia et al. (1997) for the general model.

In this paper, we are not concerned with the terms f_{S}, f_{I} and f_W. The first and last of these terms can be relatively easily added on top of what is presented here, and would only lead to unnecessary complications. Although the term f_W is related to well couplings, it only provides weighting of the probabilities found in this paper. The main problem is to get the couplings correctly based only on the geometry. Interaction does pose a problem for the conditioning algorithm for the general object models, but it may be approximated. For the fluvial algorithm, the interaction is easily included.

This leaves us with the geometry term f_{M} and the indicator. The geometry of the objects is assumed to be independent. The unconditional likelihood for a realisation

$r = (n, b_1, \ldots, b_n)$, where n is the number of objects and b_i is the ith object, is then

$$f_{\mathrm{M}}(r) = n! \prod_{i=1}^{n} f_{\mathrm{MB}}(b_i),$$

where $f_{\mathrm{MB}}(b_i)$ is the likelihood for the geometry of body b_i. The factorial term is included since the numbering of bodies is irrelevant. By sorting the bodies by depth, each realisation has a unique representation, and the likelihood must be scaled with the number of possible permutations.

The fluvial channel is given by a straight line defining the expected channel location and its direction and four Gaussian fields defining the thickness (VT), width (HW), and vertical and horizontal displacements from the line (VD and HD). In addition, two 2-dimensional Gaussian fields are added, one at the top and one at the bottom of the channel. The channel is parameterized by observing these parameters in planes perpendicular to the line, equally spaced along it. This is shown in Fig. 1. Between these planes, which we call sections, linear interpolation is used to determine the channel location. The 2-dimensional Gaussian fields are sampled on finer grids and superimposed on the channel. Figure 2 shows a complete channel.

For the general object model, the objects are defined by a general basic shape. Each object has this shape, but it is scaled by length, width and thickness parameters,

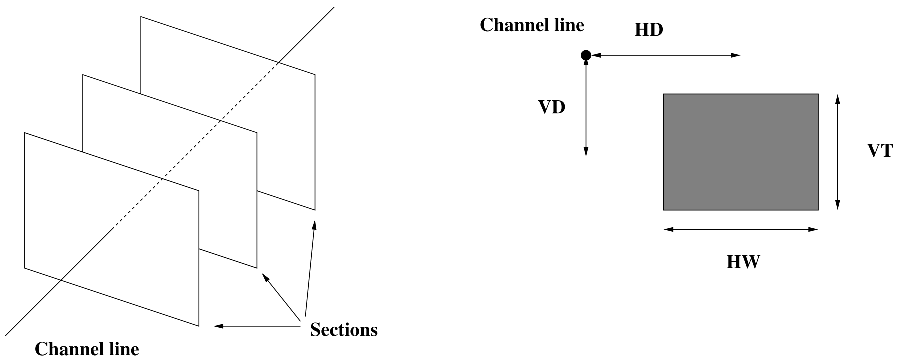

Fig. 1 Basic geometry of the fluvial model showing the channel line and sections (on the *left*) and a section view (on the *right*)

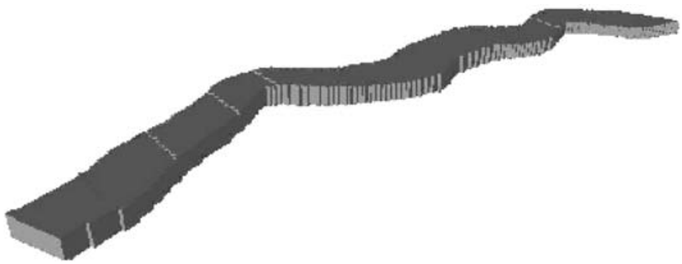

Fig. 2 Example of channel

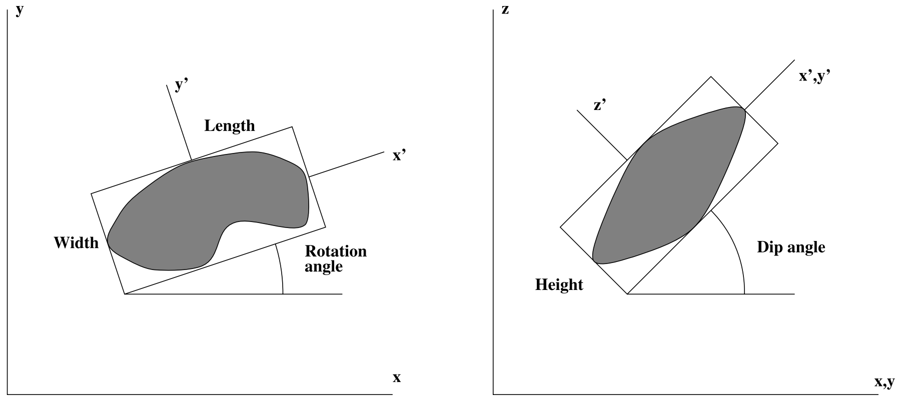

Fig. 3 Basic geometry of the general model shown in the xy-plane (on the *left*) and a plane parallel with the z-axis (on the *right*)

and also a rotation and dip angle are assigned. This is shown in Fig. 3. The shape may also depend on some stochastic parameters that differ between objects; this is not important for the conditioning algorithm. What is important is that these objects do have 2-dimensional Gaussian fields added to the top and bottom.

Conditioning Algorithms

For both models, we use Metropolis–Hastings (MH) algorithms (Hastings 1970), which means that objects are iteratively proposed and then accepted or rejected until convergence is reached. This is time consuming, but allows correct handling of well observations, which is very difficult in any direct sampling scheme.

An important reason for including both algorithms in this paper is that they are very different. The fluvial algorithm is based on an idea of maximum utilisation of information, and thus has slow, but hopefully efficient iterations, whereas the algorithm for the general object model is more brute force, relying on many but fast iterations. This difference in philosophy is based on the fact that the fluvial channels in general are larger than the objects modelled by the other algorithm. Larger objects are closer to more wells, and therefore more difficult to get correct by trial and error.

Fluvial Algorithm

The fluvial algorithm is based on simulated annealing. Both the net/gross term and partially the well conditioning term are annealed. During the simulation, channel facies cannot be placed where the wells have background, whereas the lack of channels in well observations is controlled by an annealing term. This allows to use an empty reservoir as the initial state. The indicator term $I(r \mid w, \gamma)$ in (1) is replaced by $J(r \mid w, \gamma, T)$, where T is the annealing temperature. As $T \to 0$, $J \to I$. Furthermore, $J = 1$ if $I = 1$, otherwise less than 1, and $J = 0$ only when r gives objects where the wells say there are none. A standard MH algorithm is used, with an advanced proposal function heavily influenced by the well data.

The MH algorithm is designed to simulate from any distribution and does not require the normalizing constant of the distribution to be known. It generates a Markov chain with the desired stationary distribution. A candidate state r_p is drawn from a proposal distribution $q(r_p \mid r)$, where r is the current state. The new state is then accepted with the acceptance probability

$$\alpha(j \mid i) = \min\left(1, \frac{\pi(r_p)q(r \mid r_p)}{\pi(r)q(r_p \mid r)}\right). \tag{3}$$

Any q can be used, but a proposal function $q(r_p \mid r) \approx c\pi(r)$, where c is a constant, will increase the acceptance probability and the rate of convergence.

In each step of our MH algorithm, we propose to add one channel, to remove one channel, or to change one of the existing channels. Of these three actions, the most complicated is to add a new channel. Let i be the current state, j be the proposed state with an object added, and n be the number of objects in state j. Furthermore, let $\beta(i \mid j)$ be the last term in (3). We have

$$\begin{aligned}
\beta(j \mid i) &= \frac{\pi(j)q(i \mid j)}{\pi(i)q(j \mid i)} \\
&= \frac{c \cdot n! \prod_{k=1}^{n} f_{\mathrm{MB}}(b_k) J(r_j \mid w, \gamma, T) p_{\mathrm{rem}}/n}{c \cdot (n-1)! \prod_{k=1}^{n-1} f_{\mathrm{MB}}(b_k) J(r_i \mid w, \gamma, T) p_{\mathrm{add}} g(b_n)} \\
&= \frac{f_{\mathrm{MB}}(b_n) p_{\mathrm{rem}} J(r_j \mid w, \gamma)}{g(b_n) p_{\mathrm{add}} J(r_i \mid w, \gamma)},
\end{aligned}$$

where p_{add} and p_{rem} are the probabilities of suggesting to add or to remove, and $g(b)$ is the likelihood of proposing the object b for addition. The p and J terms can be easily computed; the difficulty lies in the ratio $f_{\mathrm{MB}}(b_n)/g(b_n)$.

Our proposal function $g(b_n)$ is given by the following algorithm.

(1) Draw a channel line. With probability p_o, this line is drawn close to one of the currently unconditioned well observations, if such exists.
(2) Find all sections where there are wells close to the channel line. Identify observations in these wells that may be conditioned by this channel. Observations where the channel may be completely eroded are treated as if there was no well observation there.
(3) Choose one of these sections and a direction randomly. Draw the channel position in this section conditioned on all well observations that are closer to this section than any other.
(4) Choose a direction at random. Move along the sections in this direction, and draw the channel location in all sections close to wells. This is done from a distribution conditioned on the previously drawn sections and the well observations.
(5) The probability of including unconditional observation is approximated according to the part of the channel drawn so far.
(6) When the end of the channel is reached, proceed from the starting section and draw all the conditioning sections in the opposite direction.

(7) When drawing is conditioned on wells, compute the ratio between the conditional probability used and the unconditional probability of the channel location. The product of these ratios and the ratio computed in step 1 give the ratio $f_{\mathrm{MB}}(b_n)/g(b_n)$.

The idea here is to cancel out the more complicated terms in the ratio $f_{\mathrm{MB}}(b_n)/g(b_n)$, as it is rather inconvenient to compute the likelihood of Gaussian fields. We want $g(b_n) \approx f_{\mathrm{MB}}(b_n)$, with b_n avoiding all background observed in wells, and a significant probability of b_n passing through unconditioned well observations. Theoretically, any choice of g will do as long as the ratio can be computed, but the convergence could be extremely slow. For further details on our approximation, see Skorstad et al. (1999). The point here is that this algorithm enables us to compute the ratio.

When the edge of an object is observed, one or more of the parameters are deterministically determined, and thus there is a difference in dimension between the likelihood in the prior and the likelihood in the proposal function. However, this does not constitute a problem, as the imbalance here is compensated by the well conditioning indicator, which is substituted with a simulated annealing term. The ratio of the J-terms balances the acceptance probability for channels conditioning observations.

With the probability for adding a channel in place, the rest is straightforward. The acceptance probability for the removal depends on the inverse ratio, and since the channel has already been generated, the term $f_{\mathrm{MB}}(b)/g(b)$ has already been computed. This factor is stored with the channel and does not need to be computed again for the removal. Finally, a change is only a simultaneous addition and removal, and does not give rise to any new complications.

This algorithm depends on many parameters. Although the algorithm assures convergence for nearly any choice of these, the convergence speed may be seriously affected. There are two annealing terms that need to be scaled against each other and the rest of the model. Furthermore, the probability p_o of starting a channel based on an observation, the probability of including an observation in a channel and the probability for which edge is seen in a well (top, bottom, right or left) can all be chosen freely. The two latest parameters should intuitively be as close to the correct value as possible. For the other parameters, the choice is not so obvious, it is based more on trial and error, which makes validation important.

General Object Model Algorithm

The algorithm for the general object model handles the well conditioning by an initialising step. This step places objects in all well observations. Thus, in the main algorithm, the well conditioning term $I(r \mid w)$ is always satisfied. This has the advantage of avoiding the annealing term and its associated parameters which occurred in the previous algorithm. The drawback is that the interaction in the model cannot be handled completely correctly, since the observed objects are placed before unobserved ones are added.

The initialising step draws one unconditioned observation at random and runs a MH algorithm to generate an object covering this observation. This is repeated until all observations are conditioned. Note that since one object may condition several observations, the number of MH loops is generally less than the number of observations.

When only one object is changed, the likelihood for all other objects cancel out in the acceptance probability (3). This means that it does not matter whether the other objects are already generated or not, so we can fill the reservoir object by object. However, since all wells must be conditioned, a problem arises when a proposed object covers observations different from the current one. This implies that some other objects must also be changed, and so complicates the acceptance probability. We solve this problem by computing an approximate correction for these cases, based on the likelihood of having objects conditioning the observations that are not covered by both the current and proposed object.

The initialising algorithm is as follows.

(1) Draw at random one unconditioned observation o_i.
(2) Run a MH loop to find an object for this observation:
   (a) Draw the position x from the distribution $g_x(x \mid o_i)$.
   (b) Let ω be the marks except for the Gaussian fields. Draw ω from the distribution $g_\omega(\omega \mid x, o_i)$.
   (c) Identify all wells that the object may pass through (since the top and bottom Gaussian fields are not drawn yet, not all intersections are certain).
   (d) For each of these wells, draw whether the object should avoid the well or condition an observation in the well. This probability can be chosen freely, but should be close to the correct probability, and thus depend on x, ω and the observation. Let the product of the probabilities for the outcomes be denoted by $p(h \mid x, \omega)$, where h denotes all observations conditioned by this object. Note that passing through a well completely eroded is counted as avoiding the well.
   (e) Compute the conditioning likelihood for the top and bottom Gaussian fields, $l_c = f_z(z \mid h, w_a)$, where w_a are the wells the Gaussian field must avoid.
   (f) Compute
$$t_s = \frac{l_c \lambda(x) f_g(\omega \mid x)}{p(h \mid x, \omega) g(x \mid o_i) g_g(\omega \mid x, o_i)} \prod_{\substack{j \in o \\ j \neq i}} l(o_j)^{-1}, \tag{4}$$
   where $l(o_j)$ is the likelihood of an object conditioning only the observation o_j.
   (g) Accept the new proposal with probability $\min(1, t_s/t_c)$, where $t_c = 0$ initially. If it is accepted, set $t_c = t_s$.
(3) Continue until all the observations will be conditioned.

The Step 2g is a MH acceptance step. The ratio in the expression for t_s is the ratio of the prior likelihood of the newly generated object to the likelihood with which it was suggested, as in standard MH. The product term is due to the possibility of comparing objects which condition different sets of observations.

In the fluvial case, we avoided this problem by including a simulated annealing term. Here, we do an approximation by comparing the sets of objects which cover the same observations. Let h_c and h_s be the observations conditioned by the current object b_c and the suggested object b_s, respectively. The ratio in (4) becomes

$$\frac{t_s}{t_c} = \frac{f(b_s)\prod_{j\in h_s\backslash h_c} l(o_j)g(b_c)p(h_c \mid x_c, \omega_c)}{f(b_c)\prod_{j\in h_c\backslash h_s} l(o_j)g(b_s)p(h_s \mid x_s, \omega_s)}, \tag{5}$$

where f is the combined prior likelihood, g is the combined proposal likelihood and $h_s\backslash h_c$ is the observations that are in h_s but not in h_c. We see here that this is the acceptance probability when the state has been extended to include objects so that the same set of observations is covered by multiplying in the likelihood of having single objects in the remaining observations.

These likelihoods are given by

$$l(o_i) = \int_X \int_\Omega \lambda(x) f_g(\omega, x) I_{o_i}(\omega, x \mid w) f_z(z \mid \omega, x, o_i)\, \mathrm{d}\omega\, \mathrm{d}x,$$

where x is the location of the object, z denotes the top and bottom Gaussian fields, and ω is the rest of the parameters needed to describe the object. The indicator I_{o_i} is 1 if the object conditions observation o_i and no other observations and is not in conflict with any other well observation, and 0 otherwise. This can be computed by stochastic integration. By using these likelihoods instead of actual objects, we avoid the problem of generating these objects for each iteration, and we can also accept the generated object regardless of what will be added here later.

However, in the acceptance step we have explicitly stated that the observations in $h_c\backslash h_s$ and $h_s\backslash h_c$ should not be coupled with any other observation. This is not part of the model, since that would severely restrict our state space. Instead, we view this as an approximation, which is good if the probability of having isolated objects is large compared to the probability of having couplings in these observations.

The exact acceptance probability for our full state space could be found by substituting $\prod_{j\in h_s\backslash h_c} l(o_j)$ in (5) with the likelihood for all remaining unconditioned observations not in h_s, and by a similar operation in the denominator. However, this likelihood is too difficult to compute, and the number of such likelihoods is also much larger than the number of observations. Therefore, we choose to use this approximation which will tend to favour large couplings.

After the initialising step is done, the remaining objects are added by another MH algorithm. As for the fluvial model, this works by adding, removing or changing one object in each iteration. The significant difference is that all wells are now correctly conditioned at all times. This implies that if an object which conditions well observations is changed, the new object must condition exactly the same observations. Furthermore, observed objects can never be removed, only changed. This means that all couplings are decided by the initialising algorithm. The main reason for the ability to change observed objects is to better satisfy the interaction term.

Test Design and Results

Testing of the self-consistency of an algorithm can easily be done by using the principle of double expectation

$$E\big(E\big(f(\pi) \mid o\big)\big) = E\big(f(\pi)\big).$$

The double expectation of a stochastic variable is the same as the unconditional expectation value, given that the observations actually come from the correct distribution.

This gives us the following algorithm for consistency checking:

(1) Generate an unconditional reservoir.
(2) Drill synthetic wells at fixed locations in this reservoir.
(3) Generate a reservoir conditioned to these wells.
(4) For both reservoirs, compute and store the properties which we are checking.
(5) Repeat from Step 1 until the desired samplesize is reached.
(6) Compare the observed properties from the unconditional realisations with those of the conditional. They should be equal.

The point is that by generating new well observations for each conditional realisation, the observations are sampled from the model, and we get the double expectation correct. Using fixed well locations is especially effective for the net/gross error checking, since moving the wells around would make average maps unusable to detect net/gross bias around wells.

Note that this is a check of the self-consistency of the algorithm. It checks whether the algorithm samples from the correct conditional distribution, where the unconditional distribution is defined by the algorithm run without wells. This means that this setup can be used to check any algorithm for generating objects, whether it is model based or not. For model based algorithms, the main problem is to implement well conditioning. In this case, the unconditional algorithm will sample from the model distribution, and this test checks whether the conditional algorithm satisfies the model.

We applied this consistency check to our two model algorithms, and checked the properties mentioned in the introduction, that is well couplings, net/gross distribution, and size distributions. Table 1 shows some of the key parameters we used for the fluvial algorithm. The consistency checking algorithm given above was run for 100 iterations, generating one conditional and one unconditional realisation for each iteration. Our well information was gathered at 20 vertical well locations, concentrated in the upper left area of the reservoir. This leaves large areas without wells, making it easier to see if the net/gross distribution is perturbed by the wells.

The relative average net/gross map for our conditional realisations, together with the well locations, is shown in Fig. 4. The main channel direction is clearly visible, as the variance along this direction is small compared to the variance across. This map is the average net/gross map for the conditional realisations divided by the same map for the unconditional realisations, so a value of 1.2 means that there was 20% more net/gross at this location in the conditional realisations than in the unconditional ones.

Table 1 Key parameters used when testing the fluvial algorithm

Reservoir size	26 500 m by 15 000 m
Mean direction	130°
Mean width	700 m
Mean thickness	4 m
Net/gross	0.38

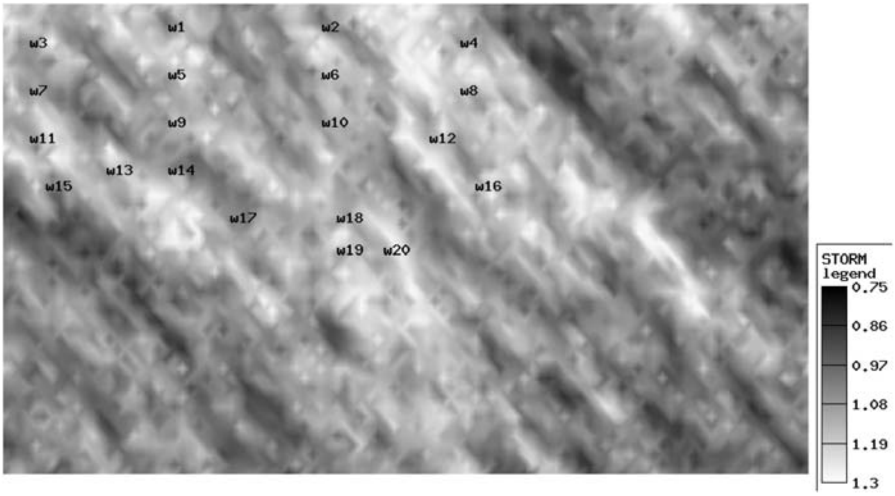

Fig. 4 Ratio of the mean net/gross in conditional realisations to that of unconditional ones for the fluvial model

This is more stable than the conditional net/gross map, due to the correlation between conditional and unconditional simulation in the wells. Ideally, this map should be 1 everywhere, but due to the small number of realisations used for the average, some variance must be expected.

Looking at this map, we see that there is no large bias in the net/gross. Most values are rather close to 1, and although there is some over-representation of high values in the well area, patches of values below 1 are also found in the same area, as well as high value areas far away from the wells. The discrepancy between conditional and unconditional net/gross is never above 30%. In a single unconditional realisation, the expected net/gross map value for these simulations is 0.38, with the standard deviation of 0.20. The average over 50 of these has the same expectation, but the standard deviation of 0.028. A ratio of 1.3 between such average maps occurs if they move two standard deviations in opposite directions. With 5000 nodes in the map, this is not very unlikely, but it should not happen often. Thus, the map indicates that while the algorithm is not a perfect sampler, it is good.

The number of couplings in the conditional and unconditional realisations is shown in the histogram in Fig. 5. The number of unobserved channels is 10 977 in the conditional realisations, compared to 10 374 in the unconditional ones. These numbers were not included on the graph, as they are too large compared to the rest. Again, the match is good. There may be a slight tendency towards creating too large

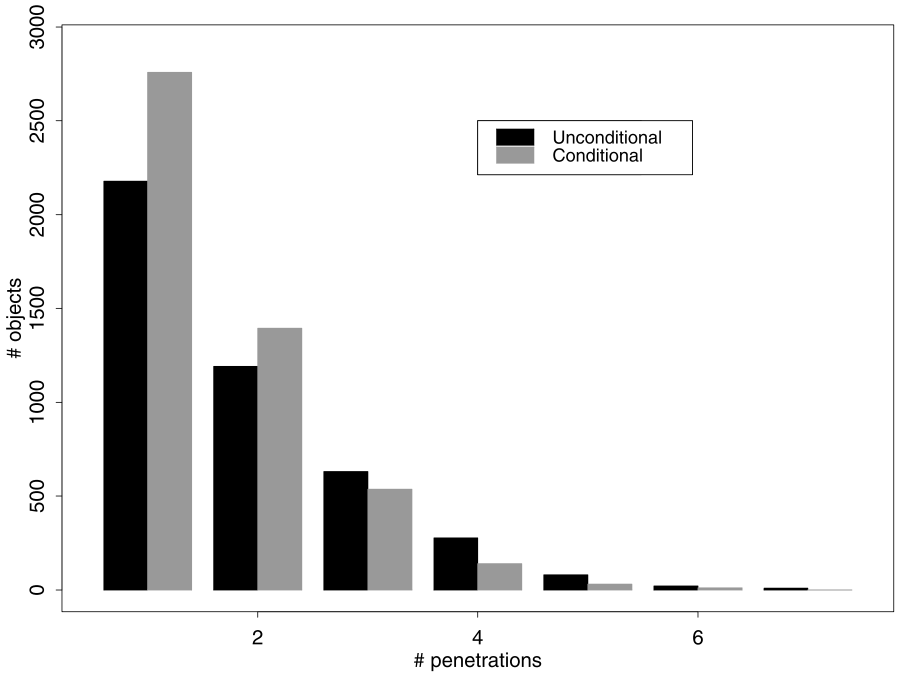

Fig. 5 Number of well penetrations per object in conditional and unconditional realisations for the fluvial model

couplings in the conditional reservoirs, but not on a scale that could be detected in a single realisation.

The width distribution is plotted against the number of well penetrations in Fig. 6. As expected, the width increases with the number of penetrations, since wide channels are more likely to get penetrated. The correspondence between conditional and unconditional realisations is very good here; the main discrepancies occur for large couplings where the sample-size is small. This is also a good indication of convergence, as no special care is taken in the proposal function to recreate this distribution.

We do not believe that the discrepancies between conditional and unconditional results are due to a lack of convergence here, as these results are equivalent to those of an earlier simulation with fewer iterations. That is, increasing the number of iterations did not alter the results. To the extent that the errors are outside normal variations, as may be the case with the coupling probabilities, the most likely explanation lies in our approximations. Our approximations seem to slightly overestimate the probability of suggesting an observed channel, thereby making it harder to accept and easier to remove. This will pull towards fewer observed channels and more couplings. However, the error is not large; the number of observed channels in the conditional realisations is less than 10% lower than in the unconditional ones.

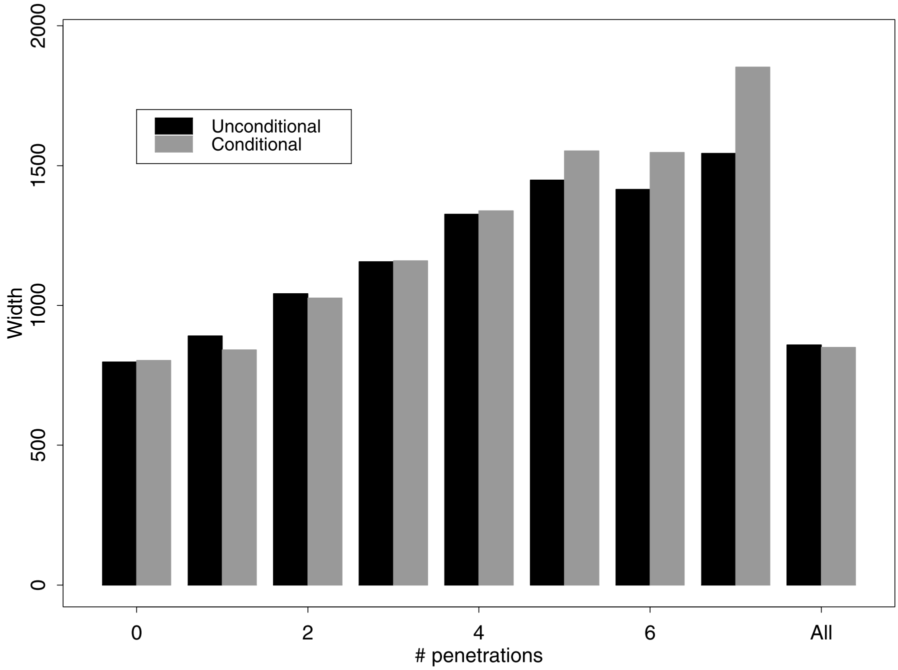

Fig. 6 Width of channels against the number of penetrations in conditional and unconditional realisations for the fluvial model. The *rightmost column* is the average width over all channels

Table 2 Key parameters used when testing the general algorithm

Reservoir size	1000 m by 1000 m by 100 m
Object shape	Rectangular
Mean width	300 m
Mean length	300 m
Mean thickness	10 m
Net/gross	0.065

The main reservoir model parameters for the general object model are shown in Table 2. The net/gross specified in the table is the average value obtained in the unconditional realisations, since the model uses an intensity parameter to control the number of objects. With this algorithm, 500 realisations were made of both the unconditional and conditional reservoir. Each pair of realisations took about 20 minutes to generate.

We placed 25 vertical wells randomly in the reservoir, as shown in Fig. 7, which also shows the relative net/gross map for these realisations. The map indicates a good match, with all conditional values within 25% of the corresponding unconditional ones. There are a couple of areas far from the wells, the lower left corner and middle of the upper edge, which both have low ratios. A low ratio indicates that there is

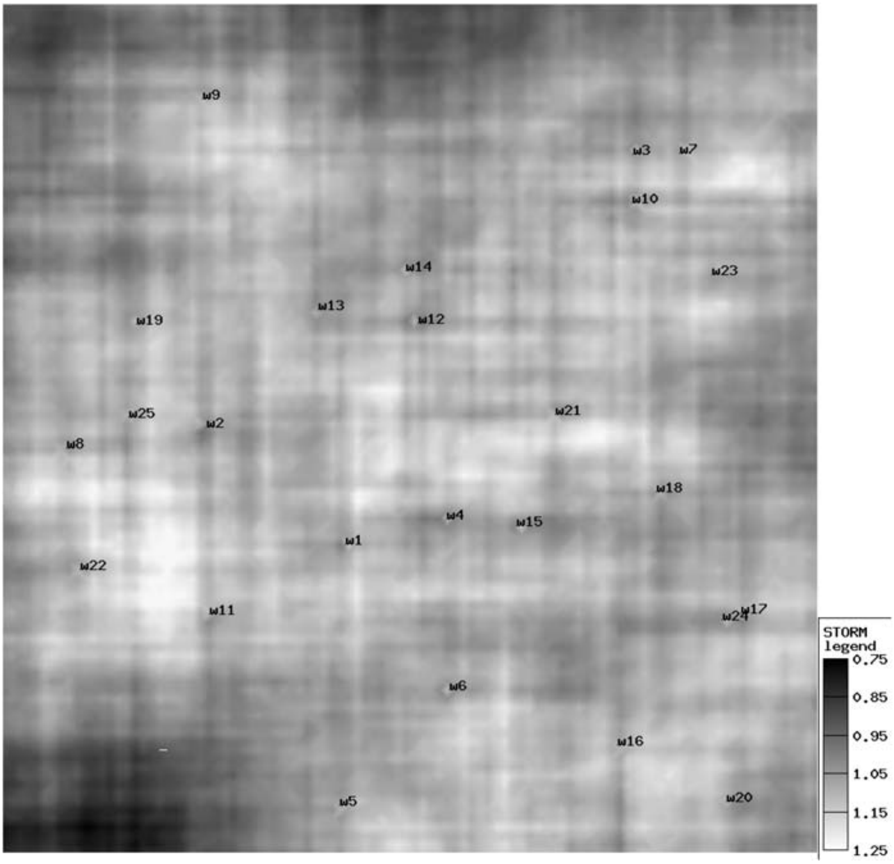

Fig. 7 Ratio of the mean net/gross in conditional realisations to that of unconditional ones for the general model

less net/gross in the conditional realisations than in the unconditional ones. This may indicate a slight tendency to put too many objects close to the wells.

The number of couplings is shown in Fig. 8. Again the match is good, especially for the real couplings, that is, objects with two or more penetrations. For 0 and 1 penetrations, there is a small bias towards observed objects, consistent with what was seen on the net/gross map. This is due to a large number of completely eroded penetrations, with 462 in the conditional realisations compared to 215 in the unconditional ones. It is difficult to believe that this discrepancy could be due to the lack of convergence, and there is no obvious link to the approximation used in the MH acceptance. The most probable explanation is that the approximated probability for being completely eroded is biased.

The width distribution of the objects in Fig. 9 shows that it is correctly reproduced even for the large couplings where observations are scarce. Since the objects used here are more rigid than the channels, the variance in size at a given coupling is smaller and decreasing with coupling size.

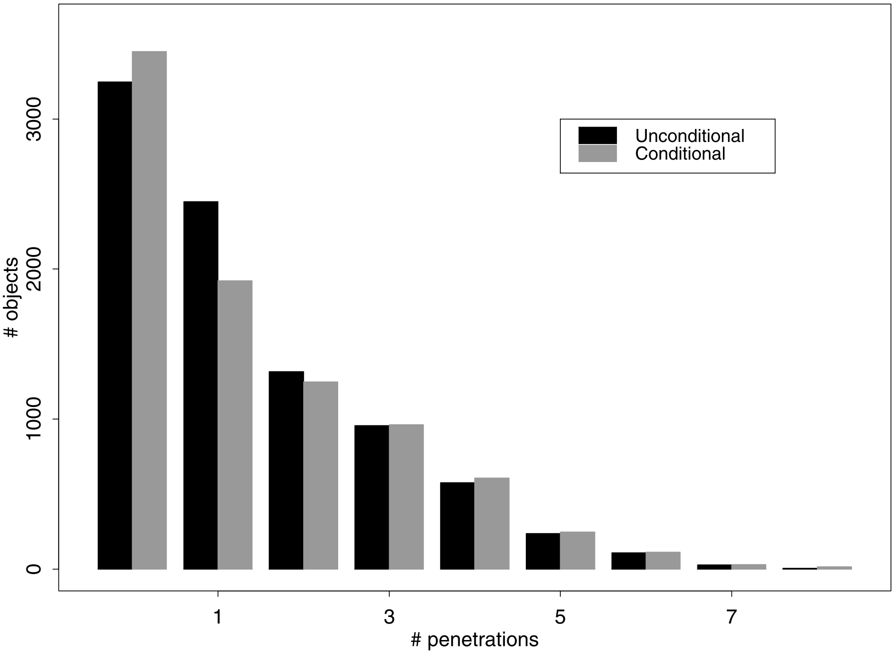

Fig. 8 Number of well penetrations per object in conditional and unconditional realisations for the general model

Discussion and Conclusions

Correct conditioning of object models on well data is complex. Theoretically, the problem can be solved by a Metropolis–Hastings algorithm, but it is difficult to design an algorithm which can place likely objects in the well observations and compute the correct acceptance probability. In addition, some measure of convergence must be used to make sure that the algorithm has run to convergence.

The algorithms proposed here use approximations, and hence do not simulate from the correct distribution. An exact algorithm could be made by substituting the complex proposal function in the fluvial algorithm with the simple proposal function of the general algorithm. However, this could dramatically decrease the acceptance rate, and a low acceptance rate, together with simulated annealing, would give severe convergence problems.

In practical use, the consistency and CPU time are under consideration. Although not theoretically perfect, these algorithms do reproduce important reservoir features in a reasonable time, and hence the approximations do not seem to introduce significant bias. The bias is smaller than the uncertainty in the model, the parameters and the understanding of multi-phase flow in highly heterogeneous reservoirs. Since the unconditional realisations are correct samples from our model, the algorithms presented do sample from the conditional model for all practical purposes. The tests can also be used to check convergence. Although a mathematical convergence is probably not achieved here, the satisfaction of these tests (and the fact that they do not change

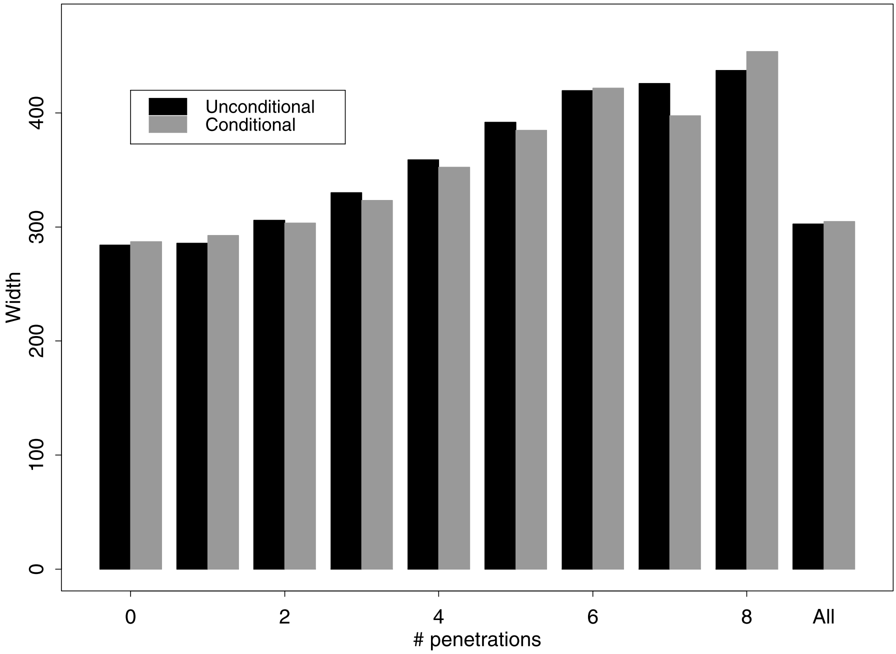

Fig. 9 Width of objects against the number of penetrations in conditional and unconditional realisations for the general model. The *rightmost column* is the average width over all objects

when the number of iterations is increased) indicates that the convergence is good enough for practical use.

References

Deutsch CV, Wang L (1996) Quantifying object-based stochastic modeling of fluvial reservoirs. Math Geol 28(7):857–880

Hastings WK (1970) Monte Carlo sampling methods using Markov chains and their applications. Biometrika 57(1):97–109

Holden L, Hauge R, Skare Ø, Skorstad A (1998) Modelling of fluvial reservoirs with object models. Math Geol 30(5):473–496

Lantéjoul C (1997) Iterative algorithms for conditional simulations. In: Baafi EY, Schofield NA (eds) Geostatistics Wollongong '96, proceedings '5th international geostatistics congress', Wollongong, Australia, 1996. Kluwer Academic, Dordrecht, pp 27–40

Lia O, Tjelmeland H, Kjellesvik L (1997) Modeling of facies architecture by marked point models. In: Baafi EY, Schofield NA (eds) Geostatistics Wollongong '96, proceedings '5th international geostatistics congress', Wollongong, Australia, 1996. Kluwer Academic, Dordrecht, pp 386–397

Skorstad A, Hauge R, Holden L (1999) Well conditioning in a fluvial reservoir model. Math Geol 31(7):857–872

Syversveen AR, Omre H (1997) Marked point models for facies units conditioned on well data. In: Baafi EY, Schofield NA (eds) Geostatistics Wollongong '96, proceedings '5th international geostatistics congress', Wollongong, Australia, 1996. Kluwer Academic, Dordrecht, pp 415–423

Viseur S, Shtuka A, Mallet JL (1998) New fast, stochastic, boolean simulation of fluvial deposits. In: 1998 annual technical conference and exhibition, society of petroleum engineers, New Orleans, LA, USA