Mathematical Geology, Vol. 38, No. 8, November 2006 (© 2007)
DOI: 10.1007/s11004-006-9057-5

Conditional Simulation of Multi-Type Non Stationary Markov Object Models Respecting Specified Proportions[1]

D. Allard,[2] R. Froidevaux,[3] and P. Biver[4]

Although Boolean model simulation has been widely used during the last two decades to simulate sedimentary bodies (especially in fluvio-deltaic environments), a key issue has subsisted. One of the most important parameter for object model simulation, namely the (non stationary) intensity of the underlying object process, is not a parameter provided by the end user but must instead be computed from other input parameters, such as local proportions of lithofacies, erosion rule between objects of different types and interaction between objects. This paper revisits a birth and death algorithm for simulating conditional, non stationary, multi-type objects models with interaction. It provides workable approximations for computing the local intensity of the underlying point process to respect proportion maps. Simulated examples show that this algorithm is able to reproduce the desired proportions. Important issues for implementing this algorithm are discussed.

KEY WORDS: Markov point processes, Strauss process, birth and death processes, reservoir modeling.

INTRODUCTION

Modeling heterogeneity is the first and, possibly, the most important step of a reservoir characterization study. Depending on the geological context, several simulation techniques can be envisioned to perform this first step: sequential indicator simulation (Goovaerts, 1997), transition probability simulation (Carle and Fogg, 1996), sequential simulation using multi-points statistics (Strebelle, 2002), truncated gaussian or plurigaussian simulation (Le Loc'h and Galli, 1997), Boolean simulation (Haldorsen and Macdonald, 1987; Lantuéjoul, 2002) or object models with interactions (Syversveen and Omre, 1997; Lia, Tjelmeland, and Kjellesvik, 1997). An important feature of object simulation, which sets it apart from the other techniques, is the fact that it is not pixel-based, i.e it does not generate

[1]Received 17 November 2005; accepted 6 March 2006; Published online: 27 February 2007.
[2]INRA, Unité de Biométrie, 84914 Avignon Cedex 9, France; e-mail: allard@avignon.inra.fr.
[3]FSS Consultants SA, 9 rue Boissonnas, 1227 Geneva, Switzerland.
[4]Total, CSTJF BA3112, Avenue Larribau, 64018 Pau Cedex, France.

values at the nodes of a pre-defined grid. Rather, it generates geometric shapes in space according to some probability laws. Any general purpose object simulation program for reservoir characterization should have the following features:

- Allow the simulation of multiple object types;
- Implement user-defined erosion rules between object of different types;
- Reproduce specified a priori proportions, after erosion, for each object type;
- Account for nonstationary proportions, object dimensions and orientations;
- Account for interactions (attraction or repulsion) between objects;
- Be conditional to existing hard-data.

Although Boolean model simulation has been widely used during the last two decades to simulate sedimentary bodies (especially in fluvio-deltaic environments), a key issue has subsisted. The most important parameter for object model simulation, namely the (non stationary) intensity of the underlying object process, is not a parameter provided by the end user but must instead be computed from the other input parameters listed above. A second very important issue is the fact that Boolean models assume independence between objects. As such they are inadequate for reproducing interactions between objects. Only a few work have attempted to incorporate interaction between objects. In Syversveen and Omre (1997a,b) a marked point process for modeling shale bodies is proposed but it does not contain conditioning data. Lantuéjoul (1997, 2002) proposed a birth and death process for simulating conditional Boolean models with non stationary intensities but does not address the problem of making a bridge between intensity and local proportion. In Lia, Tjelmeland, and Kjellesvik (1997) a rather general model for simulating objects using marked point processes is described. For conditioning on global constraints such as a global volume fraction they introduce a simulated annealing schedule encapsulating the MCMC simulation algorithm. This approach is very computer intensive and not optimal. Recently, Benito García-Morales (2003) proposed a method based on a Wiener filter to estimate a non stationary intensity from non stationary proportions. This method however assumes a single type of object, no interaction between objects and stationary distributions of size parameters.

In this paper, we revisit birth and death algorithms for simulating object models based on point processes with interaction. Going a step further than previous works, we derive accurate approximations for computing the local intensity, taking into account (i) non stationary proportions and non stationary object parameters; (ii) erosion rules in the case of multi-type objects; (iii) attraction or repulsion between objects. These approximations allow us to simulate object models with interaction that naturally honor global and local volume fractions. We then detail an effective algorithm based on birth and death processes able to simulate this object model respecting all the above-mentioned constraints. Important issues

arising when implementing this kind of algorithm are discussed and the algorithm is illustrated.

BOOLEAN MODELS

General Overview

A single object type Boolean model in $\mathbf{R}^d$, $d \geq 1$, or in a domain $\mathcal{D}$ of $\mathbf{R}^d$ is made of two parts (Stoyan, Kendall, and Mecke, 1995):

- A set of points of $\mathcal{D}$ (sometimes called seeds), denoted $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$, which follows a Poisson process characterized by its intensity θ describing the expected number of seeds per unit volume. This intensity may be varying in space; in which case it is denoted $\theta(\mathbf{u})$, $\mathbf{u} \in \mathcal{D}$. As a consequence of the Poisson assumption, seeds are independent to each other.
- Random variables, independent of the Poisson process, that attach to each of these seeds random marks describing the shape, dimensions and orientation of objects A. These random variables are described by their joint probability density, denoted ψ. The random marks are independent one from the other.

The key parameter for Boolean models simulation is the intensity parameter $\theta(\mathbf{u})$ which describes how many object centroïds are expected per volume unit. It is generally not readily available: geologists have good ideas about the proportion for each geological object type, but they have no feel for the number of such objects. Hence the need to compute the intensity from the proportion. In stationary conditions, it is well known that for Boolean models the proportion is related to the intensity according to the following relationship (Stoyan, Kendall, and Mecke, 1995; Lantuéjoul, 2002):

$$p = 1 - \exp\left\{-\theta \int_{\mathbf{R}^3} E[\mathrm{I}(\mathbf{u} \in A(\mathbf{v}))]\, d\mathbf{v}\right\} = 1 - \exp\{-\theta V\}, \tag{1}$$

where $\mathrm{I}(\mathcal{E})$ is the indicator function of the event $\mathcal{E}$, ($\mathrm{I}(\mathcal{E}) = 1$ if $\mathcal{E}$ is true, and 0 otherwise) $\mathbf{u}$ is any point of $\mathcal{D}$, $A(\mathbf{v})$ is a random object centered in $\mathbf{v}$ and V is the expectation of the volume of a random object A whose mark density is ψ. Inverting this relationship yields to

$$\theta = -\frac{1}{V} \ln(1 - p). \tag{2}$$

In practice, this congenial situation is the exception rather than the rule: a priori proportions and mark densities are not stationary; there are multiple types of

objects that overlap each other according to erosion rules; objects of a given type may show a tendency to attract each other or, conversely, to repulse each other. In all these situations Eq. (2) cannot be used directly and new equations must be derived. In most cases, more than one type of object needs to be simulated. Each of these $k = 1, \ldots, K$ types of objects has its own set of points $\mathbf{X}_k$, proportion $p_k(\mathbf{u})$, intensity $\theta_k(\mathbf{u})$ and mark density $\psi_k(\mathbf{u})$, $\mathbf{u} \in \mathcal{D}$. Using common usage, $p_k(\mathbf{u})$ is referred to as the proportion of facies k at location $\mathbf{u}$, but can also be understood as a probability of occurence, e.g. available from inverted seismic. The important point is that one must have at each location $\mathbf{u}$: $\sum_{k=1}^{K} p_k(\mathbf{u}) < 1$. The complementary to one, $p_0 = 1 - \sum_{k=1}^{K} p_k(\mathbf{u})$ is the probability or proportion of matrix.

Accounting for Multiple Object Types with Erosion Rules

Because objects of different types can overlap, one needs to adopt an "erosion rule" determining which type of object erodes the other. Three are commonly used: *random overlapping*, *vertical erosion* (objects with higher centroïds erode objects with lower centroïds) and *hierarchical erosion* rule whereby objects of type 1 always erode objects of type 2 which, in turn, always erode objects of type 3, etc. To ensure that in case of multiple object type simulation the target proportion of each type is correctly reproduced, the proportion p_k in Eq. (2) is substituted by a corrected proportion p'_k, as shown below. For the sake of lighter notation, we drop the reference to $\mathbf{u}$ in the equations below.

Hierarchical Erosion

In this case, the proportion of type 1 objects does not need to be corrected. Type 2 objects will be partly eroded by objects of type 1: for a visible proportion p_2, a corrected proportion $p'_2 = p_2/(1 - p_1)$ needs to be simulated. Recursively, for type k, the corrected proportion is:

$$p'_k = \frac{p_k}{1 - \sum_{i=1}^{k-1} p_i}. \tag{3}$$

It is easy to check that $p'_k \leq 1$ provided that $p_{\text{tot}} = \sum_{k=1}^{K} p_k < 1$.

Random Overlapping

Derivation of a corrected proportion in this case somewhat more complicated. We show in Appendix A that a second order approximation of this corrected

proportion is:

$$p'_k = p_k \left(1 + \frac{(1 + p_{\text{tot}})(p_{\text{tot}} - p_k)}{2} \right). \tag{4}$$

This correction accounts for intersection between two objects of different types. Intersections of three object types, or more, are ignored. Consider for example the following example $p_1 = 0.1$, $p_2 = 0.2$ and $p_3 = 0.3$. The total proportion is thus 0.6 and the proportion of "void" is 0.4. According to (4), the corrected proportions are $p'_1 = 0.140$, $p'_2 = 0.264$ and $p'_3 = 0.372$. The probability of "void" is thus $(1 - p'_1)(1 - p'_2)(1 - p'_3) = 0.394$, very close to the target proportion 0.4. In Appendix A we also show that this approximation always leads to a corrected proportion $p'_k < 1$ if $p_{\text{tot}} < 1$.

Vertical Erosion

In this case, it is also equally likely that an object of type k erodes an object of type l than the opposite. Hence, the same corrections as those used for random overlapping are used.

Accounting for Non Stationarity

If object proportions or object parameters are non stationary, the relationship between the non stationary proportions and non stationary parameters is similar to Eq. (1) but somewhat heavier. Dropping, for the sake of clearer notations, the subscript referring to the object type, the non stationary relationship between intensity and proportion is, at a point $\mathbf{u} \in \mathcal{D}$:

$$p(\mathbf{u}) = 1 - \exp\left\{ - \int_{\mathbf{R}^3} \theta(\mathbf{v}) E_{\psi(\mathbf{v})}[\mathrm{I}(\mathbf{u} \in A(\mathbf{v}))] \, d\mathbf{v} \right\}, \tag{5}$$

in which the expectation is computed with respect to the mark density with local parameters $\psi(\mathbf{v})$. This expression is extremely difficult (if not impossible) to invert. If $\psi(\mathbf{u})$ and $\theta(\mathbf{u})$ are smooth and slowly varying functions, we show in Appendix B that first order expansions in Eq. (5) can be used locally to approximate the local intensity $\theta_k(\mathbf{u})$ from the local corrected proportion $p'_k(\mathbf{u})$:

$$\theta_k(\mathbf{u}) \simeq -\frac{1}{V_k(\mathbf{u})} \ln(1 - p'_k(\mathbf{u})), \tag{6}$$

where $V_k(\mathbf{u})$ is the local expectation computed using the local mark probability density $\psi_k(\mathbf{u})$ and $p'_k(\mathbf{u})$ is the proportion corrected to account for erosion as described above.

MARKOV OBJECT MODELS

It is sometimes necessary to impose that objects of a given family are attracted to each other or on the contrary that there is some sort of repulsion between them. The general model is to consider that repulsion or attraction is a feature of the underlying point processes, but that marks will still be independent to each other. The appropriate framework for such point processes are Markov Point Processes (MPP). Poisson point processes on which are built Boolean models are particular cases of MPP, for which there is no repulsion and no attraction. We first make a quick survey of Markov point processes; a more comprehensive presentation of MPP can be found in Stoyan, Kendall, and Mecke (1995) or van Lieshout (2000). We then present the Strauss model and derive approximations for computing the intensity in this case.

General Presentation of Markov Point Processes

Markov point processes are defined relatively to a neighborhood relationship, denoted $\sim$, which must be a reflexive and symmetrical relationship. For example, the points $\mathbf{x}$ and $\mathbf{y}$ are neighbors (denoted $\mathbf{x} \sim \mathbf{y}$) if their distance $d(\mathbf{x}, \mathbf{y})$ is less than R for some $R > 0$. Then, the neighborhood of a point $\mathbf{u}$, denoted $\partial(\mathbf{u})$ is the set of points $\mathbf{x}$ of $\mathbf{X}$ such that $\mathbf{x} \sim \mathbf{u}$. Let us denote ϕ the pdf (with respect to the distribution of a Poisson process) of a point process. A point process $\mathbf{X}$ is a Markov point process with respect to the relationship $\sim$ if, for all configurations $\mathbf{X}$ such that $\phi(\mathbf{X}) > 0$, the conditional density $\phi(\mathbf{X} \cup \{\mathbf{u}\} \mid \mathbf{X})$ depends only on the neighborhood $\partial\mathbf{u}$:

$$\phi(\mathbf{X} \cup \{\mathbf{u}\} \mid \mathbf{X}) = \phi(\mathbf{X} \cup \{\mathbf{u}\} \mid \partial\mathbf{u}),$$

According to the Hammersley-Clifford theorem (van Lieshout, 2000) the pdf of a MPP depends only on functions of sets of points called *cliques*. In a clique each point is a neighbor of all other points of the clique. The simplest possible clique to consider consists of a single point. In this case there is no interaction and we are back to the classical Poisson process framework. In order to account for interaction, cliques of more than one point need to be considered. In practice, two point cliques only will be considered and pairwise interaction functions, denoted $\beta(\mathbf{x}, \mathbf{y})$, will be used to define the density ϕ of a

configuration $\mathbf{X}$:

$$\phi(\mathbf{X}) \propto \prod_{\mathbf{x}\in\mathbf{X}} \theta(\mathbf{x}) \prod_{\mathbf{x},\mathbf{y}\in\mathbf{X}:\mathbf{x}\sim\mathbf{y}} \beta(\mathbf{x},\mathbf{y}).$$

Strauss Processes

Among all pairwise interaction point processes, the simplest one is the Strauss process (Strauss, 1975; Kelly and Ripley, 1976) for which the interaction function is constant:

$$\beta(\mathbf{x},\mathbf{y}) = \beta \text{ if } \mathbf{x}\sim\mathbf{y} \quad \text{and} \quad \beta(\mathbf{x},\mathbf{y}) = 1 \text{ otherwise,} \tag{7}$$

with $0 \leq \beta \leq 1$. Its density is thus

$$p(\mathbf{X}) = \alpha\beta^{n(\mathbf{X})} \prod_{\mathbf{x}\in\mathbf{X}} \theta(\mathbf{x}), \tag{8}$$

where $n(\mathbf{X})$ is the number of neighbor pairs $\{\mathbf{x},\mathbf{y}\}$ of $\mathbf{X}$ with respect to the relationship $\sim$ and α is the normalization constant that does not need to be known explicitly for simulation purpose.

- If $\beta = 1$, there is no interaction whatsoever, and we are back to the non stationary Poisson point process with intensity $\theta(\mathbf{u})$.
- If $\beta < 1$, there is some repulsion. Configurations with a high number of neighbors have a smaller density and are thus less likely than configurations with a low number of neighbors. The point process is thus more regular than a Poisson point process. If $\beta = 0$, configurations with neighbors have a null density and are thus impossible. This model is referred to as the "hard core model". For $0 < \beta < 1$, it is sometimes called "soft core models."
- The case of an attraction corresponds to $\beta > 1$, but without additional constraints it is mathematically not admissible because the associated density does not integrate to a finite quantity (Kelly and Ripley, 1976). At the cost of changing slightly the model to ensure that the model integrates to a finite quantity, it is possible to define admissible models for attraction that are close to the Strauss process. Specifically, the interaction function

$$\beta(\mathbf{x},\mathbf{y}) = \begin{cases} 0 & \text{if } d(\mathbf{x},\mathbf{y}) < r \\ \beta & \text{if } r \leq d(\mathbf{x},\mathbf{y}) \leq R \\ 1 & \text{if } d(\mathbf{x},\mathbf{y}) > R \end{cases} \tag{9}$$

with $0 < r < R$, is an admissible model. The hard-core condition for distances less than r ensures the integrability of the density on finite domains of $\mathbf{R}^3$. In practice, this restriction is not important because it can be chosen arbitrarily small. A typical choice is the mesh of the grid on which the simulation is represented.

Neyman–Scott processes are parent-children processes (Stoyan, Kendall, and Mecke, 1995) ofen used to model clusters of objects: a parent point process is first simulated; then, a random number of children objects are randomly located in the vicinity of each parent point according to some distribution. Although being intuitively appealing to model clusters of objets, we will instead consider Strauss models for both cases with the additional hard-core condition (9) in case of attraction. In this unified framework all equations relating the proportions to the intensities are identical; only the value of the β parameter changes. Also, a single simulation routine can handle both cases, which would not be the case if a Neyman–Scott process was chosen to model attraction.

For Strauss models, the conditional density of adding to the configuration $\mathbf{X}$ a new point in $\mathbf{u}$ is

$$\phi(\mathbf{X} \cup \{\mathbf{u}\} \mid \mathbf{X}) = \phi(\mathbf{X} \cup \{\mathbf{u}\})/\phi(\mathbf{X}) = \theta(\mathbf{u})\beta^{n(\partial \mathbf{u})}, \tag{10}$$

where $n(\partial \mathbf{u})$ is the number of neighbors of $\mathbf{u}$. Hence, the parameter β can be interpreted as a factor multiplying locally the intensity for each point in the neighborhood of $\mathbf{u}$.

Deriving the Intensity for Object Models Based on Strauss Processes

We first consider the stationary case and turn to the general case later. Recall that for a Boolean model, intensity and proportion are related through Eq. (2): $\theta V = -\log(1 - p)$. For MPP, the proportion is not easily related to the intensity; there is no such relationship. We approximate Eq. (2) by its Taylor expansion of order two: $\theta V \simeq p + p^2/2$. The first term corresponds to the proportion of non intersecting objects. The second term corresponds to the correction induced by the intersection of two objects. Higher order intersections are ignored in this approximation. The accuracy of this second order approximation is to the order of $p^3/3$, e.g. the error is approximately 0.01 for $p = 0.30$.

Similarly, we propose to compute the correction for Markov point processes in the case of only two intersecting objects and to ignore higher order corrections. Therefore, $\theta V \simeq p(1 + cp/2)$, where c is a correction factor related to the parameter β, the size of the objects and the size of the interaction regions. In Appendix C, it is shown that cp is approximately the conditional probability that a point of $\mathcal{D}$ is

in an object A' given that it is already in an object A. The case $c = 0$ corresponds to non intersecting objects and $c = 1$ to the absence of interaction.

In reservoir simulations, objects have generally random size to account for the natural variability of geological objects. In order to end up with tractable equations we will consider here that objects and interaction boxes are rectangular. The case of more complex shapes is addressed by considering that they are encapsulated in boxes, the proportion being adjusted using fill factors. For the directions $i \in \{1, 2, 3\}$, let us denote X_i the dimensions of an object, R_i the dimension of the interaction box and r_i the minimal distance in case of attraction. In Appendix C, it is shown that

$$c(X_1, X_2, X_3) = 1 - (1 - \beta)\frac{\min(R_1, X_1)\min(R_2, X_2)\min(R_3, X_3)}{X_1 X_2 X_3} - \beta\frac{\min(r_1, X_1)\min(r_2, X_2)\min(r_3, X_3)}{X_1 X_2 X_3}. \tag{11}$$

Dimensions of objects are usually random, in which case expectation of the right hand-side of (11) must be taken. Two cases are considered here: (i) fixed interaction boxes and random size objects; (ii) random size objects with an interaction box proportional to the object.

Fixed Interaction Boxes

Markov point processes are usually defined for fixed interaction distances while object dimensions are usually random with a probability density functions (pdf) $f_i(x)$ on (a_i, b_i). We consider here independent dimensions in each direction, but other cases could also be considered. In Appendix C, it is shown that taking the expectation of Eq. (11) leads to

$$c = 1 - (1 - \beta)g_1(R_1)g_2(R_2)g_3(R_3) - \beta g_1(r_1)g_2(r_2)g_3(r_3), \tag{12}$$

where

$$g_i(u) = F_i(u) + u\{h_i(b_i) - h_i(u)\}, \quad h_i(u) = \int_u^{b_i} f_i(x)/x\, dx.$$

$F_i(x)$ is the cumulative probability function (cpf) corresponding to $f_i(x)$.

Proportional Interaction Boxes

Fixed interaction boxes are not always well suited to reservoir simulations. As an example, let us consider the case of avoiding objects modeled by a Strauss

process with $\beta = 0$. Then, small and large objects will have the same interaction box, larger than the largest object. This would lead to unacceptable geometries where small objects would be separated by artificially large distances. As an alternative, we propose that the interaction domain has dimensions proportional to the (random) dimensions of the object. In this case the ratios R_i / X_i are fixed, and equal to, say η. In Appendix C, it is shown that in this case

$$c = (1 - \beta)(1 - \eta^3)\mathrm{I}(\eta < 1) + \beta\{1 - g_1(r_1)g_2(r_2)g_3(r_3)\}. \tag{13}$$

Note that this equation is valid both for $\eta \geq 1$ (boxes are larger than the objects) and $\eta < 1$ (boxes are smaller than objects). This last case can correspond to objects authorized to present partial overlapping near the edges.

In a non stationary and multi-type object context, the local intensity of object type k must be computed using the local values of the parameters. It is thus:

$$\theta_k(\mathbf{u}) \simeq -\frac{p'_k(\mathbf{u})}{V_k(\mathbf{u})}\left(1 + c_k(\mathbf{u})\frac{p'_k(\mathbf{u})}{2}\right), \tag{14}$$

where $p'_k(\mathbf{u})$ is the proportion corrected for the erosion as described above and $c_k(\mathbf{u})$ is computed using the local probability distribution of the dimensions of object type k.

SIMULATION ALGORITHM

General Presentation of Birth and Death Processes

Non conditional Boolean models can be simulated directly: for each object type k, the number N_k of objects is drawn from a Poisson random variable with parameter $\Theta_k = \int_{\mathcal{D}} \theta_k(\mathbf{u})\, d\mathbf{u}$. Then these N_k objects are located randomly according to the intensity $\theta_k(\mathbf{u})$.

In all other cases (presence of conditioning data and/or Markov object models) simulation must be performed using a birth and death process. Birth and death processes are continuous time Markov Chains and belong the family of Markov Chain Monte Carlo (MCMC) methods. We refer to van Lieshout (2000) and Lantuéjoul (2002) for a more detailed presentation of birth and death processes applied to the simulation of spatial models. Since we are not interested by the dynamical aspect of the process, but rather by its equilibrium state, we will consider only the discrete time version of the process for sake of simplicity. Starting from an initial configuration, an object is either removed or added according to some transition probability that depends on the current state of the simulation at each time step. The transition probability is chosen in such a manner that the stationary

distribution of the Markov chain is precisely the density (8) we wish to simulate from. According to standard results of Markov chain theory, if the birth and death process is ergodic, then there exists a stationary spatial distribution and the convergence to the stationary distribution will always occur independently on the initial configuration (van Lieshout, 2000, p. 79). Ergodicity holds if the so called detailed balance equation is verified at each iteration.

We first show how a non stationary single type Strauss point process can be simulated on a domain $\mathcal{D}$. We suppose that the intensity $\theta(\mathbf{u})$ is known for all $\mathbf{u} \in \mathcal{D}$: for instance it has been derived from non stationary proportions as presented above. We denote $\Theta = \int_{\mathcal{D}} \theta(\mathbf{u})\, d\mathbf{u}$ the sum of $\theta(\mathbf{u})$ on $\mathcal{D}$. Θ is thus the expectation of the random number of objects in $\mathcal{D}$, also called the target number of objects in $\mathcal{D}$. For a current configuration $\mathbf{X}$, let us denote $q(\mathbf{X})$ the probability of choosing a birth ($1 - q(\mathbf{X})$ being thus the probability of choosing a death). A new point is proposed in $\mathbf{u}$ according to a density $b(\mathbf{X}, \mathbf{u})$ and accepted with a probability kernel $A(\mathbf{X}, \mathbf{X} \cup \{\mathbf{u}\})$. A point $\mathbf{x}_i$ of $\mathbf{X}$ is proposed for removal with the probability $d(\mathbf{X}, \mathbf{x}_i)$ and is indeed removed with a probability kernel $A(\mathbf{X}, \mathbf{X} \backslash \{\mathbf{x}_i\})$. The detailed balance condition states that a move from $\mathbf{X}$ to $\mathbf{X} \cup \{\mathbf{u}\}$ is equally likely than a move from $\mathbf{X} \cup \{\mathbf{u}\}$ to $\mathbf{X}$. It reads:

$$\begin{aligned} \phi(\mathbf{X}) q(\mathbf{X}) b(\mathbf{X}, \mathbf{u}) A(\mathbf{X}, \mathbf{X} \cup \{\mathbf{u}\}) &= \phi(\mathbf{X} \cup \{\mathbf{u}\})(1 - q(\mathbf{X} \cup \{\mathbf{u}\})) \\ &\quad \times d(\mathbf{X} \cup \{\mathbf{u}\}, \mathbf{u}) A(\mathbf{X} \cup \{\mathbf{u}\}, \mathbf{X}). \end{aligned} \tag{15}$$

If Eq. (15) is verified at each iteration, the Markov Chain will converge to the stationary distribution $\phi(\mathbf{X})$. Any choice of functions for $q(\cdot)$, $b(\cdot, \cdot)$ and $d(\cdot)$ are possible as long as Eq. (15) is verified at each iteration by computing the appropriate probability kernels A. Some choices are more natural than others however and lead to simplifications. We make the following choices. The density of a birth will be chosen proportional to the density $\theta(\mathbf{u})/\Theta$ and proportional to a field $\sigma(\mathbf{u})$ to be determined later: $b(\mathbf{X}, \mathbf{u}) = \theta(\mathbf{u})\sigma(\mathbf{u})/\Theta$. Mathematically speaking this auxiliary field is not necessary, but its introduction will prove to be useful. Points will be proposed for removal uniformly: $d(\mathbf{X}, \mathbf{x}_i) = 1/n(\mathbf{X})$. A birth will be proposed according to the probability $q(\mathbf{X}) = \Theta/(\Theta + n(\mathbf{X}))$. With the approximation $n(\mathbf{X})/\Theta \simeq 1$ once convergence is reached, Eq. (15) reduces to

$$A(\mathbf{X}, \mathbf{X} \cup \{\mathbf{u}\})\sigma(\mathbf{u}) = A(\mathbf{X} \cup \{\mathbf{u}\}, \mathbf{X})\beta^{n(\partial \mathbf{u})}. \tag{16}$$

For conditional simulations, the conditioning must be checked each time a new object is added or removed. The conditioning must take into account the erosion rule in the case of multi-type conditional simulations.

Choice of a Field $\sigma(\mathbf{u})$

Many choices are possible for $\sigma(\mathbf{u})$ provided that Eq. (16) is verified at each iteration. They are all mathematically equivalent. A first choice is to take $\sigma(\mathbf{u}) = 1$ and to determine the corresponding $A(\cdot, \cdot)$. This choice leads unfortunately to numerical instabilities and endless rejections in regions with low probability of acceptance. It was found to be more efficient to use the following fields:

- For a repulsion: $\sigma(\mathbf{u}) = \beta^{n(\partial\mathbf{u})}$. In this case, Eq. (16) is verified for $A(\cdot, \cdot) \equiv 1$, i.e. every action (addition or deletion) honoring the conditioning is accepted.
- For an attraction: $\sigma(\mathbf{u}) = \beta^{\min\{(n(\partial\mathbf{u})-n_{\max}),0\}}$, where $n_{\max}$ is a limit to the number of neighbors. Let us chose $A(\mathbf{X}, \mathbf{X} \cup \{\mathbf{u}\}) = \mathrm{I}[n(\partial\mathbf{u}) \leq n_{\max}]$, where again $\mathrm{I}[\cdot]$ is the indicator function. All additions are accepted provided that the number of neighbors of $\mathbf{u}$ does not exceed $n_{\max}$. Equation (16) is then verified for $A(\mathbf{X} \cup \{\mathbf{u}\}, \mathbf{X}) = \mathrm{I}[n(\partial\mathbf{u}) \leq n_{\max}]\beta^{-n(\partial\mathbf{u})}\beta^{n(\partial\mathbf{u})-n_{\max}} = \beta^{-n_{\max}}$: the rejection rate is constant. It is possible to show that regarding the stationary distribution, all constant rejection rates are equivalent. Hence, one can consider that $A(\mathbf{X} \cup \{\mathbf{u}\}, \mathbf{X}) = 1$. The main effect of introducing the parameter $n_{\max}$ is to stabilize the algorithm by avoiding a large quantity of objects piling on each other without increasing the proportion of this object type.

  For avoiding instabilities in the computation of $\sigma(\mathbf{u})$, one can alternatively chose $\sigma(\mathbf{u}) = \beta^{\min\{(n(\partial\mathbf{u})-n_{\max})/n_{\max},0\}}$, which is also equivalent to consider the above procedure for $\beta^{1/n_{\max}}$. But in this case, the proportion must be computed in Eqs. (12) and (13) with $\beta^{1/n_{\max}}$ instead of β.

Detailed Algorithm

Once the intensities have been derived from the specified proportions as presented in Sections 2 and 3, the algorithm proceeds as follows:

1. Build an initial configuration: birth and death processes need an initial configuration, i.e. an initial set of objects honoring the conditioning data. In the case of non conditional simulations, this initial configuration can be void. How an initial configuration is built will be detailed later.
2. Determine, for each object type k, the expected number of objects Θ_k to be simulated

$$\Theta_k = \int_{\mathcal{D}} \theta_k(\mathbf{u})\, d\mathbf{u} \simeq v \sum_{i \in G} \theta_k(i)$$

where G is a grid over the domain $\mathcal{D}$, v is the volume of a mesh of G and $\theta_k(i)$ is a discretization of $\theta(\mathbf{u})$ on this grid.

Compute the probabilities $\pi_k = \Theta_k / \sum_{l=1}^{K} \Theta_l$, for $k = 1, \ldots, K$.

3. Start the MCMC iterative process:
   a) Draw the type of object k to be added or removed using the probabilities $(\pi_k)_{k=1,\ldots,K}$.
   b) Draw the action to be taken (birth or death) using

$$q_k = \Theta_k/(N_k + \Theta_k), \quad 1 - q_k = N_k/(N_k + \Theta_k)$$

   where q_k is the probability of a birth and N_k is the current number of objects of type k.
   c) Depending on the selected action
      - in case of birth (addition of an object), draw randomly the location $\mathbf{u}$ of the new object centroïd proportionally to the intensity $\theta_k(\mathbf{u})\sigma_k(\mathbf{u})$; an addition can only be accepted if $n(\partial\mathbf{u}) < n_{\max}$.
      - in case of death (removal of an object), an object is selected randomly with equal probability from the list of objects of type k.
   d) Check the conditioning. If it is not honored, the action is not accepted and a new action is drawn. If it is honored, the action is accepted.
4. Repeat the process until the convergence criterion has been reached (see below).

IMPLEMENTATION ISSUES

The implementation of the algorithm described in the previous section raises some critical issues that are now discussed.

Border Effects

It is important to ensure that objects intersecting the domain $\mathcal{D}$ but whose centroïds are outside this domain can be simulated. A practical way consists in considering a bigger domain $\mathcal{D}^s$ whose dimensions are the dimensions of the domain under study $\mathcal{D}$, increased by the dimension of the largest conceivable object. There is one such domain $\mathcal{D}_k^s$ for each type of object and the expected number of objects of type k to be simulated must be computed on $\mathcal{D}_k^s$. For each object type k, the intensity $\theta_k(\mathbf{u})$ must be extrapolated on $\mathcal{D}_k^s$, for example by assigning to $\theta_k(\mathbf{u})$, $\mathbf{u} \in \mathcal{D}_k^s \backslash \mathcal{D}_k$, the value of the intensity at the closest point of $\mathcal{D}_k$.

For models with interaction, care must be taken to simulate correctly the Markov point process near the borders. An edge correction is applied for points $\mathbf{u}$ for which the interaction box has an intersection with the borders of the augmented domain $\mathcal{D}^s$. By construction there cannot be any neighbors outside $\mathcal{D}^s$. For a point $\mathbf{u}$ located near its border, the number of neighbors $n(\partial\mathbf{u})$ will therefore be underestimated as compared to points located in the center of $\mathcal{D}^s$. As a consequence, the field $\sigma(\mathbf{u})$ accounting for the interaction will be biased towards less interaction near the borders. In the case of repulsion for example, this bias results in an accumulation of objects near the border of $\mathcal{D}^s$, leading to less objects than expected in $\mathcal{D}$, and a proportion under the target.

Let $B_k(\mathbf{u})$ be the interaction box of an object centered in $\mathbf{u}$ and let $v(\mathbf{u}) = V(B_k(\mathbf{u}) \cap D_k^s)/V(B_k(\mathbf{u}))$ be the proportion of the volume of the interaction box included in the domain D_k^s. To account for this bias, the number of neighbors is corrected by dividing the actual number of neighbors by the proportion of the interaction box in D_k^s: $n(\partial\mathbf{u})^* = n(\partial\mathbf{u})/v(\mathbf{u})$. Note that when $v(\mathbf{u}) = 1$, ie. when the interaction box is completely included in $\mathcal{D}_k^s$, there is no correction.

Symmetrical Neighborhood

For Markov point processes, the neighborhood relationship must be symmetrical. This ensures the existence of the density of the point process and the convergence of the birth and death algorithm used to simulate the process. But, because the interaction boxes are not isotropic and the objects can have random orientations, it can happen that object A is in the neighborhood of object B while object B is not in the neighborhood of object A.

It is therefore necessary to symmetrized the neighborhood structure, which in Markov point processes is only used through the number of neighbors at the centroïd of the object, $n(\partial A)$. One approach is then to define weights in the neighborhood structure, such that the neighborhood will be symmetrical for the function $n(\partial A)$. We have chosen the following weight system: when A and B are both neighbor of each other, the weight is 1; when only $A \in \partial B$ but $B \notin \partial A$, both objects have a weight 1/2; and of course, when none are neighbors, their weight is 0.

Conditioning Data

Actual well data on which the simulations must be conditioned to are continuous intervals along a line. These kind of data are extremely difficult to honor exactly. Our approach is rather to assimilate the data to a discrete set of points. The algorithm checks whether the points are within the correct object, but no attempt

has been made so far ro check whether the entire sample thickness is within the correct object.

Building the Initial Configuration

In case of conditional simulation, an initial configuration is needed which will honor all the hard data. This is achieved by defining a restricted domain $\mathcal{D}^i$ which guarantees that any new simulated object located in $\mathcal{D}^i$, whatever its location, dimensions or orientation, will intersect at least one conditioning data. There is no birth and death process in this initial phase: new objects are added until all hard data are intersected. In practice, in case of strong inconsistencies between model parameters and conditioning data (e.g. between object dimensions and conditioning data inter-distances), proposed objects can be difficult to accept, and the rejection can be high. To avoid possible endless iterations, a maximum number of iterations can be specified for this phase. The proportion of hard data honored during this initial phase is an interesting measure of the consistency between the model and the data. In case of a very low conditioning rate, model parameters can be changed (e.g. object dimensions can be decreased). When the model and the data are consistent, the initial configuration phase is a matter of seconds, and more than 95% of the conditioning data are usuallyhonored.

During the iterations conditioning object can be removed only if another object also recovers the corresponding conditioning data. This will happen with probability one, except in one case: if there is only one type of object for a particular facies, and a hard-core model for this object type; but since in this case overlapping is forbidden, there is in theory non need for an MCMC algorithm. In all other cases, if we run the algorithm during a sufficient number of iterations, all objects of the initial configuration can be removed with probability one, because at some point another object has also recovered the conditioning data. In practice, only a fraction of those are replaced during a reasonable number of iterations. In general, the more the parameters and the conditioning data are consistent, the easier the conditioning objects can be replaced.

Convergence

The question of finding a criterion for deciding if the algorithm has reached convergence is a very difficult one. There is no general rule for evaluating the number of iterations necessary to reach a pre-specified distance between the theoretical stationary distribution and the actual distribution after n iterations, based only on the transition probabilities of the Markov Chain. A considerable amount of literature has been devoted to this subject. Meyn and Tweedie (1996), Gilks, Richardson, and Spiegelhalter (1996), Brooks and Roberts (1998) and Robert and Casella (1999) are important references on this topic.

The general idea for convergence diagnostic is then to select a few output parameters, and to compute simple statistics on these outputs for deciding whether convergence is reached or not. Some interesting statisticss are: the number of simulated objects, the number of conditioning objects that have been replaced and the number of neighbors. Raftery and Lewis (1992), Geyer (1994) and Yu and Mykland (1997) have proposed to estimate empirically the rate of convergence from output parameters, as the Monte Carlo Markov Chain iterates. This method presents some difficulties. In particular, an accurate estimation of the convergence rate necessitates many more iterations than necessary to reach the convergence. But the real difficulty in assessing convergence on a single Markov Chain is the fact that one might believe that convergence is reached only because the Markov Chain explored a very limited region of the state space. As Robert and Casella (1999, p. 371) point out *"you've only seen where you've been."* Gelman and Rubin (1992) and Gelman (1996) therefore proposed to run a set of Markov Chains having different starting points. The general idea is to compare on selected variables the pooled variance to the average of the variance within each Markov Chain. In case of convergence, these two variances should be close (these authors recommend a ratio less than 1.2).

In the case of conditional simulations, an initial configuration is first drawn without any reference to the model (Boolean or Strauss) and to the intensity. Lantuéjoul (2002) thus advises to run the Markov Chain until all objects of the initial configuration have been replaced. This is not always possible however. In some situations we observe that a significant fraction of these objects are indeed replaced, but that some objects are extremely difficult to remove. In other situations, there might be very few conditioning objects with many non conditioning ones. The probability to select a conditioning object among all objects for removal is thus very small. In these cases, waiting for the replacement of all objects would lead to an extremely large number of iterations, beyond any practical usefulness. Alternative criterions can be based on the number of remaining initial objects compared to the total number of objects, or on the rate at which initial objects are replaced. Lastly, one can impose a maximum number of iterations, as an increasing function of Θ_k for example.

Specifically, an approximate number of iterations, say n_0, is computed using probabilistic arguments based on the model parameters and the conditioning data. A within-chain assessment of the convergence (Brooks and Roberts, 1998) is then computed. If convergence is rejected, additional n_0 iterations are performed. This process is repeated until convergence is not rejected. An additional between-chain test of convergence (Gelman and Rubin, 1992; Gelman, 1996) is then be performed on multiple realizations of the MCMC algorithm. If convergence is rejected, additional iterations are needed.

In a series of tests not reported here, it was found that this approach leads to a sensible stopping rule. When the conditioning data were extracted from non

conditional simulations, the conditioning was almost perfectly (i.e. above 95%) honored, proportions (computed as averages on replicates) were accurately respected and the interaction was reproduced. As the inconsistency between model parameters and conditioning data increases, it was found that a larger number of iterations was necessary, and that conditioning was more difficult. When conditioning cannot be honored in a reasonable amount of time, it must interpreted as a wrong parametrization of the model, and the parameters must be changed.

SIMULATION EXAMPLES

The algorithm is now illustrated on two different examples both run on a $4000 \times 4000 \times 30$ domain (dimensions are in m), on which two types of objects are simulated.

In the first example, objects are fluviatile channel sands and coarse sand bodies (from diagenetic origin). Channels are assimilated to sinusoidal half-cylinder. The length of these channels are deemed infinite, i.e exceeding the dimension of the domain while the other parameters are randomly drawn from uniform distribution in [200, 400] [0.5, 1.5] [120, 140] [3000, 4000] [100, 300], for respectively, width, thickness, direction, wave length and amplitude. The coarse sand bodies are described as ellipsoids whose dimensions are also drawn randomly from uniform distributions in [1600, 2000], [1000, 1500], [0.5, 2.0], [110, 150], for respectively, length, width, thickness and direction. Conditioning data consists of 391 samples, from 24 wells, for which the lithological coding is available. Vertical a-priori proportion curves are imposed. For coarse sands the proportion increases steadily from a minimum value of 1% at the bottom of the reservoir to a maximum value of 57% at the top. The channels occur predominantly in the middle portion of the reservoir where the proportion is around 22%. The channel proportion decreases very sharply at the top and bottom of the reservoir. Note that the proportions (or local probability of occurrence) could be imposed on a 3D grid, but in this example we assumed horizontal stationarity of the proportions. There are no interaction between objects, neither for the channels nor for the coarse sands and a vertical erosion rule is enforced.

The criterion used to determine whether convergence is reached or not was that the number of remaining initial objects resulting from the initial conditioning phase should be less that 10% of the total number of simulated objects. 31000 iterations were necessary to meet this requirement. In addition, the between-chains Gelman ratio test of convergence (Gelman, 1996) computed after the completion of the ten realizations was positive, indicating it was reasonable to assume that convergence has occurred.

Ten realizations of the reservoir were performed. A typical cross-section of the reservoir, with conditioning data super-imposed, is shown on Fig. 1. Vertical proportion curves computed on the ten realizations correctly reproduce the

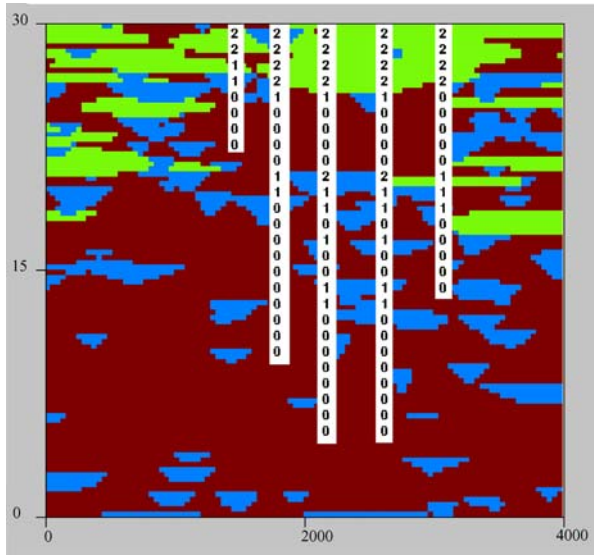


Figure 1. Vertical cross-section of a simulation with non stationary proportions and no interaction (vertical erosion rule), with conditioning data.

theoretical ones (Fig. 2). On average, the simulated proportion is 17% for the channels and 14% for the coarse sands, very close to the target proportions (16% in both cases). The variability of the experimental proportion (here computed on 10 realizations) is maximum when the target proportion is 0.5 and decreases as it gets closer to 0 and 1, an observation related to the fact that the variance of the experimental proportion is proportional to $p(1-p)$. On this example 100% of the conditioning information is honored. A word of caution about checking the conditioning is in order. Each conditioning data is assimilated to a point and the algorithm checks whether this point is within the correct object. The actual well sample, however, has a physical thickness. But not attempt has been made so far ro check whether the entire sample is within the correct object.

In the second example the objective is to perform a non-conditional simulation of two types of sedimentary bodies: channels (assimilated to sinusoidal half-cylinders) and dunes (fan-shaped). The length of the channel exceeds the domain dimensions and its other dimensions are randomly drawn from the following uniform distributions [500, 800], [0.3, 2], [−10, 10], [3000, 4000], [300, 500] for respectively, width, thickness, direction, wave length and amplitude. The dimensions of the dunes are also drawn from uniform distributions in [1200, 1400], [400, 600], [1, 3], [−10, 10], [0.6, 0.8] for respectively, length, width, thickness, direction and cone ratio. The a priori proportions are deemed stationary, 10% in

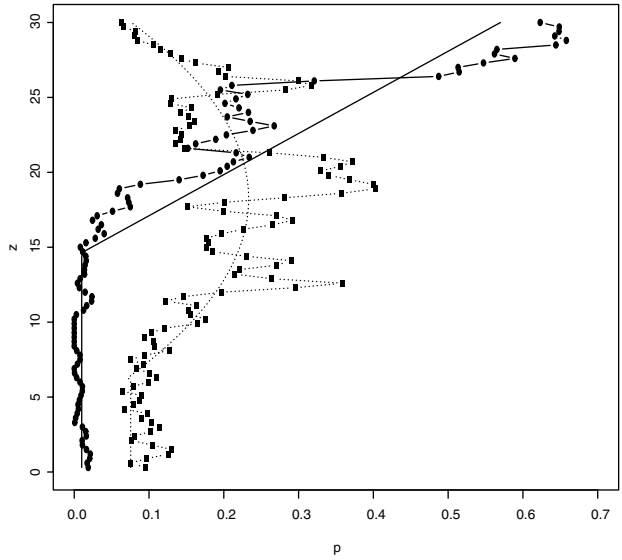
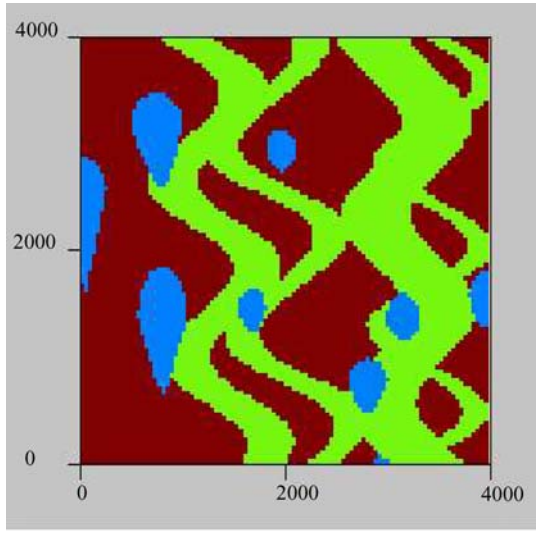


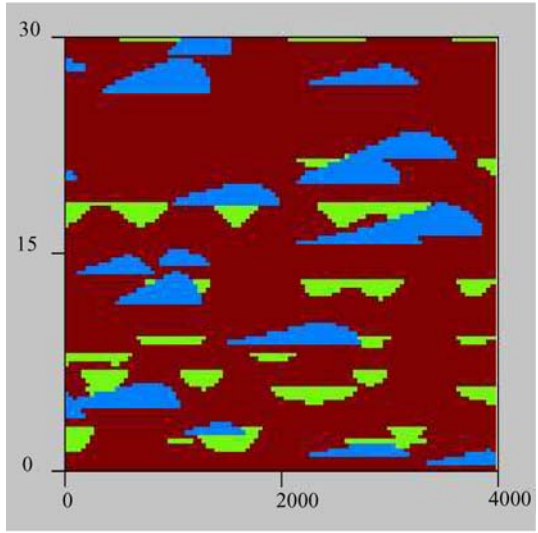
Figure 2. Theoretical and experimental vertical proportion curves computed on 10 replicates (*solid line* = fluviatile channels; *broken line* = coarse sand bodies).

both cases, but some interaction is imposed. Dunes repulse each other strongly ($\beta = 0.01$) in a relative interaction box that is 10% larger than the size of the objects. Channels will attract each other ($\beta = 10$) in a relative interaction box which is twice as large as the object width and height but that has the same length; thus the attraction operates only laterally and vertically. In this case a hierarchical erosion rule is applied: dunes always erode channels. In this case the within-chain convergence criterion was based on the total number of objects to be simulated and on the total number of neighbors. 21000 iterations were required to meet this criterion. Again, a positive result was obtained on the Gelman ratio calculated after the completion of ten realizations. Figure 3a and b show a vertical and an horizontal section through one of them. The average proportions computed on ten realizations are 9.3% for the dunes and 11.1% for the channels, very close to the target. The realizations also conform to the constraints which were imposed: dunes are distinct ones and systematically erode the channels which tend to cluster together.

These examples show that the algorithm presented above, equipped with approximations used to link the object proportions to the intensity, is able to perform simulations of the specified model while correctly reproducing the specified proportions.



(a)



(b)

Figure 3. Simulation with interactions and hierarchical erosion rule. (a) horizontal cross section; (b) vertical cross section.

DISCUSSION

The algorithm which has been presented offers a lot of flexibility and has proven effective for producing realistic simulations of reservoir heterogeneity respecting proportions in non-stationary situation and with interaction.

However, as is the case with all simulation algorithms, it is not universal and its limits of application should be respected. A critical issue, when performing conditional simulation, is the consistency between hard data, object parameters (shape and dimensions), interaction and target proportions. This consistency problem becomes even more important if some or all parameters are non-stationary.

- The approximations for computing the corrected proportions and the intensity are valid for low to moderate proportions. As a rough guideline it is recommended that each proportion does note exceed 50% and that there is at least 20% of matrix, even locally.
- The size of the object to be simulated is a critical issue. The larger the object the more difficult it will be to reproduce a target proportion and to honor conditioning data. Also, consistency between objects size (in particular thicknesses) and the typical dimensions of the conditioning lithofacies measured on wells must imperatively be verified.
- Although this algorithm can accommodate non-stationarity, care should be taken that this non stationarity should present smooth variations. In case of discontinuities, the proportion can be locally biased, event though proportions will be well respected globally.
- Consistency between the interaction parameter β, the dimension of the interaction box and the proportion is very important. For example, it is not possible to impose a hard-core condition with a large interaction dimension and a high proportion. More generally, the concept of neighborhood is not an intuitive one. Selecting too large a neighborhood may prove self-defeating, because in this case every point is a neighbor of every other point. The factor $\beta^{n(\mathbf{X})}$ is then constant, and is thus unable to reproduce any interactions.

REFERENCES

Benito García-Morales, M., 2003, Non stationnarité dans les modèles de type Boléen: application à la simulation d'unités sédimentaires: PhD Thesis, Ecole Nationale Supérieure des Mines de Paris, Centre de Géostatistique, 205 p.

Brooks, S., and Roberts, G., 1998, Assessing convergence of Markov Chain Monte-Carlo algorithms: Statistics and Computing, v. 8, p. 319–335.

Carle, S. F., and Fogg, G. E., 1996, Transition probability-based indicator geostatistics: Math. Geol., v. 28, no. 4, p. 453–476.

Gelman, A., and Rubin, D. B., 1992, Inference from iterative simulations using multiple sequences (with discussion): Stat. Sci., v. 7, p. 457–511.

Gelman, A., 1996, Inference and monitoring convergence, *in* Gilks, W. R., Richardson, S., and Spiegelhalter, D. J., eds., Markov Chain Monte Carlo in practice: Chapman & Hall, London, p. 131–144.

Geyer, C., 1994, On the convergence of Monte Carlo maximum likelihood calculations: J. Royal Stat. Soc., Ser. B, Vol. 56, p. 261–274.

Gilks, W. R., Richardson, S., and Spiegelhalter, D. J., 1996, Markov Chain Monte Carlo in practice: Chapman & Hall, London, 486 p.

Gooverts, P., 1997, Geostatistics for natural resources evaluation: Oxford University Press, Oxford, 483 p.

Haldorsen, H. H., and Macdonald, C. J., 1987, Stochastic modeling of underground reservoir facies (SMURF): Soc. Petrol. Engr. 16751, p. 575–589.

Kelly, F., and Ripley, B. D., 1976, A note on Strauss's model for clustering: Biometrika, v. 63, no. 2, p. 357–360.

Lantuéjoul, C., 1997, Iterative algorithms for conditional simulations, *in* Baafi, E. Y., and Schofield, N. A., eds, Geostatistics Wollongong '96: Kluwer Academic Publishers, Dordrecht, p. 27–40.

Lantuéjoul, C., 2002, Geostatistical simulation; models and algorithms: Springer-Verlag, Berlin, 256 p.

Le Loc'h, G., and Galli, A., 1997, Truncated plurigaussian method: Theoretical and practical points of view, *in* Baafi E. Y. and Schofield, N. A., eds, Geostatistics Wollongong '96: Kluwer Academic Publishers, Dordrecht, p. 211–222.

Lia, O., Tjelmeland, H., and Kjellesvik, L., 1997, Modeling of facies architecture by marked point models, *in* Baafi, E. Y., and Schofield, N. A., eds, Geostatistics Wollongong '96: Kluwer Academic Publishers, Dordrecht, p. 386–397.

Meyn, S. P., and Tweedie, R. L., 1993, Markov chains and stochastic stability: Springer-Verlag, London, 550 p.

Raftery, A. E., and Lewis, S. M., 1992, How many iterations in the Gibbs sampler? *in* Bernardo, J. M., Bayarri, M. J., Philip, D., A., Berger, J. O., Heckerman, D., Smith, A. F. M., and West, M., eds., Bayesian statistics 4: Oxford University Press, Oxford, p. 763–773.

Robert, C., and Casella, G., 1999, Monte Carlo statistical methods: Springer Verlag, New-York, 507 p.

Stoyan, D., Kendall, W., and Mecke, J., 1995, Stochastic geometry and its applications, 2nd ed.: John Wiley, Chichester, 436 p.

Strauss, D. J., 1975, A model for clustering: Biometrika, v. 62, no. 2, p. 467–475.

Strebelle, S., 2002, Conditional simulation of complex geological structures using multi-points statistics: Math. Geol., v. 34, no. 1, p. 1–22.

Syversveen, A. R., and Omre, H., 1997, Conditioning of marked point processes within a bayesian framework: Scand. J. Stat., v. 24, p. 341–352.

Syversveen, A. R., and Omre, H., 1997, Marked point models for facies units conditioned on well data, *in* Baafi, E. Y., and Schofield, N. A., eds., Geostatistics Wollongong'96: Kluwer Academic Publishers, Dordrecht, p. 415–423.

van Lieshout, M. N. N., 2000, Markov point processes: Imperial College Press, London, 175 p.

Yu, B., and Mykland, P., 1997, Looking at Markov sampler through cusum path plots: A simple diagnostic idea: Statistic and Computing, v. 8, p. 275–286.

APPENDIX A

We consider a random erosion rule: when two objects intersect each other, each object erodes the other one with the same probability 1/2. The approximation

(4) is derived by considering a second order correction. Corrections for intersections of three objects or more is a correction to the order of p_k^3, which can be neglected compared to p_k, when p_k is not too high.

Let us assume stationarity and let us denoted p'_k the corrected proportion of p_k. We suppose $0 < p_k < p_{\text{tot}} < 1$, where $p_{\text{tot}} = \sum_k p_k$ is the total proportion. To the second order approximation, an object of type k at a point $\mathbf{u}$ is visible either when it is not recovered by an object of a different type or when it is recovered by an object of a different type, but is eroded by an object of type k. Hence,

$$p_k = p'_k(1 - p'_{\text{tot}} + p'_k) + \frac{1}{2}p'_k(p'_{\text{tot}} - p'_k) + O\big(p'^3_{\text{tot}}\big), \tag{17}$$

where $O(p'^3_{\text{tot}})$ is a rest to the order of p'^3_{tot}. After some rearrangements,

$$p'_k = p_k + \frac{p'_k}{2}(p'_{\text{tot}} - p'_k) + O\big(p'^3_{\text{tot}}\big). \tag{18}$$

Then,

$$\begin{aligned} p'_{\text{tot}} &= \sum_k p'_k = \sum_k p_k + \frac{1}{2}\sum_k p'_k(p'_{\text{tot}} - p'_k) + O\big(p'^3_{\text{tot}}\big) \\ &= p_{\text{tot}} + \frac{1}{2}p'^2_{\text{tot}} - \frac{1}{2}\sum_l p'^2_l + O\big(p'^3_{\text{tot}}\big) \end{aligned} \tag{19}$$

Replacing p'_{tot} and p'_k by their expression above in the second term of the right hand side part of (18) and omitting all terms of order 4 or higher yields:

$$\begin{aligned} p'_k \simeq\ & p_k + \frac{1}{2}p_k(p_{\text{tot}} - p_k) + \frac{1}{4} \\ & \times \left[p'_k p_{\text{tot}}(p'_{\text{tot}} - p'_k) - 2p_k p'_k(p'_{\text{tot}} - p'_k) + p_k p'^2_{\text{tot}} - p_k \sum_l p'^2_l \right]. \end{aligned} \tag{20}$$

Considering that fourth order terms can be ignored, p'_k can be replaced by p_k in the expression into brakcets, which then simplifies and becomes:

$$2p_k p_{\text{tot}}(p_{\text{tot}} - p_k) - p_k \left(p_{\text{tot}} p_k - 2\sum_{l \neq k} p_l^2 \right). \tag{21}$$

Plugging (21) in (20) gives

$$p'_k \simeq p_k + \frac{1}{2}p_k(p_{\text{tot}} - p_k) + \frac{1}{2}p_k p_{\text{tot}}(p_{\text{tot}} - p_k) - \frac{1}{4}p_k\left(p_{\text{tot}}p_k - 2\sum_{l\neq k} p_l^2\right)$$

$$= p_k\left[1 + \frac{1}{2}(p_{\text{tot}} - p_k)(1 + p_{\text{tot}}) - \frac{1}{4}\left(p_{\text{tot}}p_k - 2\sum_{l\neq k} p_l^2\right)\right]. \tag{22}$$

The third term is always much smaller than the first and second ones. For the sake of simplicity it will be omitted. Finally, the approximation is

$$p'_k \simeq p_k\left[1 + \frac{1}{2}(p_{\text{tot}} - p_k)(1 + p_{\text{tot}})\right]. \tag{23}$$

Note that since $1 + p_{\text{tot}} < 2$, one has always $p'_k < p_k(2 - p_k) < 1$, because $x(2 - x) < 1$ when $0 < x < 1$. Let us consider the following example: There are three object types with $p_1 = 0.1,\ p_2 = 0.2,\ p_3 = 0.3$. Hence, $p_{\text{tot}} = 0.6$. The corrections given by (22) are:

$$p'_1 = 0.1(1 + 0.40 + 0.05) = 0.145$$
$$p'_2 = 0.2(1 + 0.32 + 0.02) = 0.268$$
$$p'_3 = 0.3(1 + 0.24 - 0.02) = 0.366$$

Using (23) instead of (22), i.e. omitting the last term in the correction, leads to the following values: $p'_1 = 0.140$, $p'_2 = 0.264$ and $p'_3 = 0.372$. The differences are much smaller than the statistical fluctuations due to the simulations. Plugging these values back in (17) leads to the following proportions: $p_1 = 0.095$, $p_2 = 0.196$ and $p_3 = 0.297$, all very close to the target.

APPENDIX B

We start from:

$$p(\mathbf{u}) = 1 - \exp\left\{-\int_{\mathbf{R}^3} \theta(\mathbf{v})E_{\psi(\mathbf{v})}[\mathrm{I}(\mathbf{u} \in A(\mathbf{v}))]\,d\mathbf{v}\right\}. \tag{24}$$

Let us denote $p_{\mathbf{v}}(\mathbf{v}-\mathbf{u}) = E_{\psi(\mathbf{v})}[\mathrm{I}(\mathbf{u} \in A(\mathbf{v}))]$ the probability that $\mathbf{u}$ is covered by an object with distribution $\psi(\mathbf{v})$ located in $\mathbf{v}$. Then,

$$-\ln(1-p(\mathbf{u})) = \int_{\mathbf{R}^3} \theta(\mathbf{v}) p_{\mathbf{v}}(\mathbf{v}-\mathbf{u})\, d\mathbf{v}. \tag{25}$$

We will assume that objects are stochastically symmetrical around their centroïd, i.e: $p_{\mathbf{v}}(\mathbf{v}-\mathbf{u}) = p_{\mathbf{v}}(\mathbf{u}-\mathbf{v})$. The first order expansion of $\theta(\mathbf{v})$ around $\theta(\mathbf{u})$ is

$$\theta(\mathbf{v}) = \theta(\mathbf{u}) + (\mathbf{v}-\mathbf{u})^t \nabla\theta(\mathbf{u}) + O(||\mathbf{v}-\mathbf{u}||^2).$$

Likewise, $p_{\mathbf{v}}(\mathbf{v}-\mathbf{u})$ has a first order expansion: $p_{\mathbf{v}}(\mathbf{v}-\mathbf{u}) = p_{\mathbf{u}}(\mathbf{v}-\mathbf{u}) + (\mathbf{v}-\mathbf{u})^t \nabla p_{\mathbf{u}}(\mathbf{v}-\mathbf{u}) + O(||\mathbf{v}-\mathbf{u}||^2)$, with

$$\nabla p_{\mathbf{u}}(\mathbf{v}-\mathbf{u}) = \frac{\partial E_{\psi(\mathbf{v})}[\mathrm{I}(\mathbf{u} \in A(\mathbf{v}))]}{\partial a(\mathbf{u})} \frac{\partial a(\mathbf{u})}{\partial \mathbf{u}},$$

where $a(\mathbf{u})$ denotes the set of parameters of the random objects distribution functions. Hence, the first order approximation of (25) is:

$$\begin{aligned} -\ln(1-p(\mathbf{u})) &= \int_{\mathbf{R}^3} \theta(\mathbf{v}) p_{\mathbf{v}}(\mathbf{v}-\mathbf{u})\, d\mathbf{v} \\ &= \theta(\mathbf{u}) \int_{\mathbf{R}^3} p_{\mathbf{u}}(\mathbf{v}-\mathbf{u})\, d\mathbf{v} + \nabla\theta(\mathbf{u}) \int_{\mathbf{R}^3} (\mathbf{v}-\mathbf{u})^t p_{\mathbf{u}}(\mathbf{v}-\mathbf{u})\, d\mathbf{v} \\ &\quad + \theta(\mathbf{u}) \int_{\mathbf{R}^3} (\mathbf{v}-\mathbf{u})^t \nabla p_{\mathbf{u}}(\mathbf{v}-\mathbf{u})\, d\mathbf{v} + R(\mathbf{u}), \end{aligned}$$

where $R(\mathbf{u})$ is a rest. But, $\int_{\mathbf{R}^3} p_{\mathbf{u}}(\mathbf{v}-\mathbf{u})\, d\mathbf{v} = V(\mathbf{u})$ and $\int_{\mathbf{R}^3} (\mathbf{v}-\mathbf{u})^t p_{\mathbf{u}}(\mathbf{v}-\mathbf{u})\, d\mathbf{v} = 0$ because of the symmetry assumption. Since objects have finite dimensions, the integral on $\mathbf{R}^3$ in the third term is in fact an integral on a domain whose dimensions are the maximal dimensions of objects. If $p_{\mathbf{u}}(\mathbf{v}-\mathbf{u})$ is slowly varying and object dimensions are not too large, the third integral is negligible compared to the first term. Then,

$$-\ln(1-p(\mathbf{u})) \simeq \theta(\mathbf{u}) V(\mathbf{u}) + O(||\mathbf{v}-\mathbf{u}||^2).$$

APPENDIX C

In this appendix we establish Equations (11), (12) and (13). We recall that for a Boolean process,

$$\theta V \simeq p + p^2/2 = p(1 + p/2) = P(\mathbf{u} \in \text{object } A) \times\{1 + P(\mathbf{u} \in \text{object } A' \mid \mathbf{u} \in \text{object } A))/2\}.$$

But, according to (10), for a Strauss process,

$$P(\mathbf{u} \in \text{object } A' \mid \mathbf{u} \in \text{object } A) = cV\theta,$$

where $c = \beta$ if the centroïd of A' is in the neighbourhood of A and $c = 1$ otherwise. Thus,

$$\begin{aligned}\theta V &\simeq p(1 + c\theta V/2)\\ &\simeq p\big(1 + \frac{c}{2}p(1 + c\theta V/2)\big)\\ &\simeq p(1 + cp/2 + c^2 p\theta V/4).\end{aligned}$$

Hence, to the second order, $\theta V \simeq p + cp^2/2$ and c is a correction factor equal to β for distances less than the iteraction distance and equal to 1 otherwise.

We now compute explicitely c. To introduce our derivation, we first consider a 1-d process of objects with fixed dimension X and fixed interaction box of dimension R. Then, $c(X) = \beta$ if $X \leq R$ and $c(X) = \beta R/X + (X - R)/X$ if $X > R$. It is possible to rearrange these conditions into a single equation:

$$c(X) = 1 - \min(R, X)/X + \beta \min(R, X)/X. \tag{26}$$

In the case of attraction with a minimal interaction distance r, Eq. (26) becomes

$$c(X) = 1 - \min(R, X)/X + \beta\{\min(R, X)/X - \min(r, X)/X\}. \tag{27}$$

Similarly, for three dimensional objects with dimensions (X_1, X_2, X_3), interaction box (R_1, R_2, R_3) and minimal interaction distances (r_1, r_2, r_3),

$$\begin{aligned}c(X_1, X_2, X_3) = 1 - (1 - \beta)&\frac{\min(R_1, X_1)\min(R_2, X_2)\min(R_3, X_3)}{X_1 X_2 X_3}\\ -\beta&\frac{\min(r_1, X_1)\min(r_2, X_2)\min(r_3, X_3)}{X_1 X_2 X_3}\end{aligned} \tag{28}$$

When the dimensions of the objects are random, expectation of (28) must be taken. Let us define

$$g(u_1, u_2, u_3) = E\left\{\frac{\min(u_1, X_1)\min(u_2, X_2)\min(u3, X_3)}{X_1X_2X_3}\right\}.$$

Let us further denote $f_i(x)$ the probability distribution function (pdf) of the random dimension X_i in direction $i = 1, 2, 3$ and (a_i, b_i) its range of variation.

(i) *Fixed interaction boxes*
For independent dimensions,

$$\begin{aligned} &g(u_1, u_2, u_3) \\ &= \int_{a_1}^{b_1}\int_{a_2}^{b_2}\int_{a_3}^{b_3} \frac{\min(u_1, x_1)\min(u_2, x_2)\min(u_3, x_3)}{x_1x_2x_3} f_1(x_1)f_2(x_2)f_3(x_3)\,dx_1\,dx_2\,dx_3 \\ &= \prod_{i=1}^{3}\int_{a_i}^{b_i} \frac{\min(u_i, x_i)}{x_i} f_i(x_i)\,dx_i = \prod_{i=1}^{3} g_i(u_i). \end{aligned} \tag{29}$$

We detail how $g_i(u)$ can be computed. Dropping the index refering to the dimension for lighter notation, we have for $a \le u \le b$,

$$\begin{aligned} g(u) &= \int_a^b \frac{\min(u, x)}{x} f(x)\,dx = \int_a^u f(x)\,dx + u\int_u^b \frac{f(x)}{x}\,dx \\ &= F(u) + u\{h(b) - h(u)\}, \end{aligned} \tag{30}$$

where $F(x)$ is the cumulative probability function (cpf) corresponding to $f(x)$ and $h(u) = \int_u^b f(x)/x\,dx$. Note that $g(u) = 0$ if $u \le a$, and $g(u) = 1$ if $u \ge b$. Finally, we have

$$c = 1 - (1 - \beta)g_1(R_1)g_2(R_2)g_3(R_3) - \beta g_1(r_1)g_2(r_2)g_3(r_3).$$

(ii) *Proportional interaction box*
Let us suppose $R_i/X_i = \eta \quad$ for $i = 1, 2, 3$.

- $\eta < 1$. Then $\min(R, X) = R$ and (28) becomes

$$c(X_1, X_2, X_3) = 1 - (1 - \beta)\eta^3 - \beta\frac{r_1r_2r_3}{X_1X_2X_3}.$$

Taking its expectation yields

$$c = 1 - (1 - \beta)\eta^3 - \beta g_1(r_1)g_2(r_2)g_3(r_3).$$

- $\eta \geq 1$. Then $\min(R, X) = X$. Considering that $r_i < X_i$, (28) becomes

$$c = \beta\{1 - g_1(r_1)g_2(r_2)g_3(r_3)\}.$$

These two equations can be rearranged in a single one:

$$c = (1 - \beta)(1 - \eta^3)I(\eta < 1) + \beta\{1 - g_1(r_1)g_2(r_2)g_3(r_3)\}.$$