Simulating Large Gaussian Random Vectors Subject to Inequality Constraints by Gibbs Sampling

Xavier Emery · Daisy Arroyo · María Peláez

Abstract The Gibbs sampler is an iterative algorithm used to simulate Gaussian random vectors subject to inequality constraints. This algorithm relies on the fact that the distribution of a vector component conditioned by the other components is Gaussian, the mean and variance of which are obtained by solving a kriging system. If the number of components is large, kriging is usually applied with a moving search neighborhood, but this practice can make the simulated vector not reproduce the target correlation matrix. To avoid these problems, variations of the Gibbs sampler are presented. The conditioning to inequality constraints on the vector components can be achieved by simulated annealing or by restricting the transition matrix of the iterative algorithm. Numerical experiments indicate that both approaches provide realizations that reproduce the correlation matrix of the Gaussian random vector, but some conditioning constraints may not be satisfied when using simulated annealing. On the contrary, the restriction of the transition matrix manages to satisfy all the constraints, although at the cost of a large number of iterations.

X. Emery (✉)
Department of Mining Engineering, University of Chile, Santiago, Chile
e-mail: xemery@ing.uchile.cl

X. Emery
Advanced Mining Technology Center, University of Chile, Santiago, Chile

D. Arroyo · M. Peláez
Department of Mathematics, Northern Catholic University, Antofagasta, Chile

D. Arroyo
e-mail: darroyo@ucn.cl

M. Peláez
e-mail: mpelaezmath@gmail.com

1 Introduction

The simulation of Gaussian random vectors subject to inequality constraints arises in the analysis of spatial data, when all or part of the data values are interval constraints rather than single numbers. In a mineral resources evaluation, this situation occurs when the depth of a geological horizon is greater than the depth at which drilling has been stopped, when a measured grade is smaller than a detection limit, or when working with soft data defined by lower and upper bounds (Journel 1986). Interval constraints are also encountered when simulating continuous variables represented by chi-square random fields, as well as indicators and categorical variables represented by truncated Gaussian or plurigaussian random fields (Matheron et al. 1987; Le Loc'h et al. 1994; Bárdossy 2006; Emery 2005, 2007a, 2007b; Armstrong et al. 2011). For instance, let d be a positive integer and consider an indicator variable obtained by truncating a stationary Gaussian random field $Y = \{Y(\mathbf{x}) : \mathbf{x} \in \mathbb{R}^d\}$ at a given threshold $y \in \mathbb{R}$. The following procedure can be used to simulate the indicator (Lantuéjoul 2002):

(1) Simulate Y at the data locations, conditioned by the indicator data.
(2) Simulate Y at the target locations of $\mathbb{R}^d$, conditioned by the Y-vector obtained in step (1).
(3) Truncate the simulated Gaussian random field to obtain a realization of the indicator.

Any multivariate Gaussian simulation algorithm can be considered in step (2). Regarding step (1), one can use an iterative algorithm known as the Gibbs sampler. This algorithm is currently widely used to simulate Gaussian random vectors subject to constraints and relies on the fact that the value of Y at a given data location, conditioned by the values at the other data locations, has a Gaussian distribution whose mean and variance can be calculated by solving a kriging system (Geweke 1991; Freulon and de Fouquet 1993; Freulon 1994; Wilhelm and Manjunath 2011). In practice, when the data are numerous, kriging is performed with a moving search neighborhood rather than with a unique search neighborhood, which may make the simulated vector not follow the desired distribution. These problems are investigated in the following sections, by considering first the case of nonconditional simulation (Sect. 2), then the case of conditional simulation (Sect. 3).

2 Nonconditional Simulation

In this section, it is of interest to simulate a Gaussian random field with zero mean, unit variance, and known spatial correlation structure at a set of n locations $\mathbf{x}_1, \ldots, \mathbf{x}_n$ of $\mathbb{R}^d$. Put another way, one wants to simulate a Gaussian random vector $\mathbf{Y} = (Y_1, \ldots, Y_n)^T$ with zero mean and variance-covariance matrix $\mathbf{C} = [C_{ij}]_{i,j=1,\ldots,n}$ having unit diagonal entries

$$\forall i \in \{1, \ldots, n\}, \quad C_{ii} = 1. \tag{1}$$

2.1 Using a Unique Search Neighborhood

Y can be simulated iteratively by Gibbs sampling: At each iteration, one of the vector components is selected and updated according to its distribution conditional to the other components. Specifically, the algorithm is as follows.

Algorithm 1 (Traditional Gibbs Sampler, Unique Search Neighborhood)

(1) Initialize the simulation by an arbitrary vector of size n, for example, $\mathbf{Y}^{(0)} = (0, \ldots, 0)^T$.

(2) For $k = 1, 2, \ldots, K$:

    (a) Select an index $i \in \{1, \ldots, n\}$. Any selection strategy can be considered (regular selection, pure random selection, etc.), provided that all the indices in $\{1, \ldots, n\}$ are selected infinitely often as K tends to infinity (Amit and Grenander 1991; Roberts and Sahu 1997; Levine and Casella 2006). This condition will be assumed to be satisfied in all the algorithms presented in this work.

    (b) Set $\mathbf{Y}^{(k)} = \mathbf{Y}^{(k-1)}$, except for the ith component that is replaced by

$$Y_i^{(k)} = \sum_{j \neq i} \lambda_{j,i} Y_j^{(k-1)} + \sigma_i U^{(k)}, \tag{2}$$

    where $\lambda_{j,i}$ is the simple kriging weight assigned to Y_j when estimating Y_i, σ_i^2 is the associated simple kriging variance and $U^{(k)}$ is a standard Gaussian random variable independent of $\{Y_j^{(k-1)} : j \neq i\}$.

(3) Return $\mathbf{Y}^{(K)}$.

The sequence of simulated vectors $\{\mathbf{Y}^{(k)} : k \in \mathbb{N}\}$ forms a Markov chain whose state space is $\mathbb{R}^n$. The chain is aperiodic and irreducible, that is, any state can be reached from any other state by a finite number of steps. Furthermore, the target Gaussian distribution is invariant for the transition matrix of the chain, that is, if $\mathbf{Y}^{(k-1)}$ is a Gaussian random vector with zero mean and variance-covariance matrix $\mathbf{C}$, then so is $\mathbf{Y}^{(k)}$. Altogether, these three properties (irreducibility, aperiodicity, and existence of an invariant distribution) imply that the chain converges in distribution to a Gaussian random vector with zero mean and variance-covariance matrix $\mathbf{C}$ (Tierney 1994; Roberts 1996; Lantuéjoul 2002). In practice, the chain is stopped after a finite number K of iterations, resulting in a random vector $\mathbf{Y}^{(K)}$ whose distribution may differ from the target distribution, although the difference (measured by the total variation distance between distributions) becomes negligible if K is large enough. Several practical criteria to determine how large K must be have been proposed in the literature (Lantuéjoul 2002; Emery 2007a, 2008; Armstrong et al. 2011).

2.2 Using a Moving Search Neighborhood

If n is greater than a few thousands, calculating the kriging weights and kriging variances becomes computationally prohibitive. To circumvent this issue, one option is to apply kriging with a moving search neighborhood and to modify the algorithm as follows (Freulon and de Fouquet 1993).

Algorithm 2 (Traditional Gibbs Sampler, Moving Search Neighborhood)

(1) Initialize the simulation by an arbitrary vector of size n, for example, $\mathbf{Y}^{(0)} = (0, \ldots, 0)^T$.

(2) For $k = 1, 2, \ldots, K$:

    (a) Select an index $i \in \{1, \ldots, n\}$.

    (b) Find the subset J of indices such that $\{\mathbf{x}_j : j \in J\}$ belong to a neighborhood centered on $\mathbf{x}_i$ with given characteristics (maximum distance to $\mathbf{x}_i$ along each direction, splitting into angular sectors, maximum number of locations within each sector, etc.).

    (c) Set $\mathbf{Y}^{(k)} = \mathbf{Y}^{(k-1)}$, except for the ith component that is replaced by

$$Y_i^{(k)} = \sum_{j \neq i} \tilde{\lambda}_{j,i} Y_j^{(k-1)} + \tilde{\sigma}_i U^{(k)}, \tag{3}$$

    where $\tilde{\lambda}_{j,i}$ and $\tilde{\sigma}_i^2$ are the simple kriging weights and simple kriging variance obtained when estimating Y_i from $\{Y_j : j \in J, j \neq i\}$, and $U^{(k)}$ is a standard Gaussian random variable independent of $\{Y_j^{(k-1)} : j \neq i\}$. In Eq. (3), the convention $\tilde{\lambda}_{j,i} = 0$ has been used for $j \notin J$.

(3) Return $\mathbf{Y}^{(K)}$.

As in Algorithm 1, the sequence of simulated vectors $\{\mathbf{Y}^{(k)} : k \in \mathbb{N}\}$ forms a irreducible and aperiodic Markov chain whose state space is $\mathbb{R}^n$. Whether or not this chain converges in distribution depends on whether or not there exists a distribution that is invariant for the transition matrix of the chain (Tierney 1994; Lantuéjoul 2002). To find out this invariant distribution, first note that at each iteration the random vector $\mathbf{Y}^{(k)}$ is a zero-mean Gaussian random vector. Hence, if it exists, the invariant distribution should be Gaussian with zero mean. Let $\tilde{\mathbf{C}} = [\tilde{C}_{ij}]_{i,j=1,\ldots,n}$ be its variance-covariance matrix. Then, if $\mathbf{Y}^{(k-1)}$ is a Gaussian random vector with zero mean and variance-covariance matrix $\tilde{\mathbf{C}}$, so is $\mathbf{Y}^{(k)}$. Using Eq. (3), this implies the following identities

$$\forall i \in \{1, \ldots, n\}, \ \forall \ell \in \{1, \ldots, n\} - \{i\}, \quad \tilde{C}_{i\ell} = \mathrm{cov}\{Y_i^{(k)}, Y_\ell^{(k)}\}$$
$$= \sum_{j \neq i} \tilde{\lambda}_{j,i} \mathrm{cov}\{Y_j^{(k-1)}, Y_\ell^{(k-1)}\}$$
$$= \sum_{j \neq i} \tilde{\lambda}_{j,i} \tilde{C}_{j\ell}, \tag{4}$$

and

$$\forall i \in \{1, \ldots, n\}, \quad \tilde{C}_{ii} = \text{cov}\{Y_i^{(k)}, Y_i^{(k)}\}$$

$$= \sum_{j \neq i} \sum_{\ell \neq i} \tilde{\lambda}_{j,i} \tilde{\lambda}_{\ell,i} \text{cov}\{Y_j^{(k-1)}, Y_\ell^{(k-1)}\} + \tilde{\sigma}_i^2 \text{var}\{U^{(k)}\}$$

$$= \sum_{j \neq i} \sum_{\ell \neq i} \tilde{\lambda}_{j,i} \tilde{\lambda}_{\ell,i} \tilde{C}_{j\ell} + \tilde{\sigma}_i^2. \tag{5}$$

Using Eq. (4), Eq. (5) becomes

$$\forall i \in \{1, \ldots, n\}, \quad \tilde{C}_{ii} = \sum_{\ell \neq i} \tilde{\lambda}_{\ell,i} \tilde{C}_{i\ell} + \tilde{\sigma}_i^2. \tag{6}$$

Equations (4) and (6) can be rewritten in matrix form

$$\tilde{\mathbf{B}} \tilde{\mathbf{C}} = \mathbf{I}, \tag{7}$$

with $\mathbf{I}$ the identity matrix of size $n \times n$ and

$$\tilde{\mathbf{B}} = \begin{pmatrix} \frac{1}{\tilde{\sigma}_1^2} & \frac{-\tilde{\lambda}_{2,1}}{\tilde{\sigma}_1^2} & \cdots & \frac{-\tilde{\lambda}_{n-1,1}}{\tilde{\sigma}_1^2} & \frac{-\tilde{\lambda}_{n,1}}{\tilde{\sigma}_1^2} \\ \frac{-\tilde{\lambda}_{1,2}}{\tilde{\sigma}_2^2} & \frac{1}{\tilde{\sigma}_2^2} & & \frac{-\tilde{\lambda}_{n-1,2}}{\tilde{\sigma}_2^2} & \frac{-\tilde{\lambda}_{n,2}}{\tilde{\sigma}_2^2} \\ \vdots & & \ddots & & \vdots \\ \frac{-\tilde{\lambda}_{1,n-1}}{\tilde{\sigma}_{n-1}^2} & \frac{-\tilde{\lambda}_{2,n-1}}{\tilde{\sigma}_{n-1}^2} & & \frac{1}{\tilde{\sigma}_{n-1}^2} & \frac{-\tilde{\lambda}_{n,n-1}}{\tilde{\sigma}_{n-1}^2} \\ \frac{-\tilde{\lambda}_{1,n}}{\tilde{\sigma}_n^2} & \frac{-\tilde{\lambda}_{2,n}}{\tilde{\sigma}_n^2} & \cdots & \frac{-\tilde{\lambda}_{n-1,n}}{\tilde{\sigma}_n^2} & \frac{1}{\tilde{\sigma}_n^2} \end{pmatrix}. \tag{8}$$

Accordingly, if it exists, the Gaussian distribution with zero mean and variance-covariance matrix $\tilde{\mathbf{C}} = \tilde{\mathbf{B}}^{-1}$ is invariant for the transition matrix and, therefore, is the limit distribution of the chain. In the case of a unique search neighborhood (Algorithm 1), it is known (Dubrule 1983) that $\tilde{\mathbf{B}}$ is the inverse of $\mathbf{C}$, thus the chain converges in distribution to a Gaussian random vector with zero mean and variance-covariance matrix $\mathbf{C}$. However, in the case of a moving search neighborhood (Algorithm 2), the chain may not converge in distribution because $\tilde{\mathbf{C}}$ may not be a valid variance-covariance matrix, that is, a symmetric positive semidefinite matrix. The conditions for the existence of a limit distribution are:

(1) $\tilde{\mathbf{B}}$ is a symmetric matrix, that is,

$$\forall i, j \in \{1, \ldots, n\}, \quad \tilde{\lambda}_{j,i} \tilde{\sigma}_j^2 = \tilde{\lambda}_{i,j} \tilde{\sigma}_i^2. \tag{9}$$

(2) The eigenvalues of $\tilde{\mathbf{B}}$ are nonnegative.

2.3 Numerical Experiments

To study the effect of a moving search neighborhood, numerical experiments have been conducted with the following parameters:

Table 1 Properties of matrix $\tilde{\mathbf{B}}$ for each covariance model and moving search neighborhood

Covariance model	Neighborhood radius	Relative Frobenius norm of $\tilde{\mathbf{B}} - \tilde{\mathbf{B}}^T$	Minimum real eigenvalue of $\tilde{\mathbf{B}}$	Number of nonreal eigenvalues	Maximum imaginary part of complex eigenvalues
Spherical	5	0.0016	−0.0080	6	8.22×10^{-15}
Spherical	15	0.0053	−0.0200	14	0.0033
Spherical	25	0.0007	0.0073	4	4.16×10^{-15}
Cubic	5	0.0010	−0.0232	38	3.33×10^{-11}
Cubic	15	0.0095	−1.4919	120	0.6620
Cubic	25	0.0015	0.0072	68	0.0038

- The simulated random vector **Y** is a realization of a stationary Gaussian random field with an isotropic spherical covariance (Case 1) or an isotropic cubic covariance (Case 2), in both cases with a range of 15 units, over a regular two-dimensional grid with 50×50 nodes and a mesh of 1×1.
- The grid nodes are visited according to a random permutation strategy (Galli and Gao 2001) and the Gibbs sampler is stopped after 2,500,000 iterations, which corresponds to 1,000 permutations.
- The moving search neighborhood used for calculating the kriging weights and kriging variances is a disc centered at the target node. Three disc radii are considered (5, 15, and 25 units), for which the number of grid nodes found in the neighborhood can reach 80, 708, and 1960, respectively.

Figures 1 and 2 present the map of the final realization and a plot of the minimum and maximum simulated values as a function of the iteration number. When using a moving search neighborhood of 5 or 25 units, the apparent spatial variations agree with the covariance models (continuous variations for the spherical covariance, smooth variations for the cubic covariance), while the minimum and maximum simulated values fluctuate in a range that is consistent with what is expected from a standard Gaussian random field. However, things go wrong with a moving search neighborhood of 15 units: The map shows exaggeratedly smooth spatial variations or artifacts (Figs. 1(b) and 2(b)), and the range of the extreme simulated values becomes excessively large when the number of iterations increases (Figs. 1(e) and 2(e)). To explain these results, it is interesting to have a look at the matrix $\tilde{\mathbf{B}}$ defined in Eq. (8). Table 1 summarizes some properties of this matrix (minimum real eigenvalue, number of complex nonreal eigenvalues, maximum imaginary part of the complex eigenvalues, Frobenius norm of $\tilde{\mathbf{B}} - \tilde{\mathbf{B}}^T$ relative to the Frobenius norm of $\tilde{\mathbf{B}}$) in each case. These properties allow checking whether or not $\tilde{\mathbf{B}}$ satisfies the conditions for the existence of a limit distribution and, if not, how distant is this matrix from a symmetric positive semidefinite matrix. Not surprisingly, the case with a moving search neighborhood of radius 15 is the one for which $\tilde{\mathbf{B}}$ is far from being symmetric (higher relative Frobenius norm of $\tilde{\mathbf{B}} - \tilde{\mathbf{B}}^T$, larger number of nonreal eigenvalues) and has negative real eigenvalues with larger absolute values. Note that the requirement for a symmetric matrix (Eq. (9)) are never satisfied, thus, in all the cases, the sequences of simulated vectors do not converge
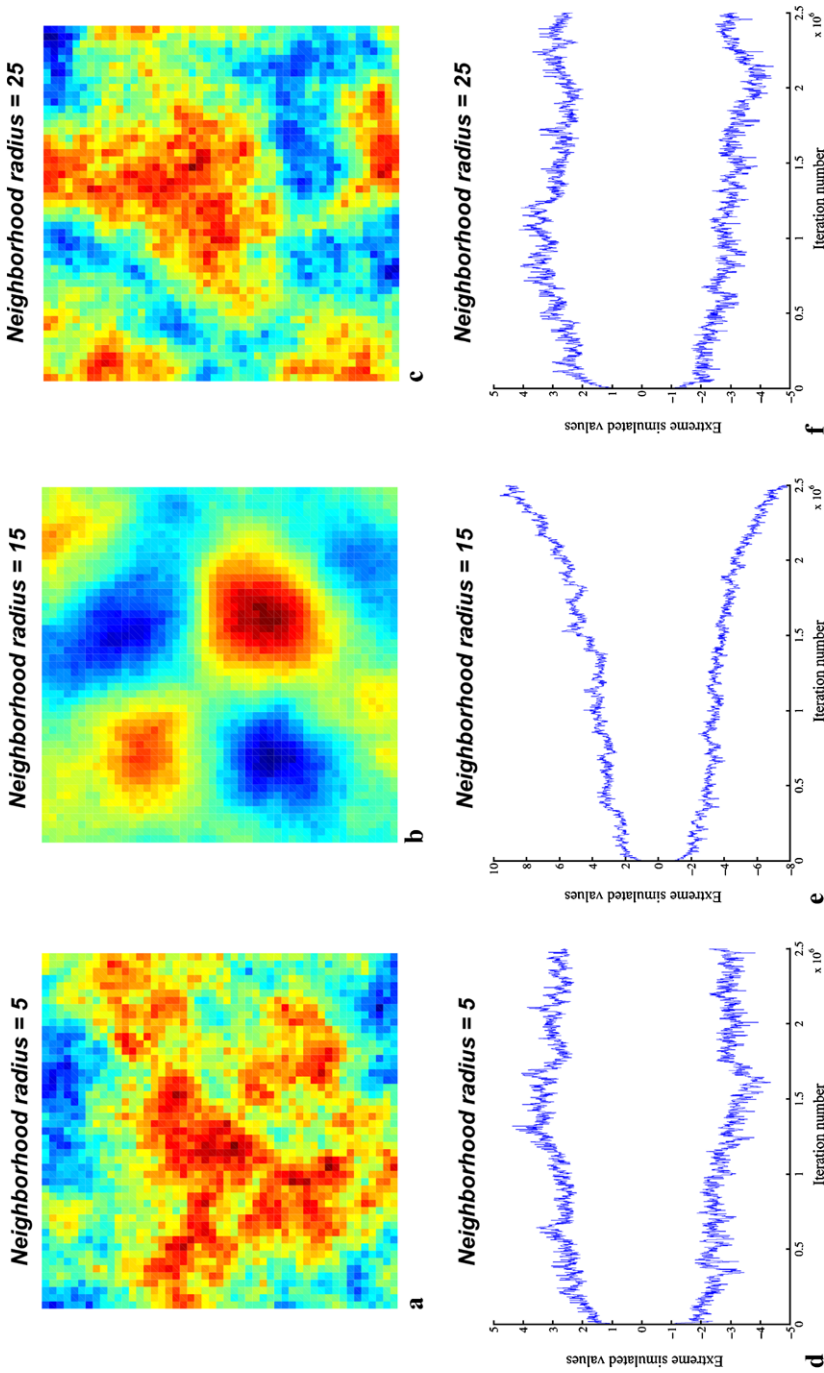
Fig. 1 (**a**), (**b**), (**c**), maps of realizations obtained after 2,500,000 iterations by using three moving search neighborhoods; (**d**), (**e**), (**f**), minimum and maximum simulated values as a function of the iteration number (Case 1: isotropic spherical model)
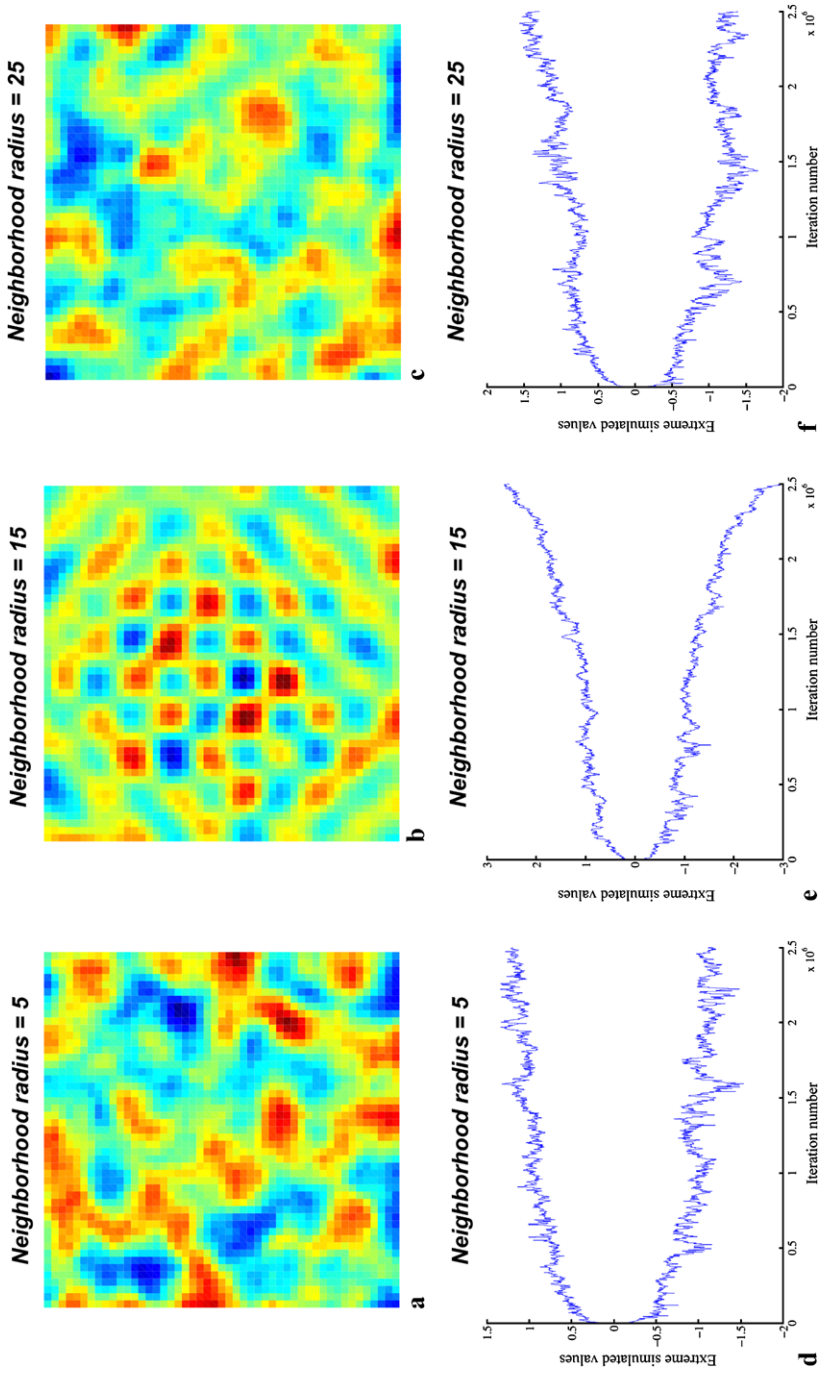
Fig. 2 (**a**), (**b**), (**c**), maps of realizations obtained after 2,500,000 iterations by using three moving search neighborhoods; (**d**), (**e**), (**f**), minimum and maximum simulated values as a function of the iteration number (Case 2: isotropic cubic model)

in distribution. Accordingly, even with the moving search neighborhoods of radius 5 or 25, the observed spatial variations or the range of simulated values may not agree with the target distribution if the algorithm is stopped after more iterations.

2.4 An Alternative for a Unique Search Neighborhood

To avoid using a moving search neighborhood, an alternative has been proposed by Galli and Gao (2001) and further examined by Lantuéjoul and Desassis (2012) and Arroyo et al. (2012). Let $\mathbf{B} = \mathbf{C}^{-1}$ and $\mathbf{X} = \mathbf{B}\mathbf{Y}$ (equivalently, $\mathbf{Y} = \mathbf{C}\mathbf{X}$). Instead of simulating $\mathbf{Y}$, the idea is to simulate $\mathbf{X}$, which is a Gaussian random vector with zero mean and variance-covariance matrix $\mathbf{B}$, and to set $\mathbf{Y} = \mathbf{C}\mathbf{X}$ once the simulation of $\mathbf{X}$ is completed. As the determination of kriging weights and kriging variances relies on the inverse of the variance-covariance matrix (i.e., $\mathbf{B}^{-1} = \mathbf{C}$), it is no longer necessary to invert $\mathbf{C}$, so that a unique search neighborhood can be considered. The algorithm is therefore as follows.

Algorithm 3 (Gibbs Sampler on $\mathbf{X}$, Unique Search Neighborhood)

(1) Initialize the simulation by an arbitrary vector of size n, for example, $\mathbf{X}^{(0)} = (0, \ldots, 0)^T$.
(2) For $k = 1, 2, \ldots, K$:

   (a) Select an index $i \in \{1, \ldots, n\}$.
   (b) Set $\mathbf{X}^{(k)} = \mathbf{X}^{(k-1)}$, except for the ith component that is replaced by

$$X_i^{(k)} = \sum_{j \neq i} \varpi_{j,i} X_j^{(k-1)} + s_i V^{(k)}, \tag{10}$$

   where $\varpi_{j,i}$ is the simple kriging weight assigned to X_j when estimating X_i, s_i^2 is the associated simple kriging variance and $V^{(k)}$ is a standard Gaussian random variable independent of $\{X_j^{(k-1)} : j \neq i\}$. Letting C_{ij} be the entry of $\mathbf{C}$ at row i and column j, then (Dubrule 1983)

$$s_i^2 = \frac{1}{C_{ii}}, \tag{11}$$

$$\varpi_{j,i} = -\frac{C_{ij}}{C_{ii}}. \tag{12}$$

   Let $\mathbf{Y}^{(k)} = \mathbf{C}\mathbf{X}^{(k)}$. Recalling that $C_{ii} = 1$ (Eq. (1)), Eq. (10) can be rewritten as

$$X_i^{(k)} = X_i^{(k-1)} - Y_i^{(k-1)} + V^{(k)}. \tag{13}$$

(3) Return $\mathbf{Y}^{(K)} = \mathbf{C}\mathbf{X}^{(K)}$.

To speed up calculations, one can rewrite Algorithm 3 to directly simulate $\mathbf{Y}$.

Algorithm 4 (Gibbs Sampler on $\mathbf{Y}$, Unique Search Neighborhood)

(1) Initialize the simulation by an arbitrary vector of size n, for example, $\mathbf{Y}^{(0)} = (0, \ldots, 0)^T$.

(2) For $k = 1, 2, \ldots, K$:

  (a) Select an index $i \in \{1, \ldots, n\}$.
  (b) Set $\mathbf{Y}^{(k)} = \mathbf{Y}^{(k-1)} + \delta \mathbf{C}_i$, where $\mathbf{C}_i$ is the ith column of $\mathbf{C}$, $\delta = -Y_i^{(k-1)} + V^{(k)}$ and $V^{(k)}$ is a standard Gaussian random variable independent of $\mathbf{Y}^{(k-1)}$.

(3) Return $\mathbf{Y}^{(K)}$.

This algorithm has been recently proposed by Lantuéjoul and Desassis (2012) and Arroyo et al. (2012). Using different approaches, these authors also established a variant in which several indices are selected in step 2(a).

3 Simulation Conditioned to Inequality Constraints

In the scope of truncated Gaussian and plurigaussian simulation, it is necessary to simulate the Gaussian random vector $\mathbf{Y}$ subject to inequality constraints on its components

$$\forall i \in \{1, \ldots, n\}, \quad a_i < Y_i < b_i, \tag{14}$$

where the bounds $a_1, \ldots, a_n, b_1, \ldots, b_n$ are real numbers or infinite. Two approaches (simulated annealing and restriction of the transition matrix) will be explored.

3.1 Simulated Annealing

Simulated annealing is an iterative procedure that can be used to simulate random vectors subject to complex conditioning constraints (Lantuéjoul 2002; Chilès and Delfiner 2012). The algorithm relies on three components: (i) a Markov chain that has the nonconditional distribution (here, a Gaussian with zero mean and variance-covariance matrix $\mathbf{C}$) as its limit distribution, (ii) an objective function $O(.)$ that measures the discrepancy between the simulated vector and the conditioning constraints, and (iii) a cooling schedule that defines a temperature t_k at the kth iteration of the algorithm. For instance, the objective function can be defined as follows

$$O(\mathbf{Y}) = \sum_{i=1}^{n} \left(\max\{a_i - Y_i, 0\} + \max\{Y_i - b_i, 0\} \right), \tag{15}$$

so that $O(\mathbf{Y}) \geq 0$, with the equality $O(\mathbf{Y}) = 0$ if all the conditioning constraints are satisfied. As for the temperature, one example is to consider a cooling schedule of the form

$$\forall k > 0, \quad t_k = t_0 \alpha^k, \tag{16}$$

with given values of t_0 in $]0, +\infty[$ and α in $]0, 1[$.

To avoid convergence problems due to a moving search neighborhood, the Markov chain considered for the simulated annealing procedure will be the Gibbs sampler proposed in Sect. 2.4 (Algorithm 4).

Algorithm 5 (Gibbs Sampler Combined with Simulated Annealing)

(1) Initialize the simulation by an arbitrary vector $\mathbf{Y}^{(0)}$ of size n, for example, a vector of zeros or a nonconditional realization of $\mathbf{Y}$.
(2) For $k = 1, 2, \ldots, K$:

    (a) Select an index $i \in \{1, \ldots, n\}$.
    (b) Set $\mathbf{Y}' = \mathbf{Y}^{(k-1)} + \delta \mathbf{C}_i$, where $\mathbf{C}_i$ is the ith column of $\mathbf{C}$, $\delta = -Y_i^{(k-1)} + V^{(k)}$ and $V^{(k)}$ is a standard Gaussian random variable independent of $\mathbf{Y}^{(k-1)}$.
    (c) Simulate a random variable U uniformly distributed in $[0, 1]$.
    (d) Calculate the current temperature t_k, according to the chosen cooling schedule.
    (e) If $U < \exp\{\frac{O(\mathbf{Y}^{(k-1)}) - O(\mathbf{Y}')}{t_k}\}$, set $\mathbf{Y}^{(k)} = \mathbf{Y}'$. Otherwise, set $\mathbf{Y}^{(k)} = \mathbf{Y}^{(k-1)}$.

(3) Return $\mathbf{Y}^{(K)}$.

One issue with the above algorithm is that the rate of convergence to the target conditional distribution may be slow, depending on the chosen initial state and cooling schedule. If the algorithm is stopped after a finite number (K) of iterations, one obtains a random vector $\mathbf{Y}^{(K)}$ that may not satisfy all the conditioning constraints (Eq. (14)). An illustration of this situation will be presented at Sect. 3.3.

3.2 Restriction of the Transition Matrix

The idea of restricting the transition matrix of the Gibbs sampler has been proposed by several authors (Geweke 1991; Freulon and de Fouquet 1993; Kotecha and Djuric 1999; Wilhelm and Manjunath 2011), who modified Algorithm 1 to forbid any transition to a state that does not satisfy the conditioning constraints (Eq. (14)). In practice, when the number n of vector components is large, a unique search neighborhood is impractical and a moving search neighborhood is used instead (Freulon and de Fouquet 1993), but as seen in Sect. 2.2, this is likely to make the sequence of simulated vectors not converge in distribution. To avoid such problems, rather than Algorithm 1, one can consider Algorithm 3 and restrict its transition matrix, as explained next.

Algorithm 6 (Gibbs Sampler on $\mathbf{X}$, with Restriction of the Transition Matrix)

(1) Initialize the simulation by a vector $\mathbf{X}^{(0)}$ that satisfies the conditioning constraints. This requires finding a solution to the following system of linear inequalities

$$\forall i \in \{1, \ldots, n\}, \quad a_i < \sum_{j=1}^{n} C_{ij} X_j < b_i. \tag{17}$$

The set of solutions of the above system is an open convex region of $\mathbb{R}^n$, delimited by the hyperplanes defined by the equations $\sum_{j=1}^{n} C_{ij} X_j = a_i$ and $\sum_{j=1}^{n} C_{\ell j} X_j = b_\ell$ for all the indices i and ℓ such that a_i and b_ℓ are finite.

(2) For $k = 1, 2, \ldots, K$:

    (a) Select an index $i \in \{1, \ldots, n\}$.

    (b) Set $\mathbf{X}^{(k)} = \mathbf{X}^{(k-1)}$, except for the ith component that is replaced by

$$X_i^{(k)} = \sum_{j \neq i} \varpi_{j,i} X_j^{(k-1)} + V^{(k)}, \tag{18}$$

where $\varpi_{j,i}$ is the simple kriging weight assigned to X_j when estimating X_i and $V^{(k)}$ is a standard Gaussian random variable independent of $\{X_j^{(k-1)} : j \neq i\}$ satisfying the constraints in Eq. (17). $V^{(k)}$ has a truncated Gaussian distribution whose range of values is an open interval bounded by (Eqs. (17)–(18))

$$\max_{p \in \{1, \ldots, n\}: C_{pi} \neq 0} \left\{ \min \left(\frac{a_p - \beta_{pi}}{C_{pi}}, \frac{b_p - \beta_{pi}}{C_{pi}} \right) \right\} \tag{19}$$

and

$$\min_{p \in \{1, \ldots, n\}: C_{pi} \neq 0} \left\{ \max \left(\frac{a_p - \beta_{pi}}{C_{pi}}, \frac{b_p - \beta_{pi}}{C_{pi}} \right) \right\}, \tag{20}$$

with $\beta_{pi} = \sum_{j \neq i} (C_{pj} + C_{pi} \varpi_{j,i}) X_j^{(k-1)}$.

    (c) Set $\mathbf{Y}^{(k)} = \mathbf{C} \mathbf{X}^{(k)}$.

(3) Return $\mathbf{Y}^{(K)}$.

The sequence of simulated vectors $\{\mathbf{X}^{(k)} : k \in \mathbb{N}\}$ forms a Markov chain that is aperiodic and irreducible: Irrespective of the current state, any point of the convex open region solution of Eq. (17) can be reached in a finite number of iterations. Furthermore, let $\mathbf{X} = \mathbf{B}\mathbf{Y}$ be a Gaussian random vector with zero mean and variance-covariance matrix $\mathbf{B} = \mathbf{C}^{-1}$, and assume that $\mathbf{X}^{(k-1)}$ is a random vector with the distribution of $\mathbf{X}$ conditioned by the inequality constraints in Eq. (17). Because the transition matrix in step (2) is identical to the transition matrix of Algorithm 3, except for the restrictions Eqs. (19) and (20) that make $\mathbf{X}^{(k)}$ satisfy Eq. (17), the distribution of $\mathbf{X}^{(k)}$ is the same as that of $\mathbf{X}^{(k-1)}$, that is, this distribution is invariant for the transition matrix. Hence, the conditions for convergence (irreducibility, aperiodicity, and existence of an invariant distribution) are satisfied (Lantuéjoul 2002): The chain $\{\mathbf{X}^{(k)} : k \in \mathbb{N}\}$ converges in distribution to a Gaussian random vector with zero mean and variance-covariance matrix $\mathbf{B}$ conditioned by the constraints of Eq. (17). Equivalently, the chain $\{\mathbf{Y}^{(k)} : k \in \mathbb{N}\}$ converges in distribution to a Gaussian random vector with zero mean and variance-covariance matrix $\mathbf{C}$ conditioned by the constraints of Eq. (14). In practice, the main difficulty of Algorithm 6 is the initialization step, for which one may need to solve a large system of linear inequalities. To avoid this difficulty, it is convenient to rewrite this algorithm to directly simulate $\mathbf{Y}$, as proposed next.

Algorithm 7 (Gibbs Sampler on **Y**, with Restriction of the Transition Matrix)

(1) Initialize the simulation by a vector $\mathbf{Y}^{(0)}$ that satisfies the conditioning the constraints of Eq. (14).
(2) For $k = 1, 2, \ldots, K$:

    (a) Select an index $i \in \{1, \ldots, n\}$.
    (b) Set $\mathbf{Y}^{(k)} = \mathbf{Y}^{(k-1)} + \delta \mathbf{C}_i$, where $\mathbf{C}_i$ is the ith column of $\mathbf{C}$, $\delta = -Y_i^{(k-1)} + V^{(k)}$ and $V^{(k)}$ is a truncated Gaussian random variable independent of $\mathbf{Y}^{(k-1)}$, such that the conditioning constraints of Eq. (14) are satisfied. The bounds for $V^{(k)}$ are

$$\max_{j \in \{1,\ldots,n\}: C_{ij} \neq 0} \left\{ \min \left(\frac{a_j - \gamma_{ij}}{C_{ij}}, \frac{b_j - \gamma_{ij}}{C_{ij}} \right) \right\} \tag{21}$$

and

$$\min_{j \in \{1,\ldots,n\}: C_{ij} \neq 0} \left\{ \max \left(\frac{a_j - \gamma_{ij}}{C_{ij}}, \frac{b_j - \gamma_{ij}}{C_{ij}} \right) \right\}, \tag{22}$$

with $\gamma_{ij} = Y_j^{(k-1)} - C_{ij} Y_i^{(k-1)}$.

(3) Return $\mathbf{Y}^{(K)}$.

As this algorithm is formally equivalent to Algorithm 6, the chain $\{\mathbf{Y}^{(k)} : k \in \mathbb{N}\}$ converges in distribution to a Gaussian random vector with zero mean and variance-covariance matrix $\mathbf{C}$ conditioned by the constraints of Eq. (14).

3.3 Numerical Experiments

We consider the two cases presented in Sect. 2.3 (simulation of a Gaussian random field with an isotropic spherical or cubic covariance model on a 50×50 grid) and now focus on simulation under inequality constraints. For the sake of simplicity, these constraints will be of the form

$$\forall i \in \{1, \ldots, n\}, \quad Y_i < 0 \quad \text{or} \quad Y_i > 0. \tag{23}$$

Two simulation algorithms are used: simulated annealing (Algorithm 5) and restriction of the transition matrix (Algorithm 7). In simulated annealing, the initial state $\mathbf{Y}^{(0)}$ is a nonconditional realization of $\mathbf{Y}$ obtained by Algorithm 4. The objective function and the cooling schedule presented in Eqs. (15) and (16) have been used, with an initial temperature $t_0 = 10{,}000$ and a cooling parameter $\alpha = 0.999999$. For the restriction of the transition matrix approach, one option to define the initial state is to simulate each vector component Y_i via an acceptance-rejection method, by drawing a standard Gaussian random variable as many times as necessary until the simulated value satisfies the conditioning constraint (Eq. (23)). However, since the vector components are simulated independently one from another, this leads to an initial state with no spatial structure and a considerable number K of iterations are necessary to get a final random vector with the desired correlation matrix. To obtain

a spatially structured initial state, another option is to run Algorithm 2 with a restriction of the transition matrix, that is, the distribution of $U^{(k)}$ in Eq. (3) is conditioned to the constraints given in Eq. (23). This option has been chosen in the experiments shown below, using a search radius of 5 units and a total of 2,500,000 iterations. For each covariance model (spherical and cubic), the experiment consists of the following steps:

(1) Construct a nonconditional simulation of **Y**. This step has been achieved by using the algorithm proposed by Arroyo et al. (2012), a variant of Algorithm 4 in which several vector components (in the present case, four components associated with adjacent grid nodes) are selected at each iteration, instead of a single component.
(2) Using the realization of step (1), define the conditioning constraints, depending on the signs of the simulated values (Eq. (23)).
(3) Use these constraints to construct a conditional simulation of **Y**, using Algorithm 5 (simulated annealing) or Algorithm 7 (restriction of the transition matrix).

The overall procedure (steps 1–3) is repeated 50 times, with different seeds for random number generation each time. To ensure that the distribution of the simulated vector is close to the target conditional distribution, a large number of iterations ($K = 25{,}000{,}000$) has been chosen for step (2) of Algorithms 5 and 7. This can be checked by comparing the experimental covariance of the conditional realizations with the prior covariance model: on average, the experimental covariance practically matches the prior model (Fig. 3). Reducing the number of iterations (for instance, to $K = 5{,}000{,}000$) significantly decreases the computation time but may lead to a mismatch in the covariance reproduction, as illustrated in Fig. 4 for the restriction of the transition matrix algorithm applied to the simulation of a random vector with a cubic covariance. Examples of nonconditional and conditional realizations are presented in Fig. 5 for the restriction of the transition matrix (Algorithm 7), corroborating that the conditional realizations satisfy the interval constraints and have the same spatial correlation structure as the nonconditional realizations. A similar figure can be obtained for simulated annealing (Algorithm 5), except that some conditioning data are not satisfied as the algorithm is stopped before the objective function could reach a zero value (Fig. 6; Table 2). The number of conditioning data that are not satisfied could be decreased further at the cost of an increase of the number of iterations and of the associated computation time. From this viewpoint, the restriction of the transition matrix approach is preferable to simulated annealing.

4 Conclusions

The Gibbs sampler is an iterative algorithm that allows constructing a sequence of random vectors converging in distribution to a Gaussian random vector with zero mean and given variance-covariance matrix. However, convergence may no longer take place if a moving search neighborhood is used in the iteration stage. To avoid such problems, an alternative version of the algorithm has been presented, consisting of simulating a Gaussian random vector with the inverse covariance matrix. This new
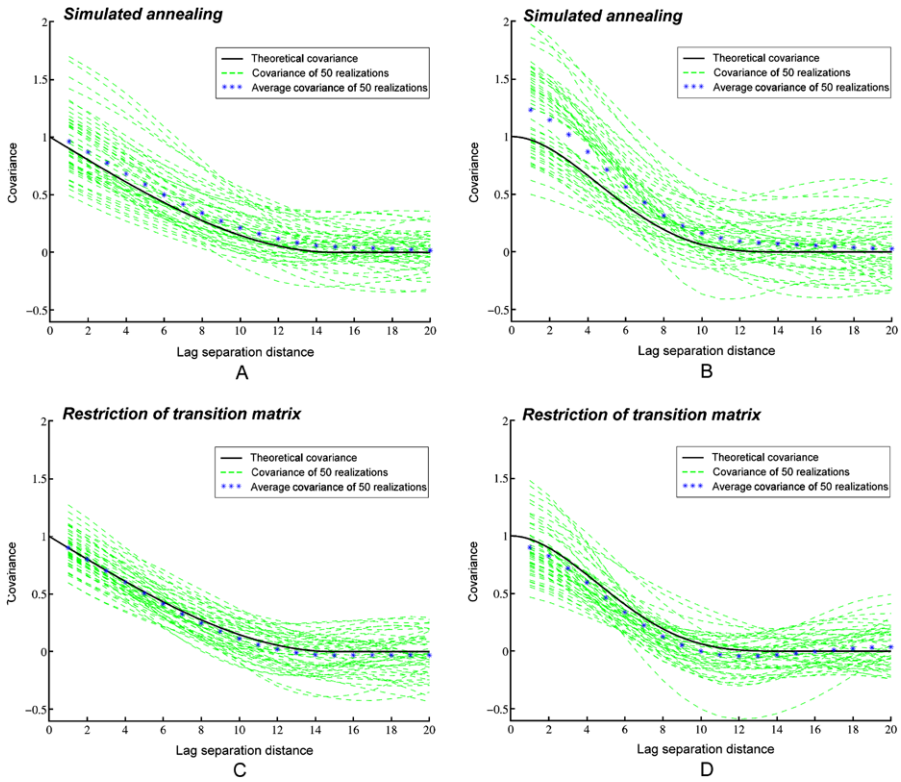
Fig. 3 Prior covariance models (*solid lines*), experimental covariances (*dashed lines*) and average experimental covariances (*stars*) for conditional realizations obtained via simulated annealing (**a**), (**b**) and restriction of transition matrix (**c**), (**d**) with $K = 25,000,000$ iterations

Fig. 4 Prior covariance model (*solid line*), experimental covariances (*dashed lines*) and average experimental covariances (*stars*) for conditional realizations obtained via restriction of transition matrix with $K = 5,000,000$ iterations
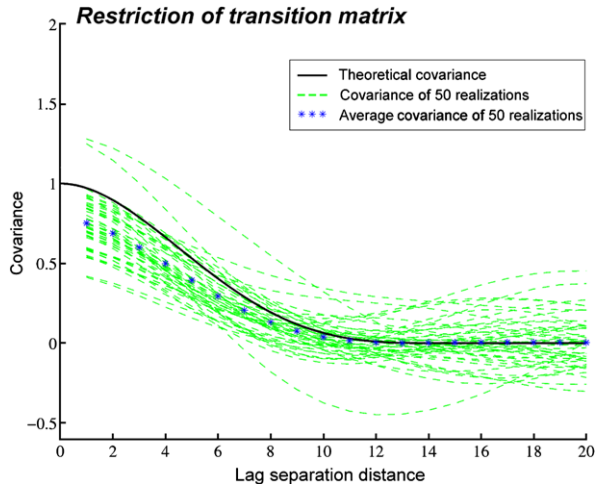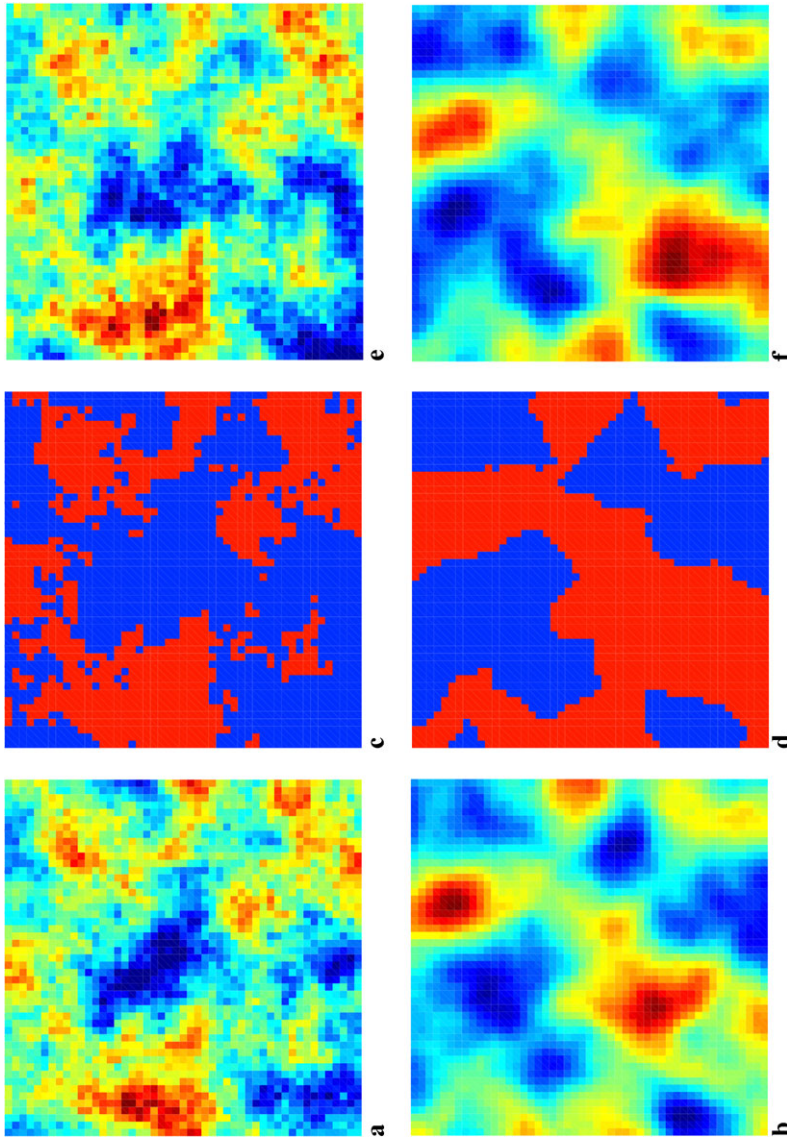
Fig. 5 (**a**), (**b**), nonconditional realizations of Gaussian random fields; (**c**), (**d**), conditioning constraints (positive values in *red*, negative values in *blue*); (**e**), (**f**), examples of conditional realizations (restriction of transition matrix algorithm). Subfigures (**a**), (**c**), (**e**) correspond to spherical covariance model, and (**b**), (**d**), (**f**) to cubic covariance model
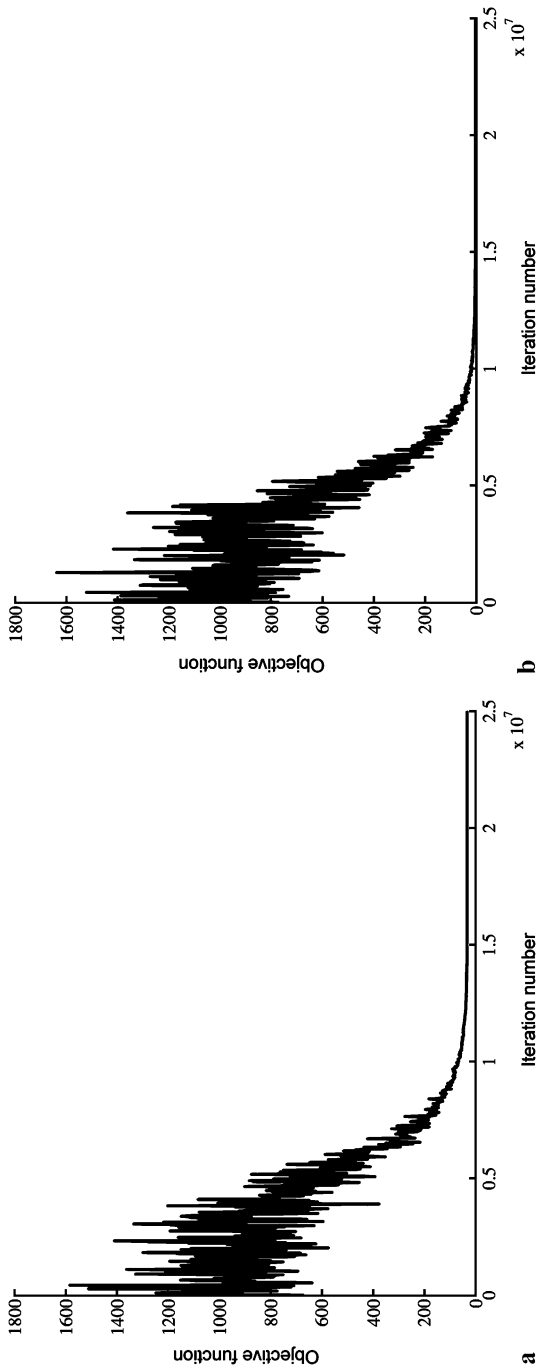
Fig. 6 Evolution of objective function with number of iterations for spherical (**a**) and cubic (**b**) covariance models (first realization)

Table 2 Number of conditioning constraints that are not satisfied after 25,000,000 iterations (statistics over 50 realizations)

	Minimum	Maximum	Mean	Mean absolute value
Spherical covariance	167	394	259.3	0.589
Cubic covariance	37	175	85.6	0.054

version can easily be performed with a unique search neighborhood. Furthermore, it lends itself to the simulation of a random vector conditioned to inequality constraints, by incorporating either a simulated annealing procedure or a restriction of the transition matrix in the iterative algorithm. In practice, simulated annealing only manages to approximately satisfy the conditioning constraints, whereas the restriction of the transition matrix can exactly satisfy these constraints. In both cases, a large number of iterations may be needed to accurately reproduce the target correlation matrix (for indication, the numerical experiments realized in this work have considered a number of iterations equal to ten thousand times the number of conditioning constraints). The number of iterations should be increased if the number of conditioning constraints increases or if these constraints are very restrictive (for example, if they consist of narrow intervals).

References

Amit Y, Grenander U (1991) Comparing sweep strategies for stochastic relaxation. J Multivar Anal 37(2):197–222

Armstrong M, Galli A, Beucher H, Le Loc'h G, Renard D, Doligez B, Eschard R, Geffroy F (2011) Plurigaussian simulations in geosciences. Springer, Berlin

Arroyo D, Emery X, Peláez M (2012) An enhanced Gibbs sampler algorithm for non-conditional simulation of Gaussian random vectors. Comput Geosci 46:138–148

Bárdossy A (2006) Copula-based geostatistical models for groundwater quality parameters. Water Resour Res 42:W11416

Chilès JP, Delfiner P (2012) Geostatistics: modeling spatial uncertainty. Wiley, New York

Dubrule O (1983) Cross validation of kriging in a unique neighborhood. Math Geol 15(6):687–699

Emery X (2005) Conditional simulation of random fields with bivariate gamma isofactorial distributions. Math Geol 37(4):419–445

Emery X (2007a) Using the Gibbs sampler for conditional simulation of Gaussian-based random fields. Comput Geosci 33(4):522–537

Emery X (2007b) Simulation of geological domains using the plurigaussian model: new developments and computer programs. Comput Geosci 33(9):1189–1201

Emery X (2008) Statistical tests for validating geostatistical simulation algorithms. Comput Geosci 34(11):1610–1620

Freulon X (1994) Conditional simulation of a Gaussian random vector with nonlinear and/or noisy observations. In: Armstrong M, Dowd PA (eds) Geostatistical simulations. Kluwer Academic, Dordrecht, pp 57–71

Freulon X, de Fouquet C (1993) Conditioning a Gaussian model with inequalities. In: Soares A (ed) Geostatistics Tróia'92. Kluwer Academic, Dordrecht, pp 201–212

Galli A, Gao H (2001) Rate of convergence of the Gibbs sampler in the Gaussian case. Math Geol 33(6):653–677

Geweke JF (1991) Efficient simulation from the multivariate normal and student-t distributions subject to linear constraints and the evaluation of constraint probabilities. In: Keramidas EM (ed) Computing science and statistics: proceedings of the twenty-third symposium on the interface. Interface Foundation of North America, Fairfax, pp 571–578

Journel AG (1986) Constrained interpolation and qualitative information—the soft kriging approach. Math Geol 18(3):269–286

Kotecha JH, Djuric PM (1999) Gibbs sampling approach for generation of truncated multivariate Gaussian random variables. In: Proceedings of the 1999 IEEE international conference on acoustics, speech, and signal processing. IEEE Comput Soc, Washington, pp 1757–1760

Lantuéjoul C (2002) Geostatistical simulation: models and algorithms. Springer, Berlin

Lantuéjoul C, Desassis N (2012) Simulation of a Gaussian random vector: a propagative version of the Gibbs sampler. In: Ninth international geostatistics Congress, Oslo. http://geostats2012.nr.no/pdfs/1747181.pdf. Accessed 2 July 2013

Le Loc'h G, Beucher H, Galli A, Doligez B (Heresim Group) (1994) Improvement in the truncated Gaussian method: combining several Gaussian functions. In: Omre H, Thomassen P (eds) Proceedings of ECMOR IV, fourth European conference on the mathematics of oil recovery. European Association of Geoscientists & Engineers, Røros. 13 p

Levine RA, Casella G (2006) Optimizing random scan Gibbs samplers. J Multivar Anal 97:2071–2100

Matheron G, Beucher H, de Fouquet C, Galli A, Guérillot D, Ravenne C (1987) Conditional simulation of the geometry of fluvio-deltaic reservoirs. In: 62nd annual technical conference and exhibition of the society of petroleum engineers, Dallas, pp 591–599. SPE Paper 16753

Roberts GO (1996) Markov chain concepts related to sampling algorithms. In: Gilks WR, Richardson S, Spiegelhalter DJ (eds) Markov chain Monte Carlo in practice. Chapman & Hall/CRC Press, Boca Raton, pp 45–57

Roberts GO, Sahu SK (1997) Updating schemes, correlation structure, blocking and parameterization for the Gibbs sampler. J R Stat Soc B 59(2):291–317

Tierney L (1994) Markov chains for exploring posterior distributions. Ann Stat 22(4):1701–1762

Wilhelm S, Manjunath BG (2011) Tmvtnorm: truncated multivariate normal and Student t distribution. R package version 1.4-8. http://cran.r-project.org/package=tmvtnorm. Accessed 2 July 2013