ORIGINAL PAPER

Iterative algorithms for non-conditional and conditional simulation of Gaussian random vectors

Daisy Arroyo[1] · Xavier Emery[2,3]

Abstract

The conditional simulation of Gaussian random vectors is widely used in geostatistical applications to quantify uncertainty in regionalized phenomena that have been observed at finitely many sampling locations. Two iterative algorithms are presented to deal with such a simulation. The first one is a variation of the propagative version of the Gibbs sampler aimed at simulating the random vector without any conditioning data. The novelty of the presented algorithm stems from the introduction of a relaxation parameter that, if adequately chosen, allows quickening the rates of convergence and mixing of the sampler. The second algorithm is meant to convert the non-conditional simulation into a conditional one, based on the successive over-relaxation method. Again, a relaxation parameter allows quickening the convergence in distribution to the desired conditional random vector. Both algorithms are applicable in a very general setting and avoid the pivoting, inversion, square rooting or decomposition of the variance-covariance matrix of the vector to be simulated, thus reduce the computation costs and memory requirements with respect to other discrete geostatistical simulation approaches.

Keywords Gaussian random fields · Gibbs sampler · Mixing · Gauss-Seidel method · Successive over-relaxation method

1 Introduction

The simulation of random fields is widespread in geostatistics to quantify the uncertainty in regionalized phenomena that have been observed at a limited number of sampling locations. Applications in the earth sciences, among other disciplines, include the modeling of mineral deposits, hydrocarbon reservoirs, aquifers, forests, bedrocks, soils, lands and agricultural fields, see for instance Delfiner and Chilès (1977), Journel and Huijbregts (1978), Delhomme (1979), Matérn (1986), Shive et al. (1990), Chilès and Allard (2005) and Webster and Oliver (2007).

When restricting to Gaussian random fields (i.e., random fields whose finite-dimensional distributions are multivariate normal) and to finitely many locations in space, the problem boils down to simulating a n-dimensional Gaussian random vector $\mathbf{Y}$ with a pre-specified mean $\mathbf{m}$ and variance-covariance matrix $\mathbf{C}$. Numerous algorithms have been proposed in the past decades to perform such a simulation, see, for instance, Lantuéjoul (2002) and Chilès and Delfiner (2012) for a general overview. A few of them, such as the covariance matrix decomposition (Davis 1987; Alabert 1987; Rue 2001), circulant-embedding (Chellappa and Jain 1992; Dietrich and Newsam 1993; Pardo-Igúzquiza and Chica-Olmo 1993), perturbation-optimization (Orieux et al. 2012), autoregressive and moving average (Box and Jenkins 1976; Guyon 1995) algorithms, are exact in the sense that the simulated vector has a multivariate Gaussian distribution with the desired first- and second-order moments, but their implementation is straightforward only for particular structures of the covariance matrix $\mathbf{C}$ or its inverse, hence, unless the dimension n is small, they are applicable only for specific covariance models and/or for a regular (gridded) configuration of the locations targeted for simulation.

✉ Daisy Arroyo
darroyof@udec.cl

Xavier Emery
xemery@ing.uchile.cl

[1] Department of Statistics, University of Concepción, Concepción, Chile

[2] Department of Mining Engineering, University of Chile, Santiago, Chile

[3] Advanced Mining Technology Center, University of Chile, Santiago, Chile

Most of the other simulation algorithms available to date are approximate, either because the simulation is not exactly Gaussian or because it does not exactly have the desired first- and second-order moments. Non-Gaussianity often stems from a central limit approximation, e.g. when using the dilution (Alfaro 1980; Matérn 1986), continuous spectral (Shinozuka 1971; Emery et al. 2016) and turning bands (Matheron 1973; Emery and Lantuéjoul 2006) algorithms. The approximate moment reproduction arises due to a moving neighborhood implementation, e.g. when using the sequential Gaussian algorithm (Emery and Peláez 2011; Safikhani et al. 2017), or to the recourse to finitely many iterations when using Monte Carlo Markov Chain algorithms (Lantuéjoul 2002).

Forcing the simulation to reproduce sampling data is another challenge in many geostatistical applications. In the case of 'hard' data, i.e., when the regionalized phenomenon is observed without uncertainty at finitely many locations in space, the simulated random field can be conditioned to these data by kriging the residual between the data values and the simulated values at the data locations and by adding the kriged residual to the non-conditional simulation (Journel and Huijbregts 1978; Chilès and Delfiner 2012). However, in the presence of very large data sets, kriging is often set up in a moving neighborhood instead of a unique neighborhood, which makes the conditional simulation approximate (Chilès and Delfiner 2012; Marcotte and Allard 2018a).

This paper addresses the problem of the non-conditional and conditional simulation of a Gaussian random vector by means of iterative algorithms and presents some enhancements aimed at quickening the rate of convergence of these algorithms to the target (non-conditional or conditional) vector. The algorithm under consideration for the non-conditional simulation is the propagative version of the Gibbs sampler, initially proposed by Galli and Gao (2001) and later by Lantuéjoul and Desassis (2012) and Arroyo et al. (2012). In this case, the proposed enhancement consists of the use of a relaxation parameter, which will be studied in the next section. Concerning the conditioning process, the algorithm under consideration will rely on the method of successive over-relaxation, as will be detailed in Sect. 3. A general discussion and conclusions will follow in Sects. 4 and 5.

The motivation is to design efficient iterative algorithms that allow both the non-conditional and conditional simulation of Gaussian random vectors or random fields with any correlation structure, without the need for pivoting, inverting, square rooting or decomposing the target variance-covariance matrix. Therefore, these algorithms can be applied to the simulation of large random vectors and/or to the conditioning to a large set of hard data with affordable computational costs and memory requirements. The case of 'soft' data, such as interval data used in the simulation of

truncated Gaussian and plurigaussian random fields (Armstrong et al. 2011), is out of the scope of this paper and the reader is referred to Geweke (1991), Freulon and de Fouquet (1993), Freulon (1994), Wilhelm and Manjunath (2010), Emery et al. (2014), Pakman and Paninski (2014) and Marcotte and Allard (2018b) for iterative algorithms adapted to this case.

2 Non-conditional simulation

2.1 Problem setting

It is of interest to simulate a Gaussian random vector $\mathbf{Y}$ with n components, with zero mean and variance-covariance matrix $\mathbf{C}$ assumed to be (strictly) positive definite, i.e., without any redundancy between the vector components; note that it is possible to reduce to the case of a positive definite covariance matrix by removing redundant vector components. Under this assumption, the covariance matrix is invertible and we can introduce the 'dual' Gaussian random vector $\mathbf{X} = \mathbf{B}\,\mathbf{Y}$, with $\mathbf{B} = \mathbf{C}^{-1}$, which has zero mean and variance-covariance matrix $\mathbf{B}$ (Galli and Gao 2001; Arroyo et al. 2012).

2.2 Gibbs sampling on the dual vector X

Following Arroyo et al. (2012), the random vector $\mathbf{X}$ can be simulated iteratively by Gibbs sampling, by constructing a sequence of vectors $\{\mathbf{X}^{(k)}: k = 0, 1, 2, \ldots\}$ in the following way:

(1)   Initialization: set $\mathbf{X}^{(0)} = \mathbf{0}$ (a column vector of zeros).

(2)   Iteration: for $k = 1, 2, \ldots$

    a. Randomly split the vector $[1, \ldots, n]$ into two disjoint subsets I and J with $n - p$ and p components, respectively, in such a way that every integer in $[1, \ldots, n]$ has a non-zero probability to be included in subset J. Since the splitting varies with k, we should actually denote the subsets as I_k and J_k, but we omit the index k to keep a lighter notation. Hereunder, the subscripts I and J will denote the subvectors or submatrices whose components are indexed by I and J, respectively, and the bullet (•) will be used as a shortcut to $[1, \ldots, n]$.

    b. Define $\mathbf{X}^{(k)}$ as the random vector such that

$$\begin{cases} \mathbf{X}_I^{(k)} = \mathbf{X}_I^{(k-1)} \\ \mathbf{X}_J^{(k)} = \boldsymbol{\Omega}_{JI}\,\mathbf{X}_I^{(k-1)} + \mathbf{R}^{(k)}, \end{cases} \quad (1)$$

where $\boldsymbol{\Omega}_{JI}$ is the $p \times (n - p)$ matrix of simple kriging weights to predict $\mathbf{X}_J$ from $\mathbf{X}_I$, and $\mathbf{R}^{(k)}$ is

a Gaussian random vector, independent of $\mathbf{X}_I^{(k-1)}$, with zero mean and variance-covariance matrix equal to the $p \times p$ variance-covariance matrix $\boldsymbol{\Sigma}_J$ of the associated simple kriging errors. These matrices can be obtained from the variance-covariance matrix $\mathbf{C}$ as follows (Emery 2009):

$$\begin{cases} \boldsymbol{\Omega}_{JI} = -\mathbf{C}_{JJ}^{-1}\,\mathbf{C}_{JI} \\ \boldsymbol{\Sigma}_J = \mathbf{C}_{JJ}^{-1}. \end{cases} \tag{2}$$

In practice, the sequence is stopped after K iterations, where K is a large enough integer, and $\mathbf{X}^{(K)}$ is delivered as an approximate simulation of $\mathbf{X}$.

Arroyo et al. (2012) propose to choose $\mathbf{R}^{(k)} = \mathbf{S}_J\,\mathbf{U}^{(k)}$, where $\mathbf{S}_J\,\mathbf{S}_J^T = \boldsymbol{\Sigma}_J$ and $\mathbf{U}^{(k)}$ is a standard Gaussian random vector with p independent components, independent of $\mathbf{X}^{(k-1)}$. $\mathbf{S}_J$ could be, for instance, the Cholesky factor of $\boldsymbol{\Sigma}_J$ or any symmetric square root matrix of $\boldsymbol{\Sigma}_J$.

A more general formalism consists in choosing $\mathbf{R}^{(k)}$ as a linear combination of a Gaussian random vector independent of $\mathbf{X}^{(k-1)}$ with zero mean variance-covariance matrix $\boldsymbol{\Sigma}_J$, and the vector of simple kriging error $\boldsymbol{\Omega}_{JI}\mathbf{X}_I^{(k-1)} - \mathbf{X}_J^{(k-1)}$:

$$\mathbf{R}^{(k)} = \rho\left(-\mathbf{C}_{JJ}^{-1}\,\mathbf{C}_{JI}\,\mathbf{X}_I^{(k-1)} - \mathbf{X}_J^{(k-1)}\right) + \sqrt{1-\rho^2}\,\mathbf{S}_J\mathbf{U}^{(k)}, \tag{3}$$

with $\rho \in\,]{-}1,1[$ and $\mathbf{U}^{(k)}$ an independent standard Gaussian random vector defined as above. It can be shown that this choice still ensures that $\mathbf{X}^{(k)}$ converges in distribution to $\mathbf{X}$ (Appendix 1).

2.3 Gibbs sampling on the direct vector Y

By putting $\mathbf{Y}^{(k)} = \mathbf{C}\,\mathbf{X}^{(k)}$, one obtains a sequence of random vectors $\{\mathbf{Y}^{(k)}: k = 0, 1, 2, \dots\}$ that starts with $\mathbf{Y}^{(0)} = \mathbf{0}$ and converges in distribution to $\mathbf{Y} = \mathbf{C}\,\mathbf{X}$ as k tends to infinity. Based on Eqs. (1) and (2), the transition from $\mathbf{Y}^{(k-1)}$ to $\mathbf{Y}^{(k)}$ can be expressed as follows:

$$\begin{aligned} \mathbf{Y}_I^{(k)} &= \mathbf{C}_{II}\mathbf{X}_I^{(k)} + \mathbf{C}_{IJ}\mathbf{X}_J^{(k)} \\ &= \mathbf{C}_{II}\mathbf{X}_I^{(k-1)} + \mathbf{C}_{IJ}\mathbf{X}_J^{(k-1)} \\ &\quad + \mathbf{C}_{IJ}\left(\mathbf{R}^{(k)} - \mathbf{C}_{JJ}^{-1}\left(\mathbf{C}_{JJ}\mathbf{X}_J^{(k-1)} + \mathbf{C}_{JI}\,\mathbf{X}_I^{(k-1)}\right)\right) \\ &= \mathbf{Y}_I^{(k-1)} + \mathbf{C}_{IJ}\left(\mathbf{R}^{(k)} - \mathbf{C}_{JJ}^{-1}\mathbf{Y}_J^{(k-1)}\right), \end{aligned} \tag{4}$$

and

$$\begin{aligned} \mathbf{Y}_J^{(k)} &= \mathbf{C}_{JI}\mathbf{X}_I^{(k)} + \mathbf{C}_{JJ}\mathbf{X}_J^{(k)} \\ &= \mathbf{C}_{JI}\mathbf{X}_I^{(k-1)} + \mathbf{C}_{JJ}\mathbf{X}_J^{(k-1)} \\ &\quad + \mathbf{C}_{JJ}\left(\mathbf{R}^{(k)} - \mathbf{C}_{JJ}^{-1}\left(\mathbf{C}_{JJ}\mathbf{X}_J^{(k-1)} + \mathbf{C}_{JI}\,\mathbf{X}_I^{(k-1)}\right)\right) \\ &= \mathbf{Y}_J^{(k-1)} + \mathbf{C}_{JJ}\left(\mathbf{R}^{(k)} - \mathbf{C}_{JJ}^{-1}\mathbf{Y}_J^{(k-1)}\right), \end{aligned} \tag{5}$$

which reduces to

$$\begin{aligned} \mathbf{Y}^{(k)} &= \mathbf{Y}^{(k-1)} + \mathbf{C}_{\bullet J}\left(\mathbf{R}^{(k)} - \mathbf{C}_{JJ}^{-1}\mathbf{Y}_J^{(k-1)}\right) \\ &= \mathbf{Y}^{(k-1)} + \mathbf{C}_{\bullet J}\left(-\rho\,\mathbf{C}_{JJ}^{-1}\,\mathbf{Y}_J^{(k-1)} \right. \\ &\quad\left. + \sqrt{1-\rho^2}\,\mathbf{S}_J\mathbf{U}^{(k)} - \mathbf{C}_{JJ}^{-1}\mathbf{Y}_J^{(k-1)}\right) \\ &= \mathbf{Y}^{(k-1)} + \mathbf{C}_{\bullet J}\left(\sqrt{1-\rho^2}\,\mathbf{S}_J\mathbf{U}^{(k)} - (1+\rho)\,\mathbf{C}_{JJ}^{-1}\,\mathbf{Y}_J^{(k-1)}\right). \end{aligned} \tag{6}$$

ρ can be interpreted as a relaxation parameter, which can range from -1 to 1 (both bounds being excluded). When $\rho = 0$, one finds the Gibbs sampler proposed by Arroyo et al. (2012).

The idea of introducing a relaxation parameter in Gibbs sampling is not new and dates back from Adler (1981), Whitmer (1984), Barone and Frigessi (1990) and Galli and Gao (2001). Its combination with the propagative version of the Gibbs sampler, where the simulation of the target vector $\mathbf{Y}$ relies on the Gibbs sampling of the dual vector $\mathbf{X}$, has been proposed by Lantuéjoul and Desassis (2012), but their formulation differs from the one above presented, the transition from $\mathbf{Y}^{(k-1)}$ to $\mathbf{Y}^{(k)}$ proposed by these authors being

$$\begin{cases} \mathbf{Y}_J^{(k)} = \sqrt{1-\rho^2}\left(\mathbf{S}_J^{-1}\right)^T\mathbf{U}^{(k)} + \rho\,\mathbf{Y}_J^{(k-1)} \\ \mathbf{Y}_I^{(k)} = \mathbf{Y}_I^{(k-1)} + \mathbf{C}_{IJ}\left(\sqrt{1-\rho^2}\left(\mathbf{S}_J^{-1}\right)^T\mathbf{U}^{(k)} - (1-\rho)\,\mathbf{Y}_J^{(k-1)}\right) \\ \qquad = \mathbf{Y}_I^{(k-1)} + \mathbf{C}_{IJ}\,\mathbf{C}_{JJ}\left(\sqrt{1-\rho^2}\,\mathbf{S}_J\mathbf{U}^{(k)} - (1-\rho)\,\mathbf{C}_{JJ}^{-1}\,\mathbf{Y}_J^{(k-1)}\right). \end{cases} \tag{7}$$

2.4 Rate of convergence

Since $\mathbf{Y}^{(0)} = \mathbf{0}$ and $\{\mathbf{U}^{(k)}: k = 1, 2,\dots\}$ is a sequence of independent Gaussian random vectors with zero mean, the successive random vectors $\mathbf{Y}^{(k)}$ $(k = 0, 1, 2, \dots)$ are also Gaussian random vectors with zero mean. To assess the convergence rate of the Gibbs sampler, let us calculate the variance-covariance matrix $\mathbf{C}^{(k)}$ of $\mathbf{Y}^{(k)}$ $(k \geq 1)$ as a function of the simulation parameters ($\mathbf{C}$ and ρ) and the variance-covariance matrix $\mathbf{C}^{(k-1)}$ of $\mathbf{Y}^{(k-1)}$. Denoting by T in superscript the transposition and by $\mathbb{E}$ the expectation operator and using Eq. (6) and the fact that $\mathbf{U}^{(k)}$ is

independent of $\mathbf{X}^{(k-1)}$ (therefore, also independent of $\mathbf{Y}^{(k-1)}$), one has:

$$
\begin{aligned}
\mathbf{C}^{(k)} &= \mathrm{E}\left\{\mathbf{Y}^{(k)}\mathbf{Y}^{(k)T}\right\} \\
&= \mathrm{E}\left\{\mathbf{Y}^{(k-1)}\mathbf{Y}^{(k-1)T}\right\} - (1+\rho)\,\mathbf{C}_{\cdot J}\,\mathbf{C}_{JJ}^{-1}\,\mathrm{E}\left\{\mathbf{Y}_J^{(k-1)}\mathbf{Y}^{(k-1)T}\right\} \\
&\quad - (1+\rho)\,\mathrm{E}\left\{\mathbf{Y}^{(k-1)T}\mathbf{Y}_J^{(k-1)T}\right\}\mathbf{C}_{JJ}^{-1}\mathbf{C}_{J\cdot} \\
&\quad + \mathbf{C}_{\cdot J}\left((1-\rho^2)\,\mathbf{S}_J\mathrm{E}\left\{\mathbf{U}^{(k)}\mathbf{U}^{(k)T}\right\}\mathbf{S}_J^T\right. \\
&\quad \left. + (1+\rho)^2\mathbf{C}_{JJ}^{-1}\mathrm{E}\left\{\mathbf{Y}_J^{(k-1)}\mathbf{Y}_J^{(k-1)T}\right\}\mathbf{C}_{JJ}^{-1}\right)\mathbf{C}_{J\cdot},
\end{aligned}
\tag{8}
$$

that is:

$$
\begin{aligned}
\mathbf{C}^{(k)} &= \mathbf{C}^{(k-1)} - (1+\rho)\left(\mathbf{C}_{\bullet J}\mathbf{C}_{JJ}^{-1}\mathbf{C}_{J\bullet}^{(k-1)} + \mathbf{C}_{\bullet J}^{(k-1)}\mathbf{C}_{JJ}^{-1}\mathbf{C}_{J\bullet}\right) \\
&\quad + \mathbf{C}_{\bullet J}\left((1-\rho^2)\,\mathbf{C}_{JJ}^{-1} + (1+\rho)^2\mathbf{C}_{JJ}^{-1}\mathbf{C}_{JJ}^{(k)}\mathbf{C}_{JJ}^{-1}\right)\mathbf{C}_{J\bullet}.
\end{aligned}
\tag{9}
$$

Together with $\mathbf{C}^{(0)} = \mathbf{0}$, it is therefore possible to calculate $\mathbf{C}^{(k)}$ for any integer k. The fact that the sequence $\{\mathbf{Y}^{(k)}: k = 0, 1, 2, \ldots\}$ converges in distribution to $\mathbf{Y}$ implies that $\mathbf{C}^{(k)}$ converges to $\mathbf{C}$ as k tends to infinity. The question to elucidate is for which value of ρ the convergence is the fastest one. A partial answer to this question will be provided in the next subsection.

2.5 Experimental results

In this subsection, we investigate the convergence of $\mathbf{C}^{(k)}$ to $\mathbf{C}$ through synthetic case studies. Though the Gibbs sampler proposed in Eq. (6) is applicable to the simulation of large random vectors, the following experiments consider a vector of reduced size ($n = 2500$ components) in order to ease the calculation and storage of the successive covariance matrices $\mathbf{C}^{(0)},\ldots, \mathbf{C}^{(K)}$ and the sensitization to the relaxation parameter ρ and to the target covariance matrix $\mathbf{C}$. The goal is to provide a 'proof of concept' and practical guidelines on how to choose a relaxation parameter that quickens the convergence of the simulated vector to the target vector and on how many iterations (K) are needed before stopping the Gibbs sampler.

In detail, we consider a two-dimensional regular grid with 50×50 nodes and mesh 1×1. The sampler is initialized with $\mathbf{Y}^{(0)} = \mathbf{0}$ and is set in order to update one vector component at a time ($p = 1$) and to visit the grid nodes following random permutations, i.e., all the vector components are updated once after each 2500 iterations of the sampler. Six spatial correlation models (all of them with a unit variance) are put to the test:

- a stationary isotropic spherical covariance with range 10 units;

- a stationary isotropic spherical covariance with range 50 units;
- a stationary isotropic cubic covariance with range 10 units;
- a stationary isotropic cubic covariance with range 50 units;
- a stationary isotropic exponential covariance with scale parameter 10 units (practical range 30 units);
- a non-stationary Matérn covariance with scale parameter varying linearly from 1 to 20 from left to right and shape parameter varying linearly from 0.25 to 1.75 from top to bottom.

The equation of the spherical, cubic and exponential covariances can be found in the literature (Chilès and Delfiner 2012). The non-stationary Matérn covariance with location-dependent scale parameter a and shape parameter ν is given by (Emery and Arroyo 2018):

$$
\begin{aligned}
C(\mathbf{x}, \mathbf{x}') &= \frac{2a(\mathbf{x})\,a(\mathbf{x}')}{\bar{a}(\mathbf{x}, \mathbf{x}')^2\,\sqrt{\Gamma(\nu(\mathbf{x}))\,\Gamma(\nu(\mathbf{x}'))}} \\
&\quad \times \left(\frac{||\mathbf{x} - \mathbf{x}'||}{2\,\bar{a}(\mathbf{x}, \mathbf{x}')}\right)^{\bar{\nu}(\mathbf{x},\mathbf{x}')} K_{\bar{\nu}(\mathbf{x},\mathbf{x}')}\left(\frac{||\mathbf{x} - \mathbf{x}'||}{\bar{a}(\mathbf{x}, \mathbf{x}')}\right),
\end{aligned}
\tag{10}
$$

with $\bar{a}(\mathbf{x}, \mathbf{x}') = \sqrt{\frac{a^2(\mathbf{x})+a^2(\mathbf{x}')}{2}}$, $\bar{\nu}(\mathbf{x}, \mathbf{x}') = \frac{\nu(\mathbf{x})+\nu(\mathbf{x}')}{2}$, Γ the gamma function and K_ν the modified Bessel function of the second kind of order ν. The chosen models exhibit varied short-scale behavior (continuous but irregular for the spherical, exponential and Matérn with shape parameter lower than 0.5, smooth for the cubic and Matérn with shape parameter higher than 0.5) and large-scale behavior (correlation range or scale parameter) of the random field to be simulated. Also, the first four models (spherical and cubic) are compactly supported, with a finite correlation range, while the last two ones (exponential and Matérn) are not.

Following Arroyo et al. (2012), the deviation between the variance-covariance matrix of the simulated random vector $\mathbf{Y}^{(k)}$ and the target variance-covariance matrix $\mathbf{C}$ is measured by the Frobenius norm of the difference between both matrices, divided by the Frobenius norm of the target covariance matrix:

$$
\eta_k = \frac{||\mathbf{C}^{(k)} - \mathbf{C}||_F}{||\mathbf{C}||_F} = \frac{||\mathbf{C}^{(k)} - \mathbf{C}||_F}{||\mathbf{C}^{(0)} - \mathbf{C}||_F},
\tag{11}
$$

where $||.||_F$ indicates the Frobenius norm. This index is equal to 1 for $k = 0$ (initial state) and to 0 when convergence is perfectly reached, which eases its interpretation and the comparison of the different target covariance models. Any other matrix norm could be used instead of the Frobenius norm, such as the spectral norm (which, here, coincide with the spectral radius, as one deals with symmetric matrices); however, the calculation of the latter is much more time consuming than that of the Frobenius norm.

The sampler is tested with ρ between − 0.8 and 0.8 and stopped after $K = 37,500$ iterations, i.e., when each vector component has been updated 15 times. For each value of ρ and k ranging from 0 to 37,500, the covariance matrix $\mathbf{C}^{(k)}$ and the standardized Frobenius norm η_k are calculated as per Eqs. (9) and (11). The results are summarized in Tables 1, 2 and 3 and Fig. 1 and call for the following comments:

(1) In all the cases, the convergence of the non-relaxed Gibbs sampler (ρ = 0) is very fast. The standardized Frobenius norm of the difference between the covariance matrix of the simulated vector $\mathbf{C}^{(k)}$ and the target covariance matrix $\mathbf{C}$ is, by construction, equal to 1 in the initial state ($k = 0$), a figure that drops to less than 0.057 after 5 permutations (12,500 iterations) and to less than 0.01 after 15 permutations, for all the models under study. In other words, after 15 permutations, the covariance matrix of the simulated vector differs from the target covariance matrix less than 1% of what it differed in the initial state, suggesting that convergence is practically reached. Similar findings about convergence have already been pointed out by Arroyo et al. (2012).

(2) For almost all the covariance models put to the test, the convergence turns out to be the slowest when choosing a relaxation parameter close to 1 (ρ = 0.8) and the fastest when choosing a negative relaxation parameter comprised between − 0.6 and − 0.8, which can reduce the standardized Frobenius norm by a factor of up to 70% with respect to the non-relaxed Gibbs sampler. In particular, positive relaxation parameters lead to a significantly slower convergence than negative or zero relaxation parameters. The only exception is the short-range spherical

model, for which the convergence turns out to be faster with a relaxation parameter of 0.2 but, even in this case, the standardized Frobenius norm η_k is less than 0.01 (i.e., convergence is practically reached) after 15 permutations or 37,500 iterations for any ρ comprised between − 0.6 and 0.6. Analogous results showing a faster convergence with the use of a negative relaxation parameter have been reported for both the traditional (Adler 1981; Whitmer 1984; Green and Han 1992; Neal 1998) and the propagative (Lantuéjoul and Desassis 2012) versions of the Gibbs sampler, although the latter authors use a variant that differs from the one presented here (Eq. 7) and do not provide details on their numerical experiments. This faster convergence can be explained by the principle of antithetic sampling, which counteracts the autocorrelation between consecutive states of the sampler, allowing greater moves and, therefore, a more efficient exploration of the state space than the non-relaxed sampler does.

(3) From the previous statements, the fastest convergence of the Gibbs sampler is almost always reached with a negative relaxation parameter. As a rule of thumb, ρ = − 0.6 proves to be appropriate (if not optimal) in all the cases under consideration, which correspond to covariance models commonly used in geostatistics. After only 15 permutations over the target grid nodes, the deviation between the covariance matrix of the simulated vector and the target covariance matrix is less than 1% of the norm of the target covariance matrix, which gives an insight into how many iterations ($K = 15\,n$, with n the dimension of $\mathbf{Y}$) are required. Users can also make small-scale experiments like the one presented here to corroborate these 'default' choice of ρ and K.

Table 1 Convergence of the Gibbs sampler: standardized Frobenius norm η_k (Eq. 11) for the spherical covariance models and $k = 12,500$, 25,000 or 37,500 iterations

ρ	Spherical covariance with range 10			Spherical covariance with range 50		
	12,500 iterations	25,000 iterations	37,500 iterations	12,500 iterations	25,000 iterations	37,500 iterations
− 0.8	0.0339	0.0222	0.0107	0.0065	0.0049	0.0030
− 0.6	0.0311	0.0211	0.0076	0.0094	0.0074	0.0044
− 0.4	0.0364	0.0243	0.0062	0.0137	0.0106	0.0060
− 0.2	0.0449	0.0289	0.0052	0.0190	0.0146	0.0078
0	0.0566	0.0345	0.0045	0.0260	0.0199	0.0099
0.2	0.0737	0.0426	0.0041	0.0360	0.0274	0.0127
0.4	0.1022	0.0564	0.0043	0.0518	0.0395	0.0171
0.6	0.1591	0.0873	0.0061	0.0818	0.0624	0.0258
0.8	0.3115	0.1917	0.0208	0.1645	0.1269	0.0535

Table 2 Convergence of the Gibbs sampler: standardized Frobenius norm η_k (Eq. 11) for the cubic covariance models and k = 12,500, 25,000 or 37,500 iterations

ρ	Cubic covariance with range 10			Cubic covariance with range 50		
	12,500 iterations	25,000 iterations	37,500 iterations	12,500 iterations	25,000 iterations	37,500 iterations
− 0.8	0.0153	0.0094	0.0046	0.0018	0.0010	0.0005
− 0.6	0.0115	0.0077	0.0032	0.0014	0.0008	0.0004
− 0.4	0.0119	0.0078	0.0033	0.0015	0.0009	0.0005
− 0.2	0.0137	0.0088	0.0039	0.0018	0.0011	0.0006
0	0.0169	0.0107	0.0049	0.0023	0.0015	0.0008
0.2	0.0218	0.0137	0.0066	0.0030	0.0019	0.0010
0.4	0.0302	0.0189	0.0093	0.0042	0.0027	0.0014
0.6	0.0478	0.0299	0.0146	0.0066	0.0042	0.0023
0.8	0.1024	0.0642	0.0306	0.0135	0.0087	0.0049

Table 3 Convergence of the Gibbs sampler: standardized Frobenius norm η_k (Eq. 11) for the exponential and Matérn covariance models and k = 12,500, 25,000 or 37,500 iterations

ρ	Exponential covariance with scale parameter 10			Matérn covariance with varying scale and shape parameters		
	12,500 iterations	25,000 iterations	37,500 iterations	12,500 iterations	25,000 iterations	37,500 iterations
− 0.8	0.0163	0.0116	0.0062	0.0076	0.0048	0.0022
− 0.6	0.0207	0.0152	0.0067	0.0080	0.0052	0.0021
− 0.4	0.0280	0.0202	0.0071	0.0108	0.0067	0.0026
− 0.2	0.0369	0.0258	0.0072	0.0148	0.0083	0.0034
0	0.0483	0.0327	0.0072	0.0201	0.0102	0.0046
0.2	0.0646	0.0420	0.0075	0.0280	0.0129	0.0064
0.4	0.0909	0.0569	0.0086	0.0409	0.0175	0.0093
0.6	0.1419	0.0872	0.0121	0.0668	0.0276	0.0141
0.8	0.2780	0.1811	0.0306	0.1409	0.0634	0.0273

2.6 Mixing

To conclude the analysis of Gibbs sampling, it remains to study the mixing property of the sequence of random vectors $\{\mathbf{Y}^{(k)}: k = 0, 1, 2, \ldots\}$, which can be helpful to derive multiple non-conditional realizations of $\mathbf{Y}$ without having to restart the sequence from scratch. The rationale is the following: starting from a zero random vector $\mathbf{Y}^{(0)}$ (or any other initial state) and based on the previous guidelines, one obtains a random vector $\mathbf{Y}^{(K)}$ considered as an acceptable simulation of $\mathbf{Y}$ after K iterations (referred to the 'burn-in' period). Now, if more realizations of $\mathbf{Y}$ are required, instead of running the Gibbs sampler again and again, it may be preferable to follow with the same sequence and to retain the vectors obtained after every q iterations, i.e., consider $\{\mathbf{Y}^{(K)}, \mathbf{Y}^{(K+q)}, \mathbf{Y}^{(K+2q)}, \ldots\}$ as the successive realizations of $\mathbf{Y}$. This procedure is advantageous if one can obtain independent realizations (i.e., $\mathbf{Y}^{(K)}$ has no or very little correlation with $\mathbf{Y}^{(K+q)}$) for some q less than K.

Consider first the Gibbs sampling on the dual random vector $\mathbf{X}$ and assume that the sampler converges to $\mathbf{X}$ after K iterations, i.e., $\mathbf{X}^{(K)}$ has zero mean and variance-covariance matrix $\mathbf{B}$. Owing to the invariance of the target distribution under the transition kernel (Lantuéjoul 2002), every vector $\mathbf{X}^{(K+q)}$ with $q > 0$ also has zero mean and variance-covariance matrix $\mathbf{B}$. It is of interest to calculate the correlation between two consecutive such vectors. Based on Eq. (1), one has:
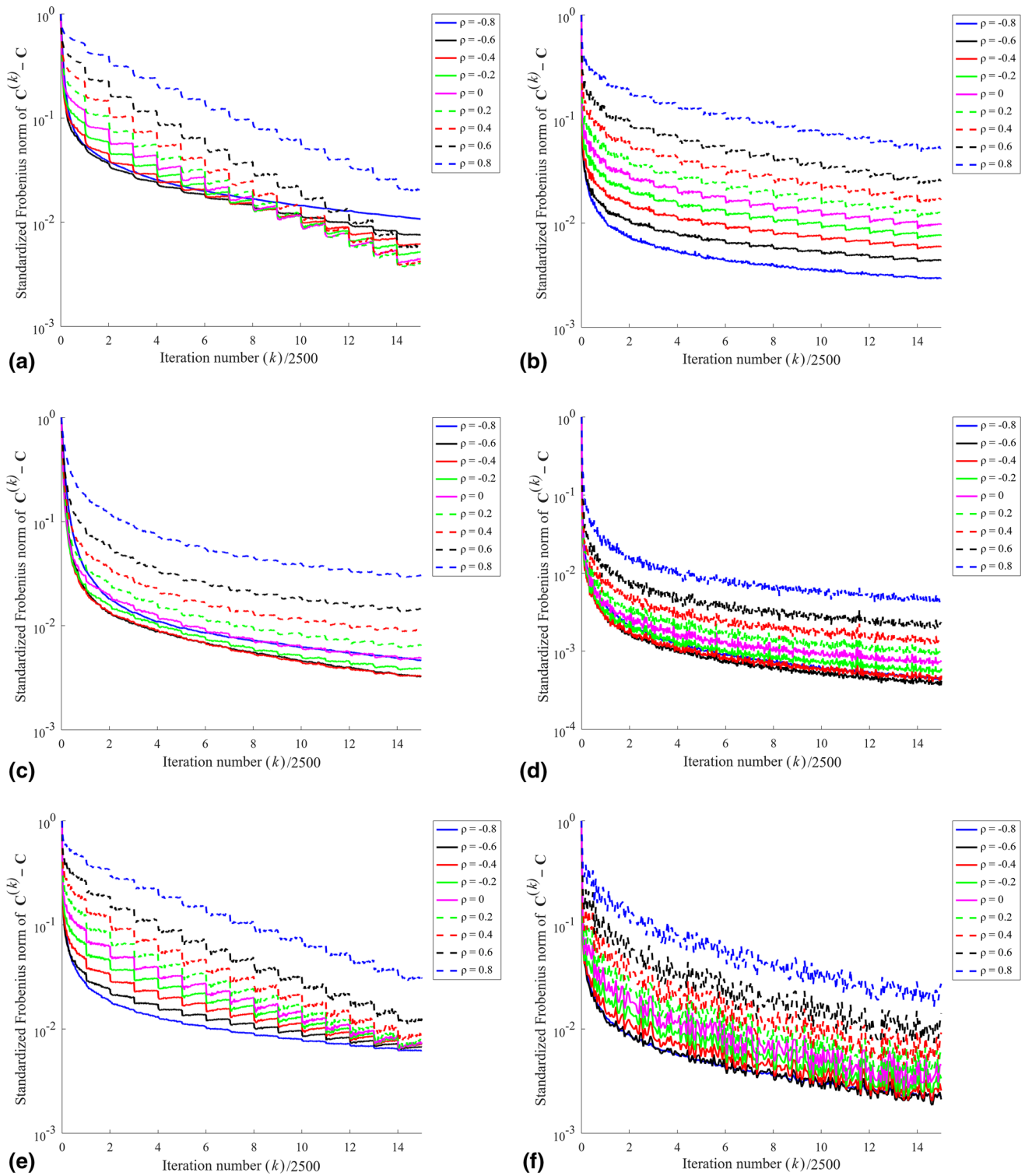
Fig. 1 Convergence of the Gibbs sampler: standardized Frobenius norm η_k for k varying between 0 and 37,500 and ρ varying between -0.8 and 0.8. **a** Spherical covariance with range 10, **b** spherical covariance with range 50, **c** cubic covariance with range 10, **d** cubic covariance with range 50, **e** exponential covariance with scale parameter 10, **f** Matérn covariance with spatially varying scale and shape parameters

$$\mathbb{E}\{\mathbf{X}^{(K+1)}\mathbf{X}^{(K)T}\} = \begin{bmatrix} \mathbb{E}\{\mathbf{X}_I^{(K)}\mathbf{X}^{(K)T}\} \\ \mathbb{E}\{(\boldsymbol{\Omega}_{JI}\mathbf{X}_I^{(K)} + \mathbf{R}^{(K+1)})\mathbf{X}^{(K)T}\} \end{bmatrix},$$

(12)

with (Eqs. 2–3)

$$\begin{cases} \boldsymbol{\Omega}_{JI} = -\mathbf{C}_{JJ}^{-1}\mathbf{C}_{JI} \\ \mathbf{R}^{(K+1)} = \rho\left(-\mathbf{C}_{JJ}^{-1}\mathbf{C}_{JI}\mathbf{X}_I^{(K)} - \mathbf{X}_J^{(K)}\right) + \sqrt{1-\rho^2}\,\mathbf{S}_J\mathbf{U}^{(K+1)}. \end{cases}$$

(13)

Accounting for the fact that $\mathbf{U}^{(K+1)}$ is not correlated with $\mathbf{X}^{(K)}$, it comes:

$$\mathrm{E}\{\mathbf{X}^{(K+1)}\mathbf{X}^{(K)T}\} =$$
$$\begin{bmatrix} \mathbf{B}_{II} & \mathbf{B}_{IJ} \\ \boldsymbol{\Omega}_{JI}\mathbf{B}_{II} - \rho\,\mathbf{C}_{JJ}^{-1}\mathbf{C}_{JI}\mathbf{B}_{II} - \rho\,\mathbf{B}_{JI} & \boldsymbol{\Omega}_{JI}\mathbf{B}_{IJ} - \rho\,\mathbf{C}_{JJ}^{-1}\mathbf{C}_{JI}\mathbf{B}_{IJ} - \rho\,\mathbf{B}_{JJ} \end{bmatrix}$$
$$= \begin{bmatrix} \mathbf{I}_{n-p} & \mathbf{0} \\ -(1+\rho)\mathbf{C}_{JJ}^{-1}\mathbf{C}_{JI} & -\rho\,\mathbf{I}_p \end{bmatrix}\mathbf{B},$$

(14)

with $\mathbf{I}_{n-p}$ and $\mathbf{I}_p$ the identity matrices of size $(n-p)\times(n-p)$ and $p\times p$, respectively, and $\mathbf{0}$ a zero matrix of size $(n-p)\times p$. In term of the direct vector $\mathbf{Y}$, this translates into:

$$\mathbb{E}\{\mathbf{Y}^{(K+1)}\mathbf{Y}^{(K)T}\} = \mathbf{C}\,\mathbb{E}\{\mathbf{X}^{(K+1)}\mathbf{X}^{(K)T}\}\,\mathbf{C}$$
$$= \mathbf{C}\begin{bmatrix} \mathbf{I}_{n-p} & \mathbf{0} \\ -(1+\rho)\,\mathbf{C}_{JJ}^{-1}\mathbf{C}_{JI} & -\rho\,\mathbf{I}_p \end{bmatrix}.$$

(15)

A recursive application of Eq. (15) allows calculating the covariance matrix $\mathbf{C}^{(K,K+q)}$ between $\mathbf{Y}^{(K)}$ and $\mathbf{Y}^{(K+q)}$ for any positive integer q.

As an illustration, let us revisit the numerical experiments shown in Sect. 2.5, consisting of simulating a Gaussian random vector of size $n = 2500$. Following Arroyo et al. (2012), we define the following standardized index aimed at measuring the mixing property:

$$\theta_q = \frac{||\mathbf{C}^{(K,K+q)}||_F}{||\mathbf{C}||_F} = \frac{||\mathbf{C}^{(K,K+q)}||_F}{||\mathbf{C}^{(K,K)}||_F},$$

(16)

with K a large integer such that the convergence of the Gibbs sampler can be considered as reached. As for η_k (Eq. 11), this new index θ_q is equal to 1 for $q = 0$ and to 0 in case of perfect mixing (independence between $\mathbf{Y}^{(K)}$ and $\mathbf{Y}^{(K+q)}$), which eases the comparison of the different target covariance models and the comparison with the convergence results presented in Sect. 2.5. The evolution of θ_q as a function of q is shown in Fig. 2, for the six covariance models under consideration, ρ varying between -0.8 and 0.8 and q varying between 0 and 37,500. Table 4 only presents the values of $\theta_{37,500}$ in each case. The following comments can be made:

(1) Mixing occurs for all the covariance models under consideration at a rate comparable to the rate of convergence (the standardized index $\theta_{37,500}$ is of the same order of magnitude as $\eta_{37,500}$ and below 1% in all the cases with a suitable choice of the relaxation parameter ρ).

(2) Unlike the convergence results, here there seems to be no or little advantage of using a relaxation parameter: taking $\rho = 0$ practically leads to the highest mixing rate for all the covariance models, irrespective of the number of iterations q (the purple curve in Fig. 2 is consistently below the other curves). This makes sense if one looks at Eq. (15), which suggests that $\rho = 0$ minimizes the correlation between consecutive states of the sampler.

(3) In practice, to generate several realizations of the target Gaussian random vectors, rather than running the sampler once and retaining every q states after the burn-in period, it may be faster to run the sampler with different initial states (for instance, the initial state can be a vector of ones multiplied by a standard normal random variable), which allows drawing several independent random vectors with the same choice of the subsets (I, J) at each iteration and saving time in the calculation of the covariance matrices and vectors (Eq. 6) required at such an iteration.

## 3 Simulation conditioned to hard data

3.1 Problem setting

It is now of interest to simulate a n-dimensional Gaussian random vector $\mathbf{Y}$ with zero mean and positive definite variance-covariance matrix $\mathbf{C}$, conditionally to the knowledge of some of the vector components. For the sake of simplicity, let us reorder the vector to be simulated as follows:

$$\mathbf{Y} = \begin{pmatrix} \mathbf{Y}_U \\ \mathbf{Y}_O \end{pmatrix},$$

(17)

where $\mathbf{Y}_U$ and $\mathbf{Y}_O$ are subvectors with $n - o$ and o components, corresponding to the unknown (U) and observed (O) values of $\mathbf{Y}$, respectively. The problem therefore amounts to simulating $\mathbf{Y}_U$ conditionally to $\mathbf{Y}_O = \mathbf{y}$.

Notation: hereunder, the subscripts NCS and CS will be used to refer to non-conditional and conditional simulation, respectively.
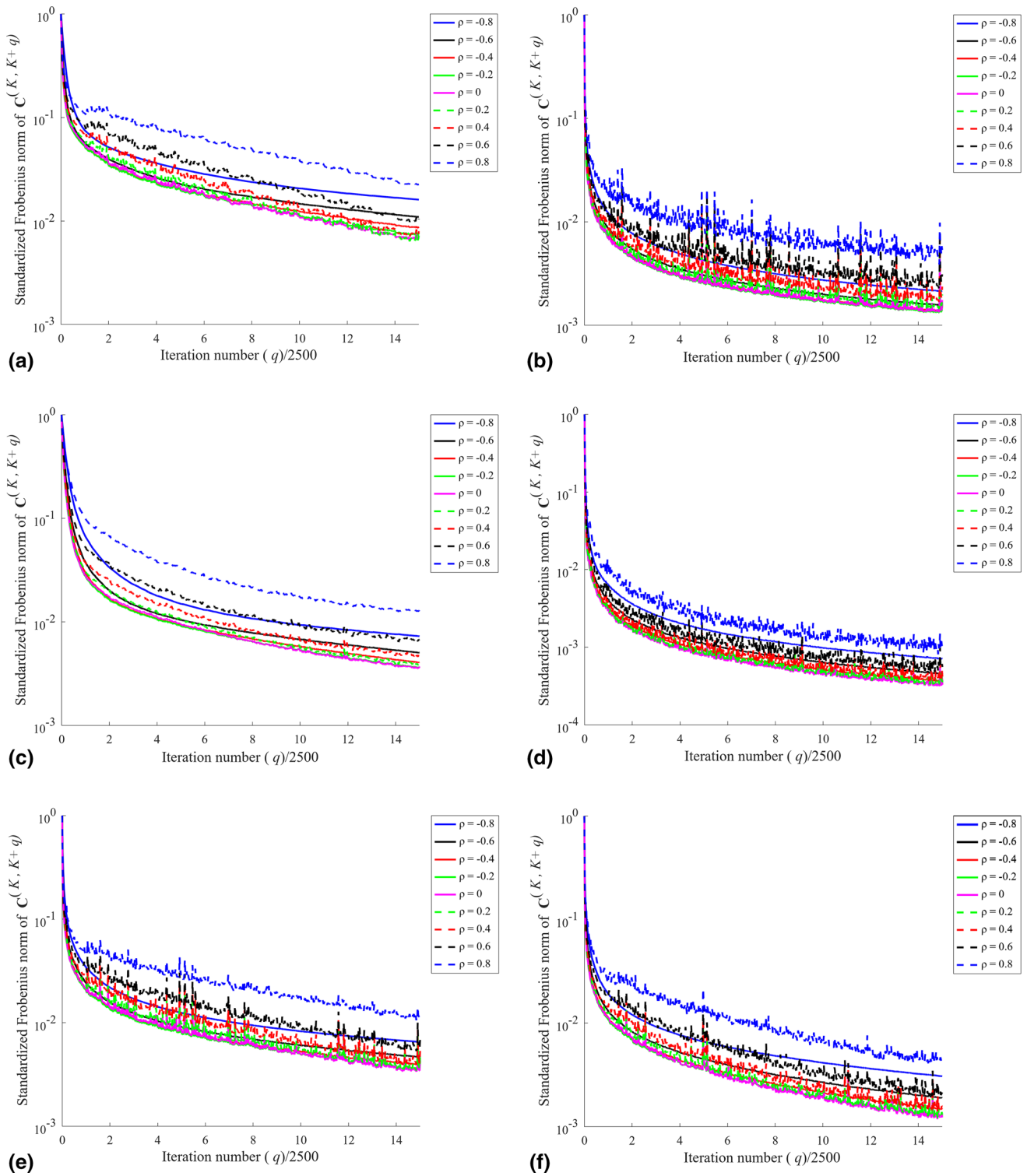
Fig. 2 Mixing of the Gibbs sampler: standardized Frobenius norm θ_q for q varying between 0 and 37,500 and ρ varying between -0.8 and 0.8. **a** Spherical covariance with range 10, **b** spherical covariance with range 50, **c** cubic covariance with range 10, **d** cubic covariance with range 50, **e** exponential covariance with scale parameter 10, **f** Matérn covariance with spatially varying scale and shape parameters

Table 4 Mixing of the Gibbs sampler: standardized Frobenius norm θ_q (Eq. 16) for the spherical, cubic, exponential and Matérn covariance models and $q = 37{,}500$ iterations

ρ	Spherical (range 10)	Spherical (range 50)	Cubic (range 10)	Cubic (range 50)	Exponential	Matérn
− 0.8	0.0160	0.0021	0.0073	0.0007	0.0065	0.0031
− 0.6	0.0110	0.0016	0.0050	0.0005	0.0048	0.0019
− 0.4	0.0088	0.0016	0.0041	0.0004	0.0043	0.0015
− 0.2	0.0077	0.0016	0.0037	0.0003	0.0042	0.0013
0	0.0072	0.0018	0.0037	0.0003	0.0044	0.0012
0.2	0.0072	0.0020	0.0041	0.0004	0.0047	0.0013
0.4	0.0080	0.0023	0.0049	0.0004	0-0053	0.0016
0.6	0.0105	0.0021	0.0068	0.0006	0.0068	0.0021
0.8	0.0226	0.0053	0.0127	0.0009	0.0124	0.0041

3.2 Conditioning the direct and dual random vectors by residual kriging

As in the previous section, let us introduce the dual vector $\mathbf{X} = \mathbf{B}\,\mathbf{Y}$ with $\mathbf{B} = \mathbf{C}^{-1}$. This vector can also be split into two subvectors with $n - o$ and o components:

$$\mathbf{X} = \begin{pmatrix} \mathbf{X}_U \\ \mathbf{X}_O \end{pmatrix}, \tag{18}$$

such that

$$\begin{pmatrix} \mathbf{Y}_U \\ \mathbf{Y}_O \end{pmatrix} = \begin{pmatrix} \mathbf{C}_{OO} & \mathbf{C}_{OU} \\ \mathbf{C}_{UO} & \mathbf{C}_{UU} \end{pmatrix} \begin{pmatrix} \mathbf{X}_U \\ \mathbf{X}_O \end{pmatrix}, \tag{19}$$

with $\mathbf{C}_{OO}$, $\mathbf{C}_{OU} = \mathbf{C}_{UO}^T$ and $\mathbf{C}_{UU}$ the suitable block-matrices extracted from $\mathbf{C}$.

Let $\mathbf{Y}_{NCS}$ be a non-conditional simulation of $\mathbf{Y}$, which can be constructed by Gibbs sampling (Sect. 2.3) or any other Gaussian simulation algorithm. A simulation $\mathbf{Y}_{CS}$ conditioned to $\mathbf{Y}_O = \mathbf{y}$ is obtained by residual kriging (Journel and Huijbregts 1978; Chilès and Delfiner 2012):

$$\begin{cases} \mathbf{Y}_{CS,U} = \mathbf{Y}_{NCS,U} + \mathbf{C}_{UO}\,\mathbf{C}_{OO}^{-1}\,(\mathbf{y} - \mathbf{Y}_{NCS,O}) \\ \mathbf{Y}_{CS,O} = \mathbf{Y}_{NCS,O} + \mathbf{C}_{OO}\,\mathbf{C}_{OO}^{-1}\,(\mathbf{y} - \mathbf{Y}_{NCS,O}) = \mathbf{y}, \end{cases} \tag{20}$$

where $\mathbf{C}_{UO}\mathbf{C}_{OO}^{-1}$ and $\mathbf{C}_{OO}\mathbf{C}_{OO}^{-1}$ are matrices of simple kriging weights. Now, let us examine the conditioning process on the dual vector $\mathbf{X}$. We start by writing the conditional vector as $\mathbf{X}_{CS} = \mathbf{B}\,\mathbf{Y}_{CS}$, which can be split into two subvectors $\mathbf{X}_{CS,U}$ and $\mathbf{X}_{CS,O}$ such that:

$$\begin{aligned} \mathbf{X}_{CS,U} &= \mathbf{B}_{UU}\,\mathbf{Y}_{CS,U} + \mathbf{B}_{UO}\,\mathbf{Y}_{CS,O} \\ &= \mathbf{B}_{UU}\,\mathbf{Y}_{NCS,U} + \mathbf{B}_{UU}\mathbf{C}_{UO}\,\mathbf{C}_{OO}^{-1}\,(\mathbf{y} - \mathbf{Y}_{NCS,O}) + \mathbf{B}_{UO}\,\mathbf{y} \\ &= \mathbf{B}_{UU}\,\mathbf{Y}_{NCS,U} - \mathbf{B}_{UO}\mathbf{C}_{OO}\,\mathbf{C}_{OO}^{-1}\,(\mathbf{y} - \mathbf{Y}_{NCS,O}) + \mathbf{B}_{UO}\,\mathbf{y} \\ &= \mathbf{B}_{UU}\,\mathbf{Y}_{NCS,U} + \mathbf{B}_{UO}\,\mathbf{Y}_{NCS,O} \\ &= \mathbf{X}_{NCS,U}, \end{aligned} \tag{21}$$

and

$$\begin{aligned} \mathbf{X}_{CS,O} &= \mathbf{B}_{OU}\,\mathbf{Y}_{CS,U} + \mathbf{B}_{OO}\,\mathbf{Y}_{CS,O} \\ &= \mathbf{B}_{OU}\,\mathbf{Y}_{NCS,U} + \mathbf{B}_{OU}\mathbf{C}_{UO}\,\mathbf{C}_{OO}^{-1}\,(\mathbf{y} - \mathbf{Y}_{NCS,O}) + \mathbf{B}_{OO}\,\mathbf{y} \\ &= \mathbf{B}_{OU}\,\mathbf{Y}_{NCS,U} + (\mathbf{I}_o - \mathbf{B}_{OO}\mathbf{C}_{OO})\,\mathbf{C}_{OO}^{-1}\,(\mathbf{y} - \mathbf{Y}_{NCS,O}) + \mathbf{B}_{OO}\,\mathbf{y} \\ &= \mathbf{B}_{OU}\,\mathbf{Y}_{NCS,U} + \mathbf{B}_{OO}\,\mathbf{Y}_{NCS,O} + \mathbf{C}_{OO}^{-1}\,(\mathbf{y} - \mathbf{Y}_{NCS,O}) \\ &= \mathbf{X}_{NCS,O} + \mathbf{C}_{OO}^{-1}\,(\mathbf{y} - \mathbf{Y}_{NCS,O}). \end{aligned} \tag{22}$$

Equations (21) and (22) have been established by using the identities $\mathbf{B}_{UU}\,\mathbf{C}_{UO} = -\mathbf{B}_{UO}\,\mathbf{C}_{OO}$ and $\mathbf{B}_{UO}\,\mathbf{C}_{UO} = (\mathbf{I}_o - \mathbf{B}_{OO}\,\mathbf{C}_{OO})$ (with $\mathbf{I}_o$ the identity matrix of size $o \times o$), which stem from the fact that $\mathbf{B} = \mathbf{C}^{-1}$. They prove that the conditioning process has no effect on subvector $\mathbf{X}_U$, the conditional simulation of which is exactly the same as its non-conditional simulation, and only affects subvector $\mathbf{X}_O$, the conditional simulation of which is the sum of the non-conditional simulation $\mathbf{X}_{NCS,O}$ and the kriging of $\mathbf{X}_{NCS,O}$ from the residual $\mathbf{y} - \mathbf{Y}_{NCS,O}$.

Accordingly, a conditional simulation of the dual vector $\mathbf{X}$, and therefore of the direct vector $\mathbf{Y}$, can be obtained by simulating $\mathbf{X}$ without any conditioning data, and then only conditioning the subvector $\mathbf{X}_O$ corresponding to the indices of the conditioning data.

3.3 Solving the conditioning equations by the Gauss-Seidel method

Because $\mathbf{Y} = \mathbf{C}\,\mathbf{X}$, Eq. (22) can be rewritten as follows:

$$\begin{aligned}
\mathbf{X}_{CS,O} &= \mathbf{X}_{NCS,O} + \mathbf{C}_{OO}^{-1}\left(\mathbf{y} - \mathbf{C}_{OU}\mathbf{X}_{NCS,U} - \mathbf{C}_{OO}\mathbf{X}_{NCS,O}\right) \\
&= \mathbf{C}_{OO}^{-1}\left(\mathbf{y} - \mathbf{C}_{OU}\mathbf{X}_{NCS,U}\right).
\end{aligned}$$

(23)

Equivalently,

$$\mathbf{C}_{OO}\,\mathbf{X}_{CS,O} = \mathbf{y} - \mathbf{C}_{OU}\mathbf{X}_{NCS,U}.$$

(24)

Having calculated a non-conditional simulation of $\mathbf{X}$, the conditional subvector $\mathbf{X}_{CS,O}$ can be obtained by solving this linear system of equations. To avoid inverting or pivoting matrix $\mathbf{C}_{OO}$, which can be large in the presence of many conditioning data ($o \gg 1$), one can solve the system iteratively with the Gauss-Seidel method (Young 2003). The convergence of this method is ensured because $\mathbf{C}_{OO}$ is symmetric positive definite.

The algorithm is the following:

(1) Initialization: set $\mathbf{X}_{CS}^{(0)} = \mathbf{X}_{NCS}$ and $\mathbf{Y}_{CS}^{(0)} = \mathbf{C}\,\mathbf{X}_{NCS}$ (non-conditional simulations of $\mathbf{X}$ and $\mathbf{Y}$).

(2) Iteration: for $m = 1, 2, \ldots$,

    (a) Set $j - (n - o + 1) = (m - 1)\,[\text{mod } o]$ (where 'mod' stands for modulo), so that j repeatedly loops over the indices of subset $O = [n - o + 1, \ldots, n]$ as m increases.

    (b) Update the j-th component of $\mathbf{X}_{CS}$ as follows:

$$\begin{aligned}
\mathbf{X}_{CS,j}^{(m)} &= \mathbf{C}_{jj}^{-1}\Big(\mathbf{y}_j - \mathbf{C}_{jU}\mathbf{X}_{NCS,U} \\
&\quad - \Big(\mathbf{C}_{jO}\mathbf{X}_{CS,O}^{(m-1)} - \mathbf{C}_{jj}\mathbf{X}_{CS,j}^{(m-1)}\Big)\Big) \\
&= \mathbf{X}_{CS,j}^{(m-1)} + \mathbf{C}_{jj}^{-1}\Big(\mathbf{y}_j - \mathbf{Y}_{CS,j}^{(m-1)}\Big).
\end{aligned}$$

(25)

    (c) Update vector $\mathbf{Y}_{CS}$ as follows:

$$\mathbf{Y}_{CS}^{(m)} = \mathbf{Y}_{CS}^{(m-1)} + \mathbf{C}_{\bullet j}\,\mathbf{C}_{jj}^{-1}\Big(\mathbf{y}_j - \mathbf{Y}_{CS,j}^{(m-1)}\Big).$$

(26)

In practice, the sequence is stopped after M iterations, where M is a large enough multiple of o, and $\mathbf{Y}_{CS}^{(M)}$ is delivered as an approximate simulation of $\mathbf{Y}_{CS}$.

Interestingly, Eq. (26) shows that the j-th component of $\mathbf{Y}_{CS}^{(m)}$ perfectly matches the j-th conditioning data: $\mathbf{Y}_{CS,j}^{(m)} = \mathbf{y}_j$. However, the remaining components of $\mathbf{Y}_{CS,O}^{(m)}$ have no reason to match the remaining data, except asymptotically when m tends to infinity and convergence of the Gauss-Seidel method is reached.

Also note the similarity between Eqs. (6) and (26) when taking $J = j$ and $\rho = 0$ (non-relaxed Gibbs sampler): both

equations formally look the same, except that the random vector $\mathbf{S}_J\,\mathbf{U}^{(k)}$ of Eq. (6) is substituted in Eq. (26) with the deterministic value $\mathbf{C}_{jj}^{-1}\,\mathbf{y}_j$. The parallel between Gibbs sampling and the Gauss-Seidel method has already been pointed out by Galli and Gao (2001).

3.4 Solving the conditioning equations by the method of successive over-relaxation

An improvement to the Gauss-Seidel method is that of successive over-relaxation, which depends on a relaxation parameter $\omega \in\,]0,2[$ (Young 2003). The update associated with the j-th component at iteration m is expressed as follows:

$$\begin{aligned}
\mathbf{X}_{CS,j}^{(m)} &= \omega\left(\mathbf{X}_{CS,j}^{(m-1)} + \mathbf{C}_{jj}^{-1}\Big(\mathbf{y}_j - \mathbf{Y}_{CS,j}^{(m-1)}\Big)\right) + (1 - \omega)\,\mathbf{X}_{CS,j}^{(m-1)} \\
&= \mathbf{X}_{CS,j}^{(m-1)} + \omega\,\mathbf{C}_{jj}^{-1}\Big(\mathbf{y}_j - \mathbf{Y}_{CS,j}^{(m-1)}\Big),
\end{aligned}$$

(27)

and

$$\mathbf{Y}_{CS}^{(m)} = \mathbf{Y}_{CS}^{(m-1)} + \omega\,\mathbf{C}_{\bullet j}\,\mathbf{C}_{jj}^{-1}\Big(\mathbf{y}_j - \mathbf{Y}_{CS,j}^{(m-1)}\Big).$$

(28)

Equations (27) and (28) generalize Eqs. (25) and (26) that correspond to the particular case $\omega = 1$.

3.5 Rate of convergence

To assess the rate of convergence of the sequence $\{\mathbf{Y}_{CS}^{(m)}: m = 0, 1, 2, \ldots\}$ when starting from a non-conditional simulation $\mathbf{Y}_{CS}^{(0)} = \mathbf{C}\,\mathbf{X}_{NCS}$ (assumed to be perfect, as if it were obtained after infinitely many Gibbs sampling iterations), let us calculate the expectation vector $\tilde{\mathbf{E}}^{(m)}$ and the variance-covariance matrix $\tilde{\mathbf{C}}^{(m)}$ of $\mathbf{Y}_{CS}^{(m)}$ (with $m \geq 1$), as a function of the expectation vector $\tilde{\mathbf{E}}^{(m-1)}$ and variance-covariance matrix $\tilde{\mathbf{C}}^{(m-1)}$ of $\mathbf{Y}_{CS}^{(m-1)}$.

3.5.1 Expectation vector

The expectation of $\mathbf{Y}_{CS}^{(0)}$ is zero: $\tilde{\mathbf{E}}^{(0)} = \mathbf{0}$. For $m > 0$, the expectation of $\mathbf{Y}_{CS}^{(m)}$ is (Eq. 28):

$$\tilde{\mathbf{E}}^{(m)} = \tilde{\mathbf{E}}^{(m-1)} + \omega\,\mathbf{C}_{\bullet j}\,\mathbf{C}_{jj}^{-1}\Big(\mathbf{y}_j - \tilde{\mathbf{E}}_j^{(m-1)}\Big).$$

(29)

3.5.2 Non-centered covariance matrix

Before dealing with the variance-covariance matrix, we start by calculating the non-centered covariance matrix of $\mathbf{Y}_{CS}^{(m)}$ (hereafter denoted with a hat):

$$\hat{\mathbf{C}}^{(m)} = \mathbb{E}\left\{ \mathbf{Y}_{CS}^{(m)} \, \mathbf{Y}_{CS}^{(m)T} \right\}. \tag{30}$$

Based on Eq. (28), it comes, for $m \geq 1$:

$$\hat{\mathbf{C}}^{(m)} = \hat{\mathbf{C}}^{(m-1)} + \omega^2 \, \mathbf{C}_{\cdot j} \, \mathbf{C}_{j \cdot} \, \mathbf{C}_{jj}^{-2} \left(\mathbf{y}_j^2 - 2\mathbf{y}_j \tilde{\mathbf{E}}_j^{(m-1)} + \hat{\mathbf{C}}_{jj}^{(m-1)} \right)$$
$$+ \omega \, \mathbf{C}_{jj}^{-1} \left(\mathbf{y}_j \tilde{\mathbf{E}}^{(m-1)} - \hat{\mathbf{C}}_{\cdot j}^{(m-1)} \right) \mathbf{C}_{j \cdot}$$
$$+ \omega \, \mathbf{C}_{jj}^{-1} \mathbf{C}_{\cdot j} \left(\mathbf{y}_j \tilde{\mathbf{E}}^{(m-1)} - \hat{\mathbf{C}}_{\cdot j}^{(m-1)} \right)^T. \tag{31}$$

3.5.3 Variance-covariance matrix

This matrix is calculated as: $\tilde{\mathbf{C}}^{(m)} = \hat{\mathbf{C}}^{(m)} - \tilde{\mathbf{E}}^{(m)} \tilde{\mathbf{E}}^{(m)T}$. Based on Eqs. (29) and (31), one finds, after simplification:

$$\tilde{\mathbf{C}}^{(m)} = \tilde{\mathbf{C}}^{(m-1)} - \omega \, \mathbf{C}_{jj}^{-1} \left(\tilde{\mathbf{C}}_{\bullet j}^{(m-1)} \, \mathbf{C}_{j \bullet} + \mathbf{C}_{\bullet j} \tilde{\mathbf{C}}_{j \bullet}^{(m-1)} \right)$$
$$+ \omega^2 \, \mathbf{C}_{jj}^{-2} \, \tilde{\mathbf{C}}_{jj}^{(m-1)} \, \mathbf{C}_{\bullet j} \, \mathbf{C}_{j \bullet}. \tag{32}$$

Together with $\tilde{\mathbf{C}}^{(0)} = \mathbf{C}$, one can calculate $\tilde{\mathbf{C}}^{(m)}$ for any integer m. The fact that the sequence $\{\mathbf{Y}_{CS}^{(m)}: m = 0, 1, 2, \ldots\}$ converges in distribution to the desired conditional random vector $\mathbf{Y}_{CS}$ implies that the expectation vector converges elementwise to the conditional expectation (simple kriging predictor) and the variance-covariance matrix converges elementwise to the variance-covariance matrix of simple kriging errors, that is:

$$\tilde{\mathbf{E}}^{(m)} \underset{m \to +\infty}{\longrightarrow} \tilde{\mathbf{E}}^{(\infty)} = \begin{pmatrix} \mathbf{C}_{UO} \, \mathbf{C}_{OO}^{-1} \, \mathbf{y} \\ \mathbf{y} \end{pmatrix}, \tag{33}$$

and (Alabert 1987)

$$\tilde{\mathbf{C}}^{(m)} \underset{m \to +\infty}{\longrightarrow} \tilde{\mathbf{C}}^{(\infty)} = \begin{pmatrix} \mathbf{C}_{UU} - \mathbf{C}_{UO} \, \mathbf{C}_{OO}^{-1} \mathbf{C}_{OU} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{pmatrix}. \tag{34}$$

Numerical experiments are presented in the next subsection, in which the rate of convergence is examined as a function of the relaxation parameter ω.

3.6 Experimental results

We go back to the numerical experiments presented in Sect. 2.5 and randomly select 200 conditioning data points among the 2500 grid nodes. For each covariance model under study, a reference non-conditional simulation is constructed with the Cholesky decomposition algorithm (Davis 1987) and used to fix the values at the conditioning data points.

The method of successive over-relaxation (Eq. 28) is then used to progressively convert another non-conditional simulation into a conditional one, for different values of ω between 0 and 2. The iterations are stopped at $M = 5000$,

i.e., after 25 loops over the 200 components of the observation subset O. For each value of ω and m, the expectation vector $\tilde{\mathbf{E}}^{(m)}$ (Eq. 29) and the variance-covariance matrix $\tilde{\mathbf{C}}^{(m)}$ (Eq. 32) are calculated. The convergence to the target expectation vector and variance-covariance matrix is assessed through the standardized Frobenius norms:

$$\mu_m = \frac{||\tilde{\mathbf{E}}^{(m)} - \tilde{\mathbf{E}}^{(\infty)}||_F}{||\tilde{\mathbf{E}}^{(0)} - \tilde{\mathbf{E}}^{(\infty)}||_F} = \frac{||\tilde{\mathbf{E}}^{(m)} - \tilde{\mathbf{E}}^{(\infty)}||_F}{||\tilde{\mathbf{E}}^{(\infty)}||_F}, \tag{35}$$

$$v_m = \frac{||\tilde{\mathbf{C}}^{(m)} - \tilde{\mathbf{C}}^{(\infty)}||_F}{||\tilde{\mathbf{C}}^{(0)} - \tilde{\mathbf{C}}^{(\infty)}||_F} = \frac{||\tilde{\mathbf{C}}^{(m)} - \tilde{\mathbf{C}}^{(\infty)}||_F}{||\mathbf{C} - \tilde{\mathbf{C}}^{(\infty)}||_F}. \tag{36}$$

The results, summarized in Tables 5 and 6 and Figs. 3 and 4, indicate that, with a suitable choice of the relaxation parameter ω, both $\tilde{\mathbf{E}}^{(m)}$ and $\tilde{\mathbf{C}}^{(m)}$ quickly converge to the target conditional expectation vector and conditional variance-covariance matrix: after only 25 permutations over the data points ($M = 5000$ iterations), μ_M and v_M decrease by more than 85% and 99%, respectively, from their initial value $\mu_0 = v_0 = 1$ when ω is comprised between 0.6 and 1.4. In particular, the convergence turns out to be the fastest with ω close to 1.4 for the compactly-supported covariance models with a short-range (10 units), and with ω close to 0.6 for the compactly-supported covariance models with a long range (50 units). As a rule of thumb, a relaxation parameter around 1.2 yields good convergence results for all the tested covariance models, irrespective of whether or not they are compactly-supported or stationary. In contrast, relaxation parameters close to 0 or 2 consistently yield the slowest convergence.

4 Discussion and synthesis

The previous theoretical and experimental results prove that iterative algorithms (propagative version of the Gibbs sampler and method of successive over-relaxation) can be used for both the non-conditional and conditional simulation of Gaussian random vectors. The simulated vector converges quickly in distribution to the desired vector, and the convergence can even be made faster with a suitable choice of the relaxation parameter. As a rule of thumb derived from the experiments shown in Sects. 2.5 and 3.6, $\rho = -0.6$ and $\omega = 1.2$ lead to nearly optimal convergence results for all the tested covariance models; similar conclusions have been obtained with experiments involving other grid sizes and covariance models (not shown in this paper).

Although the size of the chosen simulation grid was purposely small (2500 nodes) in order to evaluate and store the successive variance-covariance matrices, the proposed

Table 5 Convergence of the successive over-relaxation method: standardized Frobenius norm μ_{5000} (Eq. 35) for the expectation vector

ω	Spherical (range 10)	Spherical (range 50)	Cubic (range 10)	Cubic (range 50)	Exponential	Matérn
0.2	0.0864	0.1436	0.2294	0.0950	0.1084	0.1965
0.4	0.0317	0.1197	0.1677	0.0793	0.0594	0.1464
0.6	0.0143	0.1086	0.1310	0.0786	0.0447	0.1356
0.8	0.0069	0.0999	0.1064	0.0806	0.0344	0.1323
1	0.0033	0.0940	0.0896	0.0836	0.0252	0.1321
1.2	0.0016	0.0904	0.0781	0.0895	0.0176	0.1376
1.4	0.0010	0.0889	0.0709	0.1026	0.0119	0.1534
1.6	0.0017	0.0892	0.0706	0.1302	0.0076	0.1887
1.8	0.0159	0.0868	0.0940	0.1763	0.0282	0.2606

Table 6 Convergence of the successive over-relaxation method: standardized Frobenius norm ν_{5000} (Eq. 36) for the variance-covariance matrix

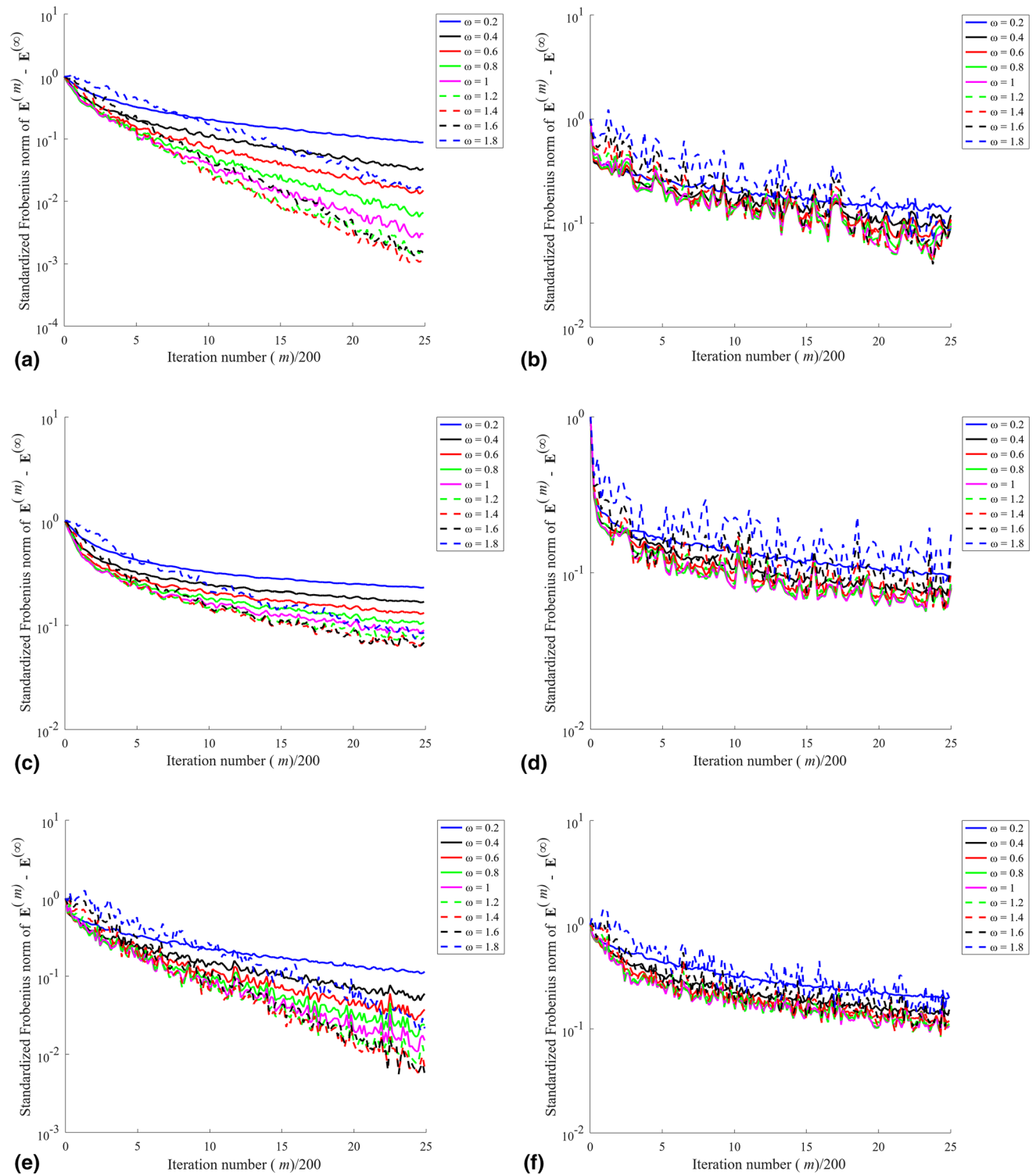| ω	Spherical (range 10)	Spherical (range 50)	Cubic (range 10)	Cubic (range 50)	Exponential	Matérn
0.2	3.57E−3	3.36E−3	4.13E−2	2.57E−3	2.63E−3	3.64E−3
0.4	5.94E−4	3.23E−3	2.26E−2	1.76E−3	1.09E−3	2.36E−3
0.6	1.55E−4	3.20E−3	1.43E−2	1.67E−3	6.07E−4	2.04E−3
0.8	4.73E−5	3.18E−3	1.01E−2	1.84E−3	3.69E−4	1.97E−3
1	1.35E−5	3.24E−3	7.93E−3	2.24E−3	2.14E−4	2.11E−3
1.2	4.02E−6	3.43E−3	6.82E−3	2.91E−3	1.17E−4	2.43E−3
1.4	2.17E−6	3.86E−3	6.47E−3	3.99E−3	7.12E−5	3.08E−3
1.6	5.30E−6	5.02E−3	7.12E−3	6.13E−3	8.33E−5	4.57E−3
1.8	2.71E−4	1.07E−2	1.25E−2	1.35E−2	7.33E−4	9.86E−3

algorithms are applicable to large-scale problems. As an illustration, Fig. 5 shows one realization of a stationary Gaussian random field with an isotropic spherical covariance of range 100 units on a grid with 500×500 nodes, conditioned to $o = 10,000$ data whose coordinates have been randomly chosen over the target grid and whose values have been generated by the Cholesky decomposition algorithm. The non-conditional simulation is obtained by Gibbs sampling with $K = 250,000$ iterations (1 permutation over all the grid nodes), a relaxation parameter $\rho = -0.6$ and the updating of $p = 5 \times 5$ adjacent nodes at each iteration. As for the conditioning process, it is obtained with a relaxation parameter $\omega = 1.2$ and $M = 2,000,000$ iterations (200 permutations over the conditioning data set). The resulting realization exhibits a continuous spatial structure with an irregular short-scale behavior, consistent with the covariance model (spherical). Also, it almost perfectly matches the conditioning data, with an average absolute deviation between data values and simulated values at the data locations equal to 0.0025 and a maximum absolute deviation of 0.0287, corroborating the

convergence of the successive over-relaxation method for the conditioning process.

The presented iterative algorithms have the following advantages over existing methods, as summarized in Table 7:

(1) *Versatility*: there is no restriction on the covariance structure of the target random vector. Accordingly, the algorithms can be used to simulate any type of Gaussian random field (stationary or not, scalar-valued or vector-valued) in any space (Euclidean space of any dimension, sphere, sphere crossed with time, etc.) at finitely many locations that can be evenly distributed or not. The proposed algorithms therefore offer a very general solution to the simulation of Gaussian random vectors and random fields, whereas most of other algorithms (in particular, turning bands, continuous and discrete spectral algorithms) are applicable only in Euclidean spaces, for regularly-spaced locations or covariance models that are stationary or have a known spectral
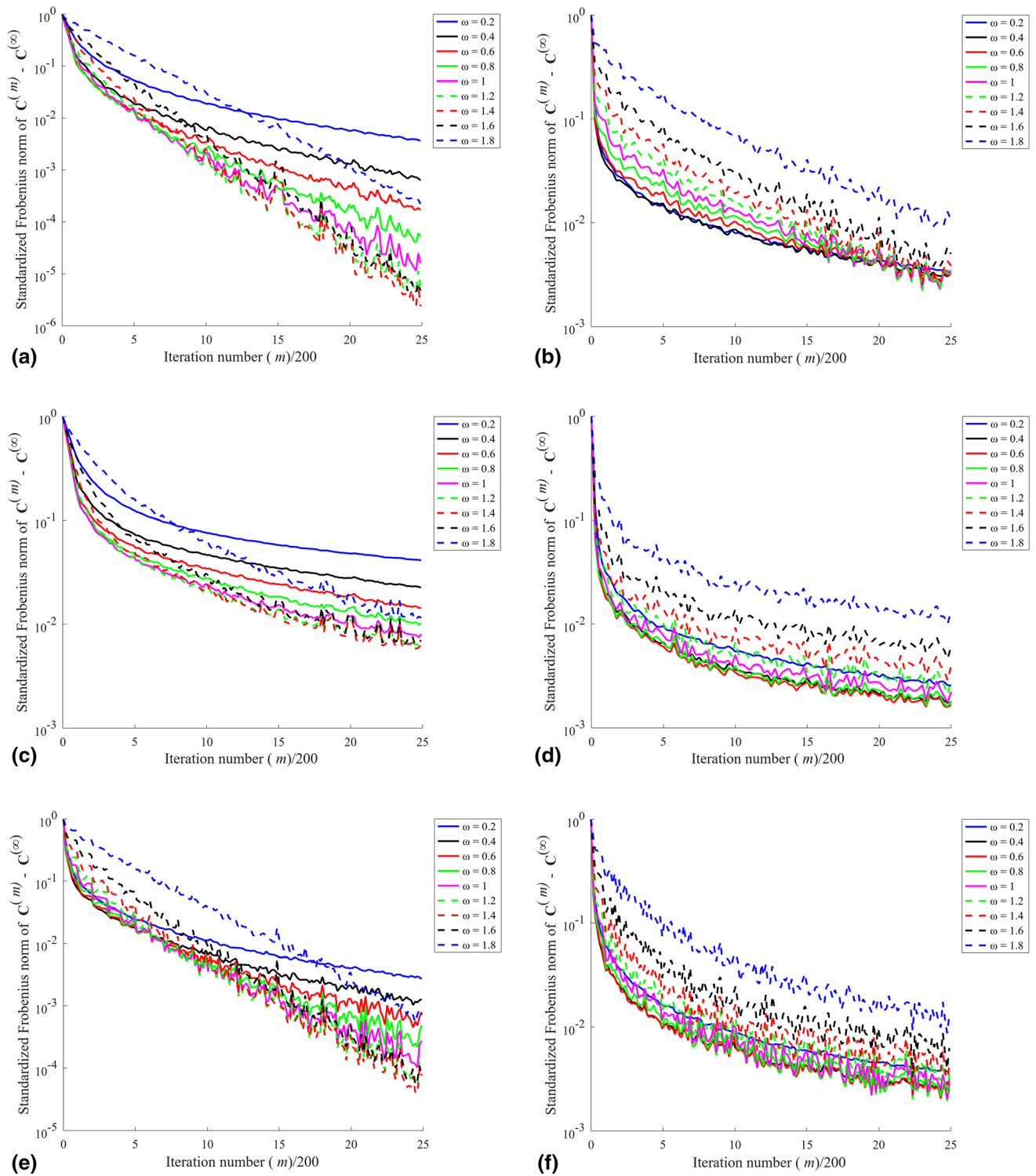
Fig. 3 Convergence of the successive over-relaxation method: standardized Frobenius norm μ_m for m varying between 0 and 5000 and ω varying between 0.2 and 1.8. **a** Spherical covariance with range 10, **b** spherical covariance with range 50, **c** cubic covariance with range 10, **d** cubic covariance with range 50, **e** exponential covariance with scale parameter 10, **f** Matérn covariance with spatially varying scale and shape parameters

representation (Lantuéjoul 2002; Chilès and Delfiner 2012).

(2) The numerical cost is proportional to the number of target locations (n) and to the number of iterations

Fig. 4 Convergence of the successive over-relaxation method: standardized Frobenius norm ν_m for m varying between 0 and 5000 and ω varying between 0.2 and 1.8. **a** Spherical covariance with range 10, **b** spherical covariance with range 50, **c** cubic covariance with range 10, **d** cubic covariance with range 50, **e** exponential covariance with scale parameter 10, **f** Matérn covariance with spatially varying scale and shape parameters

(which can itself be proportional to the number of locations), i.e., $O(n^2)$, thus cheaper than other one-size-fits-all approaches for simulating random vectors such as the Cholesky decomposition algorithm whose complexity is $O(n^3)$ (same complexity for the traditional Gibbs sampler, which requires

(a)

(b)
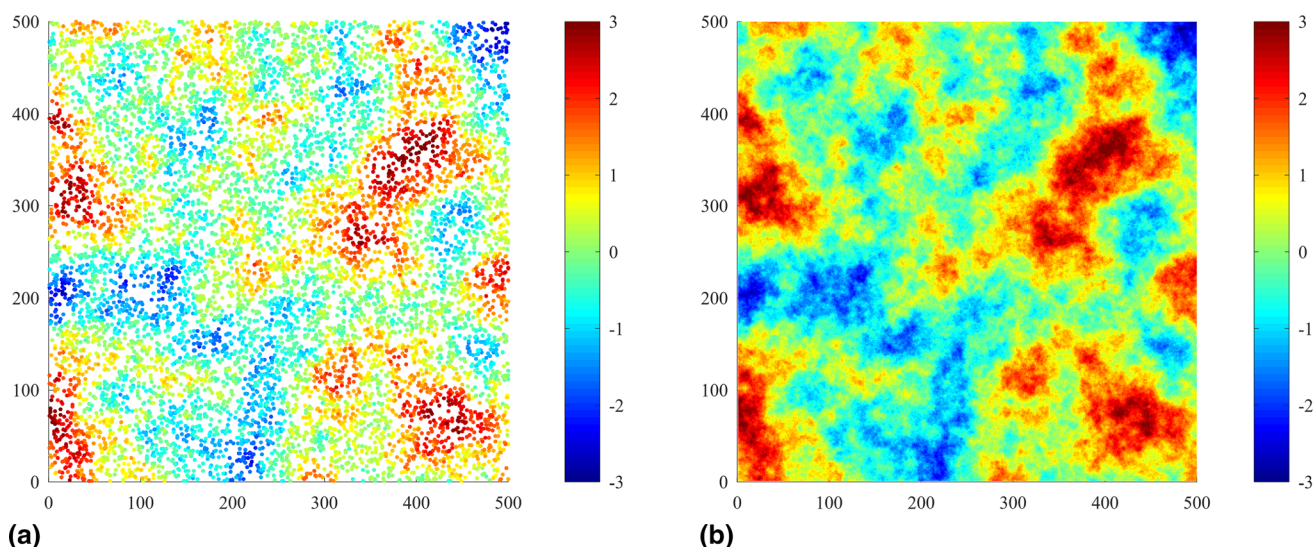
Fig. 5 Ten thousand conditioning data (**a**) and a conditional simulation (**b**) obtained by Gibbs sampling (stopped after $K = 250,000$ iterations with a 5×5 blocking strategy) followed with the successive over-relaxation method (stopped after $M = 2,000,000$ iterations) over a grid with 500×500 nodes. The covariance model is spherical with range 100 units

Table 7 Properties of current algorithms for simulating Gaussian random vectors and random fields

Algorithm	Restriction on covariance structure	Restriction on space definition	Restriction on target locations	Numerical complexity	Memory storage requirements
Covariance matrix (LU, Cholesky or square root) decomposition	None	None	None	$O(n^3)$	$O(n^2)$
Circulant-embedding and discrete spectral	Stationary covariance whose periodic repetition is still a valid covariance	Euclidean spaces and spheres	Evenly spaced	$O(n\ln(n))$	$O(n)$
Autoregressive and moving average	Stationary covariance belonging to a specific subclass	Euclidean spaces	Evenly spaced	$O(n)$	$O(1)$
Continuous spectral	Covariance with known spectral representation	Euclidean spaces and spheres	None	$O(n)$	$O(1)$
Turning bands	Stationary covariance belonging to a specific subclass	Euclidean spaces	None	$O(n)$	$O(1)$
Traditional Gibbs sampler	None	None	None	$O(n^3)$	$O(n^2)$
Propagative Gibbs sampler (our proposal)	None	None	None	$O(n^2)$	$O(n)$

inverting **C**). Also, several realizations can be obtained simultaneously by running the propagative Gibbs sampler with different initial states but the same choice of the subsets (I, J) at each iteration, which allows saving computing time with respect to the alternative of iterating the sampler K times to reach convergence and then retaining the states obtained after multiple of q iterations, based on the mixing property. The successive over-relaxation method also allows conditioning several realizations with a single run with the same choice of the index j at each iteration.

(3) The memory requirements are affordable and considerably smaller than that of Cholesky decomposition and traditional Gibbs sampling, as the storage of the covariance matrix **C** or of its inverse, square root or Cholesky factor is not needed: each iteration only requires the knowledge of one or a few (p) columns of **C**, which can therefore be calculated 'on the fly'. Memory limitations are less severe for the simulation of stationary random fields on regular grids, in which

case (unless a huge grid is considered) the entries of **C** can be calculated once for all and stored before the iteration phase.

The previous results open several perspectives, the study of which deserves further research:

(1) The use of a non-homogeneous Markov chain for the Gibbs sampler, where the transition kernel depends on the iteration k. In particular, it may be interesting to take a non-constant relaxation parameter $\rho(k)$, for instance, negative for the first iterations and progressively tending to zero as the iteration number increases, a scheme suggested by the convergence results displayed in Fig. 1.

(2) Likewise, for the successive over-relaxation method, the relaxation parameter ω could also depend on the iteration number m. Also, instead of systematically looping over the observation subset O (step 2a in the presentation of the Gauss-Seidel method), the index j may be selected randomly and non-uniformly, for instance, with a probability that is all the higher as the deviation between $\mathbf{Y}_{CS,j}^{(m-1)}$ and $\mathbf{y}_j$ is large. This way, the updating would preferentially focus on the data for with the highest mismatch between the observed and simulated values.

(3) A blocking strategy (updating p vector components, with $p > 1$, at each iteration) could also be designed for the successive over-relaxation method, as it is done for the Gibbs sampler.

(4) Another idea would be to use the successive over-relaxation method (Eq. 28) to improve the quality of a conditional simulation obtained with any approximate algorithm, e.g., a non-conditional simulation that has been turned into a conditional one by means of a kriging within a moving neighborhood. It is not obvious that such a procedure would be successful, insofar as the convergence of the method is guaranteed only if the initial vector $\mathbf{Y}_{CS}^{(0)}$ is associated with a dual vector $\mathbf{X}_{CS}^{(0)}$ such that $\mathbf{X}_{CS,U}^{(0)}$ constitutes a non-conditional simulation of $\mathbf{X}_U$ (Eq. 21). A better idea would be the following:

   (a) Non-conditionally simulate $\mathbf{X}$ by Gibbs sampling. Obtain a dual vector $\mathbf{X}_{NCS}$ and calculate the associated direct vector $\mathbf{Y}_{NCS}$.

   (b) Approximately condition $\mathbf{X}_{NCS,O}$ by kriging within a moving neighborhood (add the kriged residual $\mathbf{y} - \mathbf{Y}_{NCS,O}$ to the non-conditional simulation $\mathbf{X}_{NCS,O}$), see Eq. (22). For instance, if kriging is performed with a neighborhood containing only one data, it suffices to replace $\mathbf{C}_{OO}^{-1}$ in Eq. (22) by the identity matrix. Obtain a dual vector $\mathbf{X}_{CS}^{(0)}$ and calculate the associated direct vector $\mathbf{Y}_{CS}^{(0)}$.

   (c) Apply the successive over-relaxation method (Eq. 28) with vector $\mathbf{Y}_{CS}^{(0)}$ obtained in the previous step as the initial vector. This initial vector is 'closer' to the desired conditional vector than the non-conditional vector $\mathbf{Y}_{NCS}$, so that the convergence of the method should be faster.

5 Conclusions

Two iterative algorithms for simulating a Gaussian random vector, without (Gibbs sampling) or with (Gibbs sampling followed with successive over-relaxation) conditioning data, have been presented. Both algorithms provide a simulated Gaussian random vector that converges in distribution to the desired random vector and the convergence can be made faster with a suitable choice of the relaxation parameter.

The experimental results suggest that, most often, the optimal relaxation parameter of the Gibbs sampler is negative, whereas that of the successive over-relaxation method is greater than 1. Recommended values could be −0.6 and 1.2, respectively, as a rule of thumb. If several non-conditional realizations are drawn from the same run (by retaining the vectors obtained at iterations K, $K + q$, $K + 2q$, etc.) based on the mixing property, then the presented numerical experiments suggest to set the relaxation parameter to zero after the burn-in period.

The algorithms can be applied in very general settings to simulate stationary or non-stationary scalar or vector random fields at a set of gridded or non-gridded locations in any Euclidean or non-Euclidean space.

Appendix 1

Let $\mathbf{X}^{(0)} = \mathbf{0}$ and, for any positive integer k, $\mathbf{X}^{(k)}$ be the random vector defined as per Eqs. (1) and (3). The sequence $\{\mathbf{X}^{(k)}: k = 0, 1, 2, \ldots\}$ so obtained constitutes a Markov chain. It is easy to check that this chain is homogeneous (for $k \geq 1$, the distribution of $\mathbf{X}^{(k)}$ knowing $\mathbf{X}^{(k-1)}$ does not depend on k), irreducible (because 1–

$\rho^2 \neq 0$, any nonempty open set of $\mathbb{R}^n$ can be reached by the chain after finitely many iterations) and aperiodic.

Accordingly, to prove that the chain converges in probability to $\mathbf{X}$, it remains to show that the distribution of $\mathbf{X}$ is invariant under the transition kernel of the chain (Lantuéjoul 2002). Suppose that $\mathbf{X}^{(k-1)}$ is a Gaussian random vector with zero mean and variance-covariance matrix $\mathbf{B}$. In such a case, the simple kriging error $-\mathbf{C}_{JJ}^{-1} \mathbf{C}_{JI} \mathbf{X}_I^{(k-1)} - \mathbf{X}_J^{(k-1)}$ and $\mathbf{S}_J \mathbf{U}^{(k)}$ are two independent Gaussian random vectors with zero mean and variance-covariance matrix $\mathbf{\Sigma}_J$ and are independent of $\mathbf{X}_I^{(k-1)}$. $\mathbf{R}^{(k)}$, as defined by Eq. (3), is therefore a Gaussian random vector independent of $\mathbf{X}_I^{(k-1)}$, with zero mean and variance-covariance matrix $\mathbf{\Sigma}_J$, irrespective of the choice of ρ. The proof by Arroyo et al. (2012) can be adapted to establish that, under these conditions, $\mathbf{X}^{(k)}$ is a Gaussian random vector with zero mean and variance-covariance matrix $\mathbf{B}$, Q.E.D.

References

Adler SL (1981) Over-relaxation method for the Monte Carlo evaluation of the partition function for multiquadratic actions. Phys Rev D 23(12):2901–2904

Alabert F (1987) The practice of fast conditional simulations through the LU decomposition of the covariance matrix. Math Geol 19(5):369–386

Alfaro M (1980) The random coin method: solution to the problem of the simulation of a random function in the plane. Math Geol 12(1):25–32

Armstrong M, Galli AG, Beucher H, Le Loc'h G, Renard D, Doligez B, Eschard R, Geffroy F (2011) Plurigaussian simulations in geosciences, 2nd edn. 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

Barone P, Frigessi A (1990) Improving stochastic relaxation for Gaussian random fields. Probab Eng Inf Sci 4(3):369–389

Box GEP, Jenkins GM (1976) Time series analysis: forecasting and control, Revised edn. Holden-Day, Oakland

Chellappa R, Jain A (1992) Markov random fields: theory and application. Academic Press, London

Chilès JP, Allard D (2005) Stochastic simulation of soil variations. In: Grunwald S (ed) Environmental soil-landscape modeling: geographic information technologies and pedometrics. CRC Press, Boca Raton, pp 289–305

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

Davis MW (1987) Production of conditional simulations via the LU triangular decomposition of the covariance matrix. Math Geol 19(2):91–98

Delfiner P, Chilès JP (1977) Conditional simulations: a new Monte Carlo approach to probabilistic evaluation of hydrocarbon in place. In: SPE paper 6985. Society of Petroleum Engineers

Delhomme JP (1979) Spatial variability and uncertainty in groundwater flow parameters: a geostatistical approach. Water Resour Res 15(2):269–280

Dietrich CR, Newsam GN (1993) A fast and exact method for multidimensional Gaussian stochastic simulations. Water Resour Res 29(8):2861–2869

Emery X (2009) The kriging update equations and their application to the selection of neighboring data. Comput Geosci 13(3):269–280

Emery X, Arroyo D (2018) On a continuous spectral algorithm for simulating non-stationary Gaussian random fields. Stoch Environ Res Risk Assess 32(5):905–919

Emery X, Lantuéjoul C (2006) TBSIM: a computer program for conditional simulation of three-dimensional Gaussian random fields via the turning bands method. Comput Geosci 32(10):1615–1628

Emery X, Peláez M (2011) Assessing the accuracy of sequential Gaussian simulation and cosimulation. Comput Geosci 15(4):673–689

Emery X, Arroyo D, Peláez M (2014) Simulating large Gaussian random vectors subject to inequality constraints by Gibbs sampling. Math Geosci 46(3):265–283

Emery X, Arroyo D, Porcu E (2016) An improved spectral turning-bands algorithm for simulating stationary vector Gaussian random fields. Stoch Environ Res Risk Assess 30(7):1863–1873

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 J (1991) Efficient simulation from the multivariate normal and Student-t distributions subject to linear constraints and the evaluation of constraint probabilities. In: Keramidas EM, Kaufman SM (eds) Computing science and statistics: proceedings of the 23rd symposium on the interface. Interface Foundation of North America, Fairfax Station, pp 571–578

Green PJ, Han X (1992) Metropolis methods, Gaussian proposals and antithetic variables. In: Barone P, Frigessi A, Piccioni M (eds) Stochastic models, statistical methods, and algorithms in image analysis. Springer, New York, pp 142–164

Guyon X (1995) Random fields on a network: modeling, statistics, and applications. Springer, New York

Journel AG, Huijbregts CJ (1978) Mining geostatistics. Academic Press, London

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: Presented at the 9th international geostatistics congress, Oslo. http://geostats2012.nr.no/pdfs/1747181.pdf. Accessed December 28, 2019

Marcotte D, Allard D (2018a) Half-tapering strategy for conditional simulation with large datasets. Stoch Environ Res Risk Assess 32(1):279–294

Marcotte D, Allard D (2018b) Gibbs sampling on large lattice with GMRF. Comput Geosci 111:190–199

Matérn B (1986) Spatial variation-stochastic models and their application to some problems in forest surveys and other sampling investigations. Springer, Berlin

Matheron G (1973) The intrinsic random functions and their applications. Adv Appl Probab 5(3):439–468

Neal RM (1998) Suppressing random walks in Markov chain Monte Carlo using ordered overrelaxation. In: Jordan MI (ed) Learning in graphical models. Kluwer Academic, Dordrecht, pp 205–225

Pakman A, Paninski L (2014) Exact Hamiltonian Monte Carlo for truncated multivariate Gaussians. J Comput Graph Stat 23(2):518–542

Pardo-Igúzquiza E, Chica-Olmo M (1993) The Fourier integral method: an efficient spectral method for simulation of random fields. Math Geol 25(2):177–217

Rue H (2001) Fast sampling of Gaussian Markov random fields. J R Stat Soc Ser B (Stat Methodol) 63(2):325–338

Safikhani M, Asghari O, Emery X (2017) Assessing the accuracy of sequential Gaussian simulation through statistical testing. Stoch Environ Res Risk Assess 31(2):523–533

Shinozuka M (1971) Simulation of multivariate and multidimensional random processes. J Acoust Soc Am 49(1B):357–367

Shive PN, Lowry T, Easley DH, Borgman LE (1990) Geostatistical simulation for geophysical applications—Part 2: geophysical modeling. Geophysics 55(11):1441–1446

Webster R, Oliver MA (2007) Geostatistics for environmental scientists, 2nd edn. Wiley, New York

Whitmer C (1984) Over-relaxation methods for Monte Carlo simulations of quadratic and multiquadratic actions. Phys Rev D 29(2):306–311

Wilhelm S, Manjunath BG (2010) tmvtnorm: a package for the truncated multivariate normal distribution. R J 2(1):25–29

Young DM (2003) Iterative solution of large linear systems. Dover Publications, New York