THÈSE DE DOCTORAT

de l'Université de recherche Paris Sciences et Lettres
PSL Research University

Préparée à MINES ParisTech

Méthodes stochastiques pour la modélisation d'incertitudes sur les maillages non structurés

Ecole doctorale n°432

SCIENCES DES METIERS DE L'INGENIEUR

Spécialité GEOSTATISTIQUE

COMPOSITION DU JURY :

Prof. Guillaume CAUMON
Université de Lorraine, Président

Prof. Xavier EMERY
Université du Chili, Rapporteur

Dr. Mickaële LE RAVALEC
IFP-EN, Rapporteur

Dr. Denis ALLARD
INRA, Examinateur

Dr. Pierre BIVER
Total S.A., Examinateur

Dr. Hans WACKERNAGEL
MINES ParisTech, Examinateur

Soutenue par Victor **ZAYTSEV**
le 12 septembre 2016

Dirigée par **Hans Wackernagel**

MINES ParisTech | PSL RESEARCH UNIVERSITY PARIS

Contents

Chapter 1: Introduction

Résumé

Dans ce chapitre les simulations géostatistiques sont présentées dans un cadre général. Les applications des simulations géostatistiques sont possibles dans plusieurs domaines notamment pour la modélisation des gisements dans les industries minière et pétrolière. La différence entre l'estimation et la simulation est discutée.

1.1 General context of geostatistical simulations

Geostatistical simulations provide a framework to address a wide range of problems related to natural resources. The common application of geostatistical simulations consists in generating realistic realizations of a spatial or a spatio-temporal phenomenon and using these to evaluate some response functions. In this context, geostatistical simulations are used as part of Monte Carlo methods. Applications of geostatistical simulations can be found in such industries as mining, petroleum, meteorology and in ecological monitoring.

A classical problem for which geostatistical simulations are applied is modeling the fluid flow through a heterogeneous reservoir. This is one of the principal modeling problems for petroleum industry (where the fluids are oil, water and gas), but this problem can also arise in other contexts, such as modeling the groundwater-travel times from a nuclear repository to the surface (Goovaerts et al. 1997). Prediction methods do not provide realistic results for modeling the heterogeneities in petroleum reservoirs since they do not reproduce correctly the spatial variability of such parameters as porosity and permeability due to the interpolation smoothing effects. Only the simulation approach enables to reproduce the realistic spatial variability and thus enables to derive more realistic models of the subsurface.

Algorithms for geostatistical simulation described in the literature mostly deal with regular grids – grids composed of identical blocks. A new generation of geology modeling tools deals with unstructured grids – the grids composed of blocks of various shapes and sizes, the most usual types of blocks being Voronoï polygon prisms, tetrahedrons and triangular prisms. Unstructured grids enable building models with adaptive resolution – the blocks of the model can be smaller in the regions of particular practical interest and coarser in the less important regions. For instance, a petroleum reservoir can be modeled with fine blocks in the vicinity of the wells in order to solve more accurately the flow equations, whereas the aquifier can be modeled with lower resolution in order to reduce the computation time. This new generation of the subsurface models based on unstructured grids requires specialized methods for geostatistical simulations, since the classical methods designed for regular grids either cannot be applied at all or do not reproduce correctly

the statistical properties of the model. This thesis provides a solution to the problem of geostatistical simulation on unstructured grids with change of support effect.

1.2 Describing random fields

A complete description of a random field (RF) consists in defining all of its finite-dimensional distributions, which can be done by means of the multivariate cumulative distribution functions (CDF). Thus, in order to describe the RF $Z(x)$, for every n and every set of points $x_1, \dots x_n \in D$ the finite-dimensional CDF should be given

$$F_{x_1,\dots x_n}(z_1, \dots, z_n) = P(Z(x_1) \leq z_1, \dots, Z(x_n) \leq z_n) \quad (1.1)$$

In practice defining a complete set of finite-dimensional CDFs is almost never possible, except for the few known analytical models such as multivariate Gaussian RF. For that reason, the random fields in geostatistics are often characterized only with respect to the first and second order moments. Thus, defining a RF reduces to defining the marginal distribution $F_x(z) = P(Z(x) \leq z)$ at every point $x \in D$ and defining the covariance function $C(x, x') = cov\big(Z(x), Z(x')\big)$ for all pairs of points $(x, x') \in D$. Certainly, it is necessary to assume the existence of the first and second order moments for $Z(x)$. This definition through the marginal distribution and covariance in the general case does not determine a random field in a unique manner. Chilès and Lantuéjoul (2005) demonstrate three different random set models with the same bivariate and even trivariate distributions. Moreover, there is no general criterion which verifies if a random field with a given marginal distribution and covariance exists or not, although it is known that certain marginal distributions and covariance functions are not compatible. For example, as demonstrated by Matheron (1989), lognormal distribution is not compatible with a spherical covariance function. More precisely, RF $Z(x)$, such that $Y(x) = \ln Z(x)$ is multivariate Gaussian cannot have a spherical covariance function. To conclude, when random fields are considered only with regards to marginal distribution and covariance functions, classes of equivalence of random fields are considered.

1.3 Formal definition of geostatistical simulation

The characterization of the random field has direct implications on the definition of the geostatistical simulation. As noted by Chilès and Delfiner (2012), two definitions can be used for unconditional geostatistical simulation:

Definition 1
Simulation of RF $Z(x)$ is a realization of any RF $S(x)$ with the same multivariate distribution as $Z(x)$.

Definition 2

Simulation of RF $Z(x)$ is a realization of any RF $S(x)$ with the same marginal distribution $F(z)$ and covariance function $C(x, x')$.

For both definitions of unconditional simulations, conditioning to known data $\{(d_i, z_i), i = 1 \dots N_{data}\}$ results in retaining from the realizations of $S(x)$ only those, which coincide with the known values of at the data locations (Journel & Huijbregts 1978):

$$S(d_i) = z_i, i = 1 \dots N_{data} \tag{1.2}$$

The applicable definition of the geostatistical simulation depends on the manner in which the target RF is characterized. In rare cases when all the finite-dimensional CDF of $Z(x)$ are known analytically, as for multivariate Gaussian RF, the simulation goal can be reproduction of the multivariate distributions of $Z(x)$, and Definition 1 can be used.

Another case when Definition 1 is applied is when dealing with simulations with a training image. Considering that the training image coupled with the simulation algorithm are able to characterize the complete multivariate distribution of $Z(x)$ enables simulating more complicated spatial features of random fields, than those which can be defined only with marginal distribution and covariance (Guardiano & Srivastava 1993; Strebelle 2002). The recent advances in the theory of simulations with training image enable taking into account additional constraints and simulating non-stationary random fields which are more complex than those defined only through a covariance function and marginal distribution (Chugunova & Hu 2008; Hu & Chugunova 2008; Mariethoz & Caers 2014).

When $Z(x)$ is defined only through the first and second moments, Definition 2 should be used. This definition is less restrictive than Definition 1, which implies that different models of random fields can be used for simulating $Z(x)$ as long as they respect first and second order moments of $Z(x)$. For instance, direct sequential simulation algorithms (Oz et al. 2003; Soares 2001) and algorithms based on the normal score transform, such as sequential Gaussian simulation (SGS), Cholesky decomposition, spectral simulation and turning bands (Chilès & Delfiner 2012; Goovaerts 1997) represent two classes of algorithms based on different RF models which both aim at simulating random fields with given marginal distribution and covariance function.

This thesis focuses on simulating random fields that are characterized through the marginal distribution and a covariance function, thus, Definition 2 is used for geostatistical simulations.

1.4 Data analysis

A brief mention of the geostatistical data analysis methods is necessary, since it is from data analysis that the inputs for geostatistical simulations are obtained. In addition, many assumptions of geostatistics originate from the necessity of performing the data analysis on a small amount of available samples and these assumptions in turn have impact on the description of the random field under investigation.

In practice the description of random fields in geostatistics is often even more restrictive than defining the marginal distribution $F_x(z)$ and the covariance function $C(x, x')$ due to the fact that these inputs are stemming from a data analysis of a relatively small number of observed samples $\{(d_i, z_i), i = 1 \dots N_{data}\}$. These data do not enable deriving a point-dependent marginal distribution $F_x(z)$ at every point $x \in D$. Instead, a single marginal distribution $F(z)$ in the region D is imposed, and for the purpose of data analysis the known data values $\{z_i, i = 1 \dots N_{data}\}$ are often treated as independent realizations of RV Z with a CDF $F(z)$. From analysis of the histogram of these data the empirical density function and empirical CDF of Z are derived.

Another assumption related to the small quantity of available data is the second order stationarity for $Z(x)$, see Chilès and Delfiner (2012); Journel and Huijbregts (1978); Lantuejoul (2002). It imposes that the covariance function $C(x, x')$ depends only on the separation vector $\vec{h} = x' - x$. This assumption is absolutely of practical importance since in the general case it is the only way to obtain a sufficient amount of data to estimate $C(x, x')$. Under the assumption of second order stationarity for all data points (d_i, d_j) which are separated by the same vector $\vec{h}$, the data values at these points (z_i, z_j) are considered to be realizations of random vectors $\left(Z(d_i), Z(d_j)\right)$ with the same first and second order moments. The following formula (Journel & Huijbregts 1978) gives an estimate for the covariance function for any two points, separated by the vector $\vec{h}$:

$$\hat{C}(\vec{h}) = \frac{1}{N_h} \sum_{d_i - d_j \approx \vec{h}} z_i z_j - m^2, \tag{1.3}$$

where m is the estimated mean value of $Z(x)$ and N_h is the number of pairs of data separated by the vector $\vec{h}$. In practice, a certain degree of tolerance is introduced into the estimation procedure in order to increase the number of samples. This tolerance depends on the nature of the problem under investigation and is usually a subjective choice of the person who conducts the data analysis. Introducing the tolerance corresponds to using the sign “approximately equal” in Eqn. (1.3). An analytical covariance function model is usually fitted for the estimated covariance function $\hat{C}(\vec{h})$.

As it is noted in Journel and Huijbregts (1978), the estimator in Eqn. (1.3) is biased since the mean m is also estimated from the available data. In the general case it is more advantageous to estimate the variogram $\gamma(\vec{h}) = \frac{1}{2}Var\left[Z(x+\vec{h}) - Z(x)\right]$ through the formula

$$\hat{\gamma}(\vec{h}) = \frac{1}{2N_h}\sum_{d_i - d_j \approx \vec{h}} \left(z_i - z_j\right)^2, \tag{1.4}$$

fit the analytical model to it and use the theoretical relation (Chilès & Delfiner 2012) between the variogram and the covariance:

$$C(\vec{h}) = \sigma^2 - \gamma(\vec{h}), \tag{1.5}$$

where σ^2 is the variance of $Z(x)$. Applying formula (1.5) is only possible when $Z(x)$ is second order stationary.

The data analysis procedures described above impose that the distribution of the random field $Z(x)$ is characterized by the marginal CDF $F(z)$ which is unique for all points $x \in D$ and a covariance function $C(\vec{h})$ which depends only on the separation vector between two points: $\vec{h} = x' - x$. We assume the second order stationarity assumption for testing the algorithms presented in this work since it simplifies the approach of modeling the covariance function $C(x, x') = C(\vec{h})$. However, this assumption is not mandatory and to underline this fact, the general notation $C(x, x')$ is used for covariance functions.

The empirical CDF is a step function, which often does not reflect the continuous nature of the physical parameters simulated such as porosity and permeability. There are two practically used approaches to smoothen the empirical CDF function. The first approach is CDF-based: it results in direct smoothing of the empirical CDF $\hat{F}(z)$, or interpolating between the steps in order to obtain the CDF $F(z)$ which corresponds to a continuous property. Another approach is density-based; it consists in finding a smooth density function which approximates the empirical density. In the density-based approach the CDF $F(z)$ is derived through integration of the fitted (usually even analytical) smooth density function. The density-based approach is often preferred in the industry, since it enables working in terms of relative frequency of samples, rather than in terms of probability of exceeding some threshold.

The data analysis procedures are schematically illustrated on Figure 1-1 for a synthetic data set which consists of 100 measurements of porosity on a territory of $10 \times 10\ km^2$, see Figure 1-1(a). The histogram of the sample data is given at Figure 1-1(c) and a smooth density model is fitted. The variogram analysis with a fitted spherical covariance model is demonstrated on Figure 1-1(d). Finally, a conditional simulation of a random field which respects the data at sample locations and the

fitted distribution is depicted of Figure 1-1(b). The blue circles indicate the sample locations.

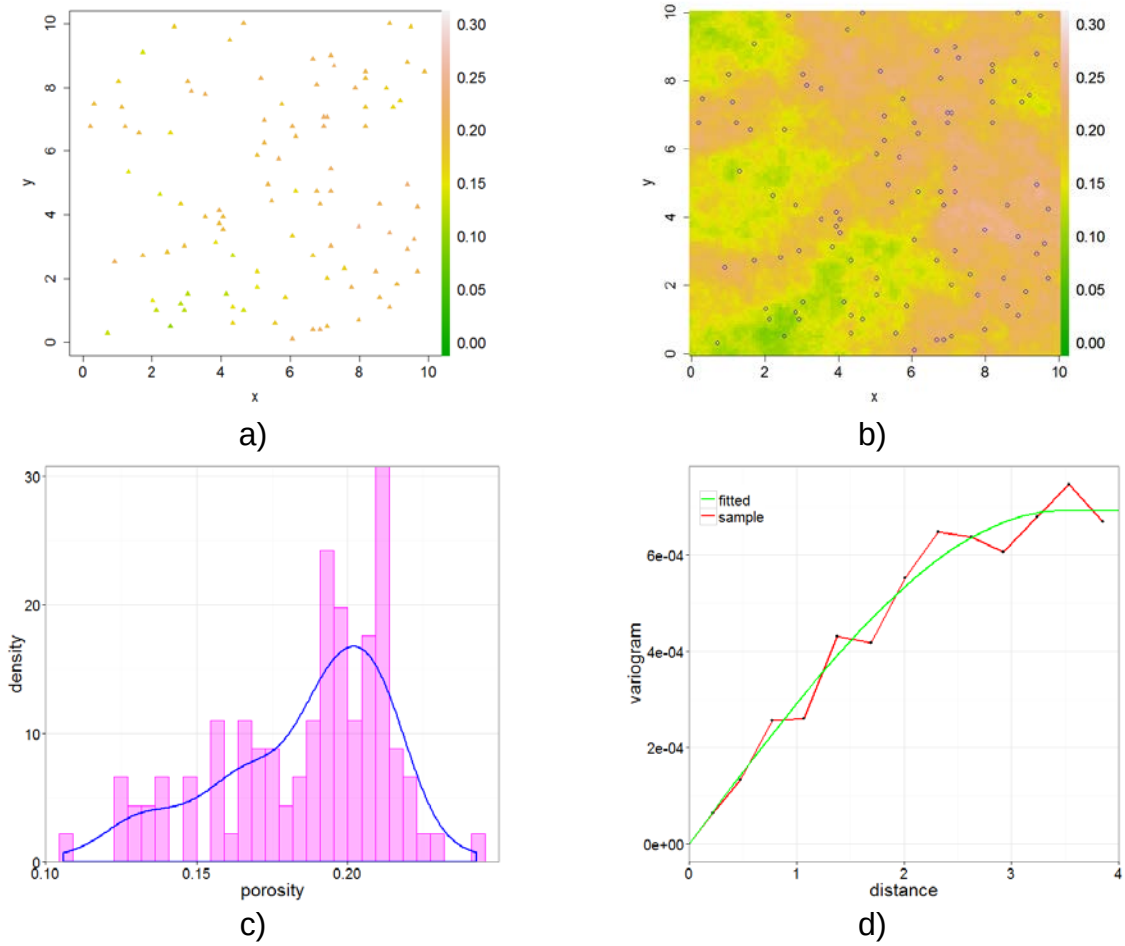


Figure 1-1. Elements of geostatistical simulation workflow a) a set of 100 samples b) conditional simulation c) fitting a smooth density model d) fitting a smooth variogram model variogram analysis

For the purpose of geostatistical simulation and prediction, it is often assumed that the investigated RF $Z(x)$ with a covariance function $C(x, x')$ is a transformation of a multivariate Gaussian RF $Y(x)$ with a covariance function $\rho(x, x')$: $Z(x) = \varphi(Y(x))$, where $\varphi(z)$ is the normal score transform function (Chilès & Delfiner 2012). In that case it is common to perform data analysis for $\rho(x, x')$ with the methods described above. This thesis uses extensively the hypothesis that $Z(x)$ is a transform of a multivariate Gaussian RF $Y(x)$ and hereafter we consider that $\rho(x, x')$ was derived from the data analysis workflow.

1.5 Problem statement

We consider the problem of simulating a random field (RF) $Z(x)$ with a marginal distribution $F(z)$ and a covariance function $C(x, x')$ defined in the region $D \in R^d$, $d = 1,2,3$ on an unstructured grid. If the values of $Z(x)$ are known at some

points of D (i.e. $Z(d_i) = z_i,\ i = 1 \dots N_{data}$), we say that the random field $Z(x)$ is conditioned to data $\{(d_i, z_i), i = 1 \dots N_{data}\}$. An unstructured grid can be defined as a finite number N_b of non-overlapping blocks $\{v_p, p = 1 \dots N_b\}$ on D.

The goal of geostatistical simulation on unstructured grid is to simulate a random vector $\left\{Z(v_p) = \frac{1}{|v_p|}\int_{v_p} Z(x)dx,\ p = 1 \dots N_b\right\}$ of average values of $Z(x)$ over the blocks of the grid which respects two conditions:

1) Correctly reproduces the marginal distribution – the marginal distribution of each component $Z(v_p)$ should coincide with the marginal distribution of $\frac{1}{|v_p|}\int_{v_p} Z(x)dx$. This property is referred to as volume support effect, since the distribution of $Z(v_p)$ is a function of the block v_p.
2) The covariance between each pair of components $Z(v_p)$ and $Z(v_q)$ should coincide with the covariance between two stochastic integrals $\frac{1}{|v_p|}\int_{v_p} Z(x)dx$ and $\frac{1}{|v_q|}\int_{v_q} Z(x)dx$.

The definition of the conditional simulation on unstructured grids is given only in the terms of first and second order statistics, which means that the distribution of the random vector $\{Z(v_p),\ p = 1 \dots N_b\}$ is not defined in a unique manner. One could also remark that this definition does not mention the conditioning data $\{(d_i, z_i), i = 1 \dots N_{data}\}$. In fact, the conditioning to data is included in this definition implicitly, since the RF $Z(x)$ respects the data z_i at locations d_i,as stated in the beginning of the sections.

1.6 Related research

Simulations on unstructured grid can be performed by simulating on a fine scale regular grid followed by upscaling. Any type of simulation method on regular grids can be applied in this case, such as SGS, Cholesky decomposition, spectral simulation and others (Chilès & Delfiner 2012; Goovaerts 1997). This approach is detailed in Chapter 2.

Algorithms based on the direct sequential simulation (DSS) approach can be applied for simulations on unstructured grids. In this work we investigate two DSS algorithms: DSS-1 proposed by Soares (2001) and DSS-HR (direct sequential simulation with histogram reproduction) proposed by (Oz et al. 2003). Application of DSS-HR to the problem of simulations on unstructured grids was proposed by Manchuk et al. (2005). DSS algorithms are also discussed in Chapter 2.

This thesis proposes a theoretical framework for geostatistical simulations on unstructured grids based on the discrete Gaussian model (DGM). Previously DGM was applied to geostatistical simulations on regular grids (Emery 2009; Emery & Ortiz

2011), where DGM was used for conditional simulation and co-simulation. Another application of DGM for geostatistical simulations can be found in Brown et al. (2008), where the DGM is conditional simulation of Cox process on a regular grid with samples defined on various supports. The generalization of the DGM for unstructured grids in this thesis was derived independently of (Brown et al. 2008) and is deemed to be applied in a different context – on unstructured grids and with congruent samples defined on quasi-point support.

1.7 Thesis contributions

The main contributions of this thesis are:

1) Investigating the properties of DSS algorithms and providing simulation parameters for which DSS algorithms fail reproducing expected statistical properties.
2) Proposing two theoretical models (DGM 1 and DGM 2) based on DGM for geostatistical simulations of a continuous variable.
3) Investigating the theoretical difference between DGM 1 and DGM 2 and demonstrating this difference through a simulation test on a synthetic dataset.
4) Demonstrating that DGM-based simulation algorithms are not restricted to utilization of the sequential simulation paradigm and enable robust reproduction of marginal distribution and covariance.
5) Formalizing the problem of simulating discrete variables on unstructured grids and proposing a new theoretical model generalizing the pluri-Gaussian simulation model for facies on unstructured grids.
6) Investigating the problem of computing the block to block covariance on unstructured grids and demonstrating the advantage of Monte Carlo integration methods over other approaches. This research can be applied in a more general scope than geostatistical simulations, in particular, to the problem of prediction through block kriging.

Chapter 2: Geostatistical simulation methods for unstructured grids

Résumé

Différentes méthodes existantes de simulation géostatistique sur les maillages non-structurés sont analysées. Les deux approches principales peuvent être appliquées – la simulation sur un maillage fin régulier suivi par l'upscaling et la simulation «directe» - sans la transformation à l'échelle Gaussienne. Les propriétés de ces méthodes et leurs inconvénients sont examinées.

This chapter summarizes various geostatistical methods that can be applied for simulations on unstructured grids. Two main classes of algorithms are considered – fine scale simulations followed by upscaling, and direct block simulations on unstructured grids. We refer to the fine scale simulation approach as "conventional", since it requires little additional development relative to the geostatistical algorithms on regular grids. The direct block simulation algorithms do not use intermediate regular fine scale grids, and operate directly on the blocks of the target unstructured grid.

2.1 Conventional approach to simulations on unstructured grids

The straightforward approach for simulating on an unstructured grid is to use an auxiliary regular fine scale grid to perform a point support simulation on it and to upscale subsequently the results to the target unstructured grid. The classical inputs to a geostatistical simulation in petroleum industry are point data of a petrophysical property, the corresponding histogram and a covariance model. A number of simulation algorithms exist, whose implementation requires various additional theoretical assumptions.

The classical and the most commonly used assumption about the spatial structure of the random field to be simulated is the multivariate Gaussian assumption, which states that $Z(x)$ is a (usually non-linear) transform of a multivariate Gaussian random field $Y(x)$. Thus, $Z(x) = \varphi(Y(x))$, where $\varphi(y)$ is the Gaussian anamorphosis function (Chilès & Delfiner 2012). The assumption of multigaussianity enables to simulate the Gaussian random field using numerous existing techniques like sequential Gaussian simulation (SGS), see Goovaerts (1997) for details and turning bands (Chilès & Delfiner 2012), among others. The above-mentioned methods provide simulations of a random field which is only approximately multivariate Gaussian due to assumptions used. A spectral simulation method for accurate generation multivariate Gaussian random fields on regular grids in 1D, 2D and 3D was proposed by Pardo-Iguzquiza and Chica-Olmo (1993). The final result of the simulation is obtained by transforming the simulated Gaussian field to the original scale using the anamorphosis φ.

Another approach to the simulation of random fields given a histogram and a covariance is provided by a family of direct sequential simulation (DSS) algorithms. In that case no explicit assumption about the multivariate distribution of $Z(x)$ is done, but a set of conditional distributions is explicitly provided. Different algorithms suggest using different local CDF in the sequential simulation procedure (Oz et al. 2003; Soares 2001). The ability of the direct sequential simulation approach to reproduce the target histogram and covariance was illustrated by Robertson et al. (2006). This approach has the advantage of avoiding the transformation to a Gaussian variable, although it raises problems concerning the internal consistency of the model used. For example, even though it is proven mathematically that a spherical covariance model is not compatible with a multivariate-lognormal random field (Matheron 1989), it is possible to perform a simulation using the DSS algorithm with such a construction. Obviously, in this case, the simulated random field will not correspond to the input parameters.

In the general case not every chain of local CDF defines a valid multivariate distribution function. Theoretical conditions for defining a valid multivariate distribution function from a set of conditional CDFs are provided by Cressie and Wikle (2011). In practice, it is not possible to verify these conditions; however, they form a valuable theoretical result.

After performing a simulation of the target random field $Z(x)$ on a fine scale grid whose cells can be considered as point support, the results are upscaled on the target unstructured grid. The approach of using a fine scale grid for simulations has the advantage of avoiding assumptions about the change of support law for the random field $Z(x)$ as well as assumptions about spatial distribution of the block average values $Z(v)$. The fine scale simulations are especially efficient for simulating the variables that do not average linearly over the volume, such as permeability. In that case, the results of fine scale simulation can be upscaled on an unstructured grid using one of the numerical methods (Farmer 2002). The simulation algorithms which operate directly on blocks have to introduce additional assumptions and transform permeability to a variable that averages linearly.

From the practical point of view, using fine scale grids and upscaling on an unstructured grid does not always seem an optimal solution. This approach has a set of disadvantages such as creating and storing an auxiliary fine scale grid, increasing the number of locations at which the random field should be simulated and transferring the results from the fine regular to the unstructured grid – a process which can be time demanding and results in artifacts if the chosen refinement level was not sufficient for the given problem. When dealing with fine scale regular grids it remains a subjective decision which level of refinement should be chosen. It is clear that the size of the blocks in the fine scale regular grid should be at least as small as the smallest block in the target unstructured grid (otherwise there will not be any simulated values for the smallest block of the unstructured grid). This consideration

leads to necessity of using fine scale grids with several millions of blocks even if the target unstructured grid has a relatively small number of blocks, but high variability in the block sizes.

2.2 Direct sequential simulation on blocks

The goal of geostatistical simulation on block support is to avoid point support simulations and subsequent upscaling, and rather to simulate a random vector $\{Z(v_i), i = 1 \dots N\}$, where $Z(v)$ denotes the average value of $Z(x)$ over a grid block v. The classical multigaussian formalism is not applicable on the block support, since the data do not average linearly after the normal score transform.

The family of DSS algorithms does not use the normal score transform and could be applied to simulations directly on the block support by merely replacing the kriging step of the SGS algorithm by a block kriging step. From the theoretical standpoint DSS is based on two mathematical methods: sequential simulation (Goovaerts 1997) and simple kriging principle (Journel 1993). The sequential simulation paradigm states that the simulation of a multivariate random vector can be done in a sequential manner. It is based on the well-known decomposition $P(A_1,A_2,\dots,A_N) = P(A_1)P(A_2 \mid A_1)P(A_N \mid A_1,\dots A_{N-1})$ where P is a probability measure and $(A_1,A_2,\dots A_N)$ is any family of events (Shiryaev 1996). It can be applied whenever all conditional distributions of a variable given all previously simulated variables are known. This decomposition is the basis for the classical SGS algorithm, in which the full conditional distribution is usually approximated by a conditional distribution using only the nearest previously simulated points. The simple kriging principle states that a sequential simulation procedure is able to reproduce the target covariance model as soon as the mean and variance of the conditional distribution at the iterations of the algorithm are determined by simple kriging. It is the simple kriging principle that enables correct reproduction of the covariance relations between blocks determined by the point-to-point covariance model $C(h)$. The possibility of applying two DSS algorithms DSS 1 (Soares 2001) and DSS-HR (Oz et al. 2003) for geostatistical simulations on unstructured grids is considered below. When the simulated variable $Z(x)$ has a Gaussian marginal distribution DSS 1 and DSS-HR coincide with the classical SGS. Therefore one should examine the applicability of these algorithms for simulating distributions which depart from the Gaussian assumption, such as strongly skewed or heavy tailed distributions.

For testing purpose, consider a random field $Z(x)$ following a lognormal distribution with logarithmic mean 0 and logarithmic variance 1. Its variance is thus $\sigma^2 = e(e-1)$. Since one of the key requirements to a geostatistical simulation algorithm on an unstructured grid is the correct reproduction of block to block covariance, it is necessary to verify that the simulation algorithm follows the simple

kriging principle, which ensures that the covariance is reproduced. In the case of a lognormal random field, the normal score transform $y = \varphi^{-1}(z) = \ln z$ for $Z(x)$ is known analytically. An analytical verification for DSS 1 is easily done within these settings. The algorithm DSS 1, as originally presented in Soares (2001), proceeds as follows:

1) Simple kriging is performed at location x, conditional on surrounding values, thus yielding a kriging prediction $z^*(x)$, and a kriging variance, $\sigma_{sk}^2(x)$.
2) The kriging prediction is then transformed to a normal score $y^*(x) = ln\,(z^*(x\,))$.
3) A Gaussian random variable $\mathrm{Y} \sim \mathrm{G}\left(\mathrm{y}^*(\mathrm{x}), \frac{\sigma_{\mathrm{sk}}^2(\mathrm{x})}{\sigma^2}\right)$ is drawn.
4) The simulated value at point $z(x)$ is then set equal to e^Y.

Thus, $Z(x)$ is lognormal with logarithmic mean $ln\big(z^*(x\,)\big)$ and shape parameter $\frac{\sigma_{sk}^2(x)}{\sigma^2}$. The mean and variance of the local CDF at location x can be determined

$$EZ(x) = e^{ln(z^*(x\,))+\frac{\sigma_{sk}^2(x)}{2\sigma^2}} = z^*(x\,) \times e^{\frac{\sigma_{sk}^2(x)}{2\sigma^2}}, \tag{2.1}$$

$$\begin{aligned} Var\,Z(x) &= \left(e^{\frac{\sigma_{sk}^2(x)}{\sigma^2}} - 1\right) e^{2ln(z^*(x\,))+\frac{\sigma_{sk}^2(x)}{\sigma^2}} \\ &= z^*(x\,)^2 \times e^{\frac{\sigma_{sk}^2(x)}{\sigma^2}} \left(e^{\frac{\sigma_{sk}^2(x)}{\sigma^2}} - 1\right). \end{aligned} \tag{2.2}$$

Obviously, both mean and variance $(EZ(x), Var\,Z(x))$ are significantly different from the simple kriging mean and variance $(z^*(x\,), \sigma_{sk}^2(x))$, so the conditions of the simple kriging principle are not satisfied and the reproduction of the covariance is not guaranteed. Moreover, the variance is proportional to the squared kriging prediction at the given point, which is explained by the proportional effect inherent to a lognormal variable (Chilès & Delfiner 2012). Summarizing this results, the DSS 1 algorithm does not respect the simple kriging principle (Journel 1993) on which it is based.

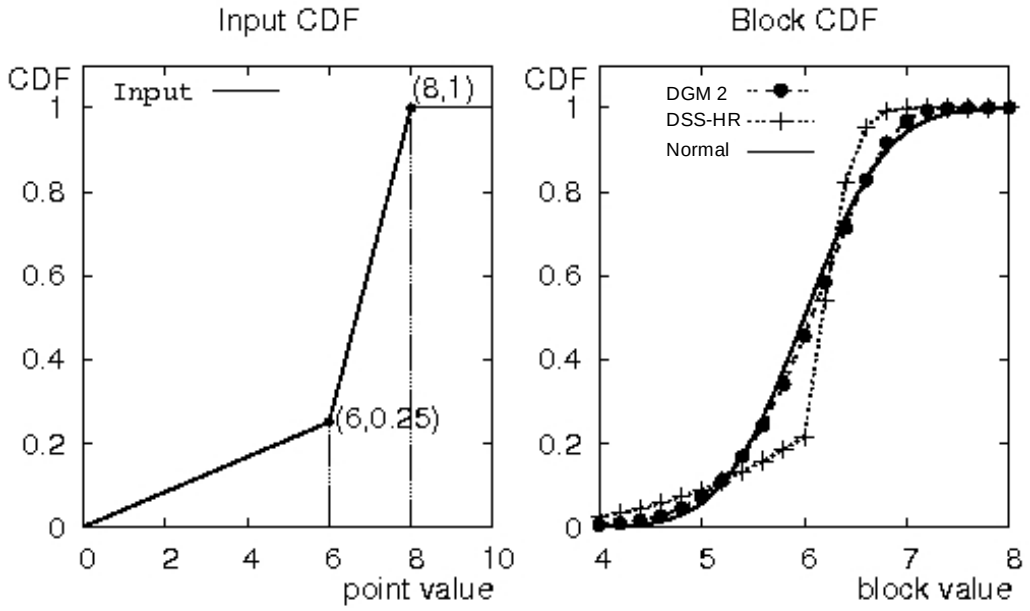
In contrast to DSS 1, the simulation algorithm DSS-HR (Oz et al. 2003), respects by construction the simple kriging principle, which guarantees the block-to-block covariance reproduction. As in the case of DSS 1, DSS-HR assumes that the local CDFs of $Z(x)$ depend only on the simple kriging mean and variance at a specific location. The algorithm consists in building a table of conditional CDF for $Z(x)$ in a constructive manner presented in (Oz et al. 2003), indexing this family of distributions by their mean and variance and using the distribution with correct mean and variance at every step of the sequential simulation. By construction, the indexed table of local distributions of $Z(x)$ is congruent with a table of Gaussian distributions.

When transferring the data from point to block support, it is expected that the mean value of the block is determined by the mean value of the point-support random field and the variance is determined by the point-support covariance and the block geometry (in the absence of conditioning data). DSS-HR is proven to preserve the mean of the simulated property and to reproduce the correct block-to-block covariance, despite the fact that the constructive manner of generating the set of conditional CDFs has certain theoretical drawbacks. DSS-HR implies marginal distributions for grid blocks which are not necessarily compatible with a random field distributed on point support. Thus, for any spatial random field $Z(x)$ with a covariance function $C(x,x')$ and with a finite integral range $A = \int_{R^3} C(x,x')\,dxdx'$ due to the central limit theorem (CLT) the distribution of the average $Z(v)$ of the block v tends to a Gaussian distribution with mean equal to the global mean, and variance $\frac{1}{|v|^2}\int_v \int_v C(x,x')dxdx'$ which decreases as the block size becomes larger. This property of the average value is often expected by the practitioners; the general proof of this statement goes beyond the scope of the present work. The compatibility with the CLT for DSS-HR does not follow straightforward from the simulation algorithm. In order to test the compatibility of DSS-HR with the CLT, one should investigate the dependence of the marginal distribution of a block depending on the size of this block. Algorithm DSS-HR implicitly assumes by construction that the unconditional marginal distribution of every block is contained in the pre-computed CDF table along with conditional distributions. It is also assumed that this marginal distribution is determined solely by the global mean and the variance of the block and this fact is used in the first step of the sequential simulation procedure.

When performing multiple unconditional simulations of a reservoir model with a given CDF table, one can expect that the marginal distribution of a given block coincides with the distribution from the CDF table which corresponds to the global mean and unconditional variance of that block only if the table of conditional distributions defines a valid multivariate distribution function. For the test it is assumed that the corresponding distribution from the table is observed. In practice, the observed distribution depends on implementation details, particularly regarding the neighborhood used for the computation of the kriging predictor and the kriging variance. The marginal distribution predicted by DSS-HR for a given block is now investigated. It should be noted that, when simulating a single block, the conditional CDF table always defines a valid multivariate distribution function. In this particular case, the assumption above is valid.

Consider a simulation of a RV with a point-support marginal distribution defined by a CDF at Figure 2-1(a). This function is selected for the test, since it has a marked difference in behavior in the neighborhood of (6, 0.25), the abscissa of which corresponds to the mean of the RV defined by this CDF. The mean and variance of the CDF at Figure 2-1 (a) are equal to 6 and 4 respectively. The distribution under concern enables to illustrate how the properties of the input CDF

are preserved by DSS-HR for all the generated distributions. It should be noted that the covariance model is not important for this test, since the block marginal distributions provided by DSS-HR and DGM can be parameterized if the block mean and variance are known. Figure 2-1(b) illustrates the CDF for a block with the variance 0.4 implied by DSS-HR for the point-support distribution from Figure 2-1(a) compared to CDF implied by DGM and to the normal distribution.



(a) (b)

Figure 2-1. Marginal distributions for (a) point support and (b) selected block support for DSS-HR, DGM 2 (see Chapter 3) and Gaussian distribution.

Figure 2-1(b) shows that even for a variance reduction factor 10 (ratio of the point-support variance to the block-support variance) the block marginal distribution proposed by DSS-HR preserves the shape of the input distribution in the neighborhood of (6, 0.25) and does not converge to the normal distribution. This behavior can be explained by the constructive manner of generating the table of CDFs. This effect is less noticeable for smooth CDF functions.

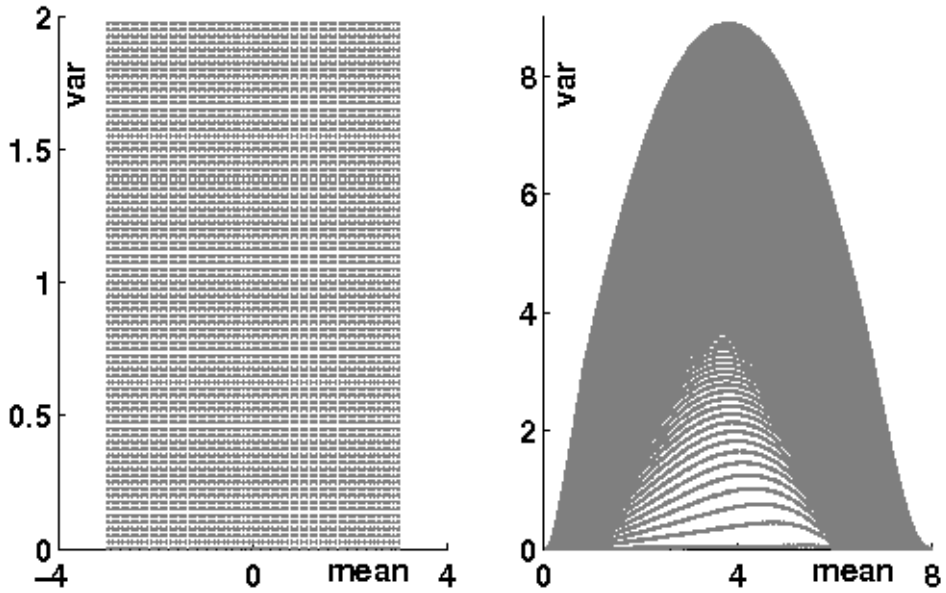


Figure 2-2. Scatter plot of pairs (mean, variance) for (a) Gaussian distributions and (b) corresponding local distributions generated by DSS-HR algorithm

A table of conditional distributions was generated for this test according to the DSS-HR algorithm. Figure 2-2 illustrates the link between the space of the Gaussian distributions and the space of the distributions constructed for the test. The example provided on Figure 2-1(a-b) shows that the applicability of DSS-HR should be tested in every particular problem and that it is unlikely that the proposed constructive algorithm could be used as a change of support model in the general case, especially for random fields with a non smooth CDF. In addition to this, DSS-HR is limited by the sequential simulation principle and suffers from the weakness of all the sequential simulation algorithms – necessity of using a limited neighborhood in the simulation. As it is demonstrated in Chapter 3, using local neighborhoods in the sequential simulation leads to underestimation of the theoretical covariance between the blocks.

Chapter 3: Discrete Gaussian model for Unstructured Grids

Résumé

Une nouvelle méthode de simulation sur les maillages non-structurés est présentée – la simulation avec le modèle Gaussien discret. Deux versions théoriques de ce modèle sont décrites – DGM 1 et DGM 2, la différence entre ces deux versions étant démontrée de façon analytique et avec un exemple synthétique de gisement. Une solution au problème de conditionnement des simulations est proposée, les aspects avancés des simulations géostatistiques sont discutés – à savoir le traitement des variables non-additives et des variables co-régionalisées.

3.1 Introduction

The Discrete Gaussian Model was originally proposed by Matheron (1976a) as a solution for approximate computation of transfer functions (conditional laws of blocks within a panel). The problem considered by Matheron (1976a) was of practical importance for the mining industry – each block within a panel is dispatched either to waste or to the mill. In order to predict the future production, It is important to assess the number and the average grade of the blocks within a panel whose grade is higher than the selected cut-off grade value. Emery (2007) investigates the properties of the DGM along with other change of support models (Hermitian, Laguerrian) and provides a simplified method for deriving the change of support coefficient and covariance between the transformed variables.

The application of DGM to geostatistical simulations is due to (Emery 2009; Emery & Ortiz 2011). In their papers authors apply DGM in the form stated in (Emery 2007) to the problem of geostatistical simulations on regular grids. As demonstrated by authors, DGM enables reducing the problem of geostatistical simulation of a continuous parameter on a regular grid with change of support effect to a problem of simulating a multivariate Gaussian random field and applying appropriate transformation functions. The main difference in application of DGM in (Emery 2009) and (Emery & Ortiz 2011) consists in using either turning bands or spectral simulation for generating the multivariate Gaussian random vectors. In this thesis we separate the theoretical model for geostatistical simulations on unstructured grids from the implementation of the simulation algorithm. The theoretical models developed in this thesis rely on the multivariate Gaussian random fields and thus the simulation algorithm requires simulating realizations of multivariate Gaussian random vectors. As it is demonstrated further in this chapter, various methods can be used as subroutines for simulating multivariate Gaussian random vectors which leads to different accuracy for the reproduction of the desired statistics.

A generalization of DGM for samples of different sizes was given in Brown et al. (2008), which presents a theoretical framework for incorporating multiple scale samples data on a regular grid. In Brown et al. (2008), an approach based on graphical models representation for grid blocks and samples and on the DGM variant from Emery (2007) is proposed. An extended version of the conditional independence assumption is then used to derive the covariance matrix between the block-transformed variables. This approach is deemed to be applicable in the mining industry, whereas this thesis proposes an approach for incorporating point-support distribution and spatial variation information on geological models with varying block size which is a common problem in petroleum reservoir engineering.

This thesis presents a generalization for unstructured grids of the classical DGM in the form given in (Chilès & Delfiner 2012). The random field $Z(x) = \varphi(Y(x))$ with covariance function $C(x, x')$ is considered to be an anamorphosis (i.e., a strictly monotonic transform) of a Gaussian random field $Y(x)$ with covariance function $\rho(x, x')$. Let $\{\varphi_i, i = 0 \dots \infty\}$ denote the coefficients of decomposition of $\varphi(y)$ in the basis of normalized Hermite polynomials $\{\chi_i(y), i = 0 \dots \infty\}$

$$\varphi(y) = \sum_{i=0}^{\infty} \varphi_i \chi_i(y), \tag{3.1}$$

The covariance functions $C(x, x')$ and $\rho(x, x')$ are related through Eqn.(3.2), see (Chilès & Delfiner 2012)

$$C(x, x') = \sum_{i=1}^{\infty} \varphi_i^2 \rho(x, x')^i \tag{3.2}$$

A subscript $\underline{x}$ will further denote a uniformly distributed randomized location of a point in the block v containing this point. For $\underline{x}$ belonging to a specific block v_p, the random variable $Y(\underline{x})$ is denoted Y_p.

The key assumption of the generalized DGM is that the average values of the blocks $\{Z(v_p), p = 1 \dots N_b\}$ are block-dependent transforms of random variables with standard Gaussian distribution

$$Z(v_p) = \varphi_{v_p}(Y_{v_p}), p = 1 \dots N. \tag{3.3}$$

A positive real number r_p called the block change of support coefficient is associated with each block v_p. By definition r_p is the correlation coefficient between Y_{v_p} and Y_p

$$r_p = cov(Y_{v_p}, Y_p). \tag{3.4}$$

The success of applying the DGM to the problem of geostatistical simulations on unstructured grids is due to the high quality of reproduction of the true marginal

distributions of the block values $Z(v) = \frac{1}{|v|}\int_v Z(x)dx$. As demonstrated analytically by Matheron (1985), DGM provides a second order accurate approximation of the density of the average value $Z(v)$ in the case of a multigaussian diffusion-type random fields, when the support v is constant throughout the domain.

The advantage of DGM relative to DSS algorithms is the compatibility with the central limit theorem (CLT). Indeed, when the change of support coefficient r_v for a block v is small, the block transformation function $\varphi_v(y)$ can be written in the following form (Chilès & Delfiner 2012)

$$\varphi_v(y) = \varphi_0 + \varphi_1 r_v y + \Delta(r_v), \tag{3.5}$$

where $\Delta(r_v)$ is small relative to r_v. And thus for small values of r_v

$$Z(v) \approx \varphi_0 + \varphi_1 r_v Y_v. \tag{3.6}$$

We start presentation of two generalizations (DGM 1 and DGM 2) of DGM from considering unconditional simulations. The additional assumptions required for conditioning will be considered after.

3.2 Discrete Gaussian model 1

In order to derive the theoretical model for the unconditional simulation $\{Z(v_p), p = 1 \dots N_b\}$ of a random field $Z(x)$ with marginal distribution $F(z)$ and covariance function $C(x, x')$ on an unstructured grid $\{v_p, p = 1 \dots N_b\}$, assumption about existence of underlying Gaussian random vector $\{Y_{v_p}, p = 1 \dots N_b\}$ is done as on Eqn. (3.3). The first generalized Discrete Gaussian Model (DGM 1) requires two main hypotheses, which are sufficient to produce unconditional simulations on unstructured grids that respect marginal and bivariate distributions:

i. The vector $(Y_{v_1}, \dots Y_{v_{N_b}})$ is stationary multivariate Gaussian;
ii. For every block v_p the joint distribution of Y_{v_p} and the value at the random point within the block Y_p is bivariate Gaussian with correlation coefficient r_p.

It is shown below, that the above assumptions are sufficient to convert the initial problem to a problem of generating unconditional Gaussian fields and deriving all the unknown parameters. Using Cartier's relation (Chilès & Delfiner 2012) p. 441

$$E[Z(\underline{x})|Z(v)] = Z(v) \tag{3.7}$$

assumption (ii) enables the derivation of the block-specific transform functions $\varphi_{v_p}(y)$ for every block v_p

$$\varphi_{v_p}(y) = \sum_{i=0}^{\infty} \varphi_i r_{v_p}^i \chi_i(y), \tag{3.8}$$

where $\{\chi_i(y), i = 0 \dots \infty\}$ is the basis of normalized Hermite polynomials. This decomposition of $\varphi_{v_p}(y)$ in the Hilbertian basis allows deriving the change of support coefficients for blocks. Indeed, since $\{\chi_i(y), i = 0 \dots \infty\}$ are orthogonal with respect to the scalar product induced by the density of the normal distribution $g(y)$, one can derive it using the point-support covariance $C(x, x')$ of $Z(x)$ and properties of isofactorial models (Chilès & Delfiner 2012; Rivoirard 1994)

$$Var\left(Z_{v_p}\right) = \frac{1}{|v|^2} \int_v \int_v C(x, x') dx dx' = \sum_{i=0}^{\infty} \varphi_i^2 {r_p}^{2i}, \tag{3.9}$$

which leads to a polynomial equation on r_p.

Assumption (i) permits to derive the covariance matrix for the random vector $(Y_{v_1}, \dots Y_{v_N})$. Indeed, the covariance R_{pq} between any two variables Y_{v_p} and Y_{v_q} can be determined through the following identity in the same manner as in Eqn. (3.9)

$$cov\left(Z_{v_p}, Z_{v_q}\right) = \frac{1}{|v_p||v_q|} \int_{v_p} \int_{v_q} C(x, x') dx dx' = \sum_{i=1}^{\infty} \varphi_i^2 {r_p}^i r_q^i R_{pq}^i. \tag{3.10}$$

The equations above enable the determination of the change of support coefficients $\{r_p, p = 1 \dots N_b\}$ and of the correlation coefficients $\{R_{pq}, p = 1 \dots N_b, q = 1 \dots N_b\}$. This converts the problem of generating a non-stationary random vector $\{Z(v_i), i = 1 \dots N_b\}$ to a classical problem of generating a multivariate Gaussian random vector $\{Y_{v_p}, p = 1..N_b\}$ with a given covariance matrix which can be solved by classical methods for simulating multivariate Gaussian random vectors (e.g. SGS). Application of block-specific transformation functions $\varphi_{v_p}(y)$ provides a realization of $\{Z(v_p), p = 1 \dots N_b\}$.

The accuracy of reproduction of the correct marginal distribution was demonstrated analytically for small change of support (Matheron 1985). For practical applications, results of numerous Monte Carlo simulations are available (Chilès 2014; Chilès & Delfiner 2012). The advantage of DGM 1 is the correct reproduction of the covariance between different blocks. Indeed, using the block-specific transformation functions and covariance between the components of the Gaussian vector derived above leads to the theoretically correct block to block covariance defined by the covariance function $C(x, x')$

$$cov\left(Z_{v_p}, Z_{v_q}\right) = \frac{1}{|v_p||v_q|}\int_{v_p}\int_{v_q} C(x, x')dxdx', \tag{3.11}$$

by the construction method of covariance table $\{R_{pq}, p = 1 \dots N_b, q = 1 \dots N_b\}$.

3.3 Discrete Gaussian model 2

This model is a generalization of another version of the DGM presented in (Emery 2007; Rivoirard 1994). At the cost of a third, more restrictive, assumption which will be detailed below, it provides a simpler approach for computing the change of support coefficients and covariance relations as compared to DGM 1. Since adding a more restrictive assumption implies that the model is less likely to fit the data, the price to pay is in some cases less accuracy in reproducing the histogram of the simulated property (Chilès 2014; Chilès & Delfiner 2012). There are also some theoretical consistency drawbacks that will be discussed. The additional assumption characterizing DGM 2 is:

iii. for any block v and two independent randomized locations $\underline{x}$ and $\underline{x}'$ within v, the joint distribution of $Y(\underline{x})$ and $Y(\underline{x}')$ is bivariate Gaussian.

One can derive from assumption (iii) the following relation for Y_v and $Y(v) = \frac{1}{|v|}\int_v Y(x)dx$ for every block (Chilès & Delfiner 2012; Emery 2007)

$$Y(v) = r_v Y_v, \tag{3.12}$$

which provides a simple formula for computing the change-of-support coefficient r_v for every block v (Emery 2007)

$$Var\big(Y(v)\big) = \frac{1}{|v||v|}\int_v\int_v \rho(x,x')dxdx' = r_v^2. \tag{3.13}$$

In an analogous manner, for any blocks v_p and v_q, the covariance R_{pq} between Y_{v_p} and Y_{v_q} is

$$R_{pq} = \frac{1}{r_p r_q} cov\,\Big(Y\big(v_p\big), Y\big(v_q\big)\Big) = \frac{1}{r_p r_q}\frac{1}{|v_p||v_q|}\int_{v_p}\int_{v_q}\rho(x,x')dxdx'. \tag{3.14}$$

Using covariance matrix $\{R_{pq}, p = 1 \dots N_b, q = 1 \dots N_b\}$ an unconditional realization of vector $(Y_{v_1}, \dots Y_{v_{N_b}})$ can be generated.

The theoretical cost of the additional assumption of DGM 2 is relatively high. It is not obvious, whether there exists a random field $Y(x)$ for which the joint distribution of the values at two random locations $\Big(Y\big(\underline{x}\big), Y\big(\underline{x}'\big)\Big)$ is bivariate Gaussian. Even in the case of a multivariate Gaussian $Y(x)$, the joint distribution of the vector $\Big(Y\big(\underline{x}\big), Y\big(\underline{x}'\big)\Big)$ is a mixture of bivariate Gaussian distributions with different correlation coefficients, which is known to be non-Gaussian, but Hermitian bivariate distribution. Indeed, suppose $\underline{x}$ and $\underline{x}'$ belong to the same block v. Due to the uniform distribution and to the independence of the randomized locations, it follows

$$P\left(Y\left(\underline{x}\right) \le y_1, Y\left(\underline{x}'\right) \le y_2\right)$$
$$= \frac{1}{|v|^2}\int_v \int_v P(Y(x) \le y_1, Y(x') \le y_2)dxdx' \quad (3.15)$$
$$= \int G_{\underline{\rho}}(y_1, y_2)\omega(d\underline{\rho}),$$

where $G_{\underline{\rho}}$denotes the bivariate standard Gaussian distribution with correlation coefficient $\underline{\rho}$ and $\omega(d\underline{\rho})$ denotes the density of $\underline{\rho}$. This formula proves that the joint distribution of $\left(Y\left(\underline{x}\right), Y\left(\underline{x}'\right)\right)$ is a mixture of standard Gaussian distributions with different correlation coefficients, which, according to (Matheron 1976b) is a Hermitian law with coefficients $\left\{T_i = \int \underline{\rho}^i \omega\left(d\underline{\rho}\right), i = 0 \ldots + \infty\right\}$. In DGM 2 these coefficients are approximated as $\left\{T_i \approx r_v^{2i}, i = 0 \ldots + \infty\right\}$ with the block change of support coefficient r_v.

Moreover, for every block v DGM 2 provides two different methods for computing the change of support coefficient r_v, using Eq. (3.9) and Eq. (3.13), which in general provide different results (Chilès 2014). Whether the simpler approach for determining the model parameters of DGM 2 leads to a correct reproduction of the correlation between blocks was theoretically investigated. The following result was obtained.

Proposition 1

The covariance between the block average values $Z(v_p)$ and $Z\left(v_q\right)$ computed with DGM 2 is biased relative to the theoretical covariance $\frac{1}{|v_p||v_q|}\int_{v_p}\int_{v_q} C(x, x')dxdx'$ between blocks v_p and v_q.

The proof of this proposition is provided in Appendix A. The bias is introduced due to the additional theoretical assumptions of DGM 2. In Emery (2009), the author presents an algorithm for geostatistical simulations on regular grids using DGM 2. The result provided in Appendix A evaluates the bias in covariance implied by this algorithm in the particular case of two identical blocks on a regular grid. The above-mentioned bias is illustrated in the following section. However, in the practical applications of petroleum industry, the range of the covariance function is usually significantly larger than the grid block size along different dimensions (in the area of main interest at least). Hence, the accuracy of DGM 2 can be considered as sufficient.

3.4 Model testing

In order to demonstrate the application of DGM 1 and DGM 2 to geostatistical simulations on unstructured grids, a two-dimensional Voronoï polygon grid of 20x20 km^2 is studied. The grid consists of 3,546 Voronoï polygon cells and includes 10 local grid refinement regions (LGR) in the areas of potential wells placement locations, the bounding box size for the smallest and the biggest blocks sizes are 36x42 and 1,035x1,052 m respectively. Figure 3-1 illustrates the studied grid highlighting the volumetric differences between the blocks.

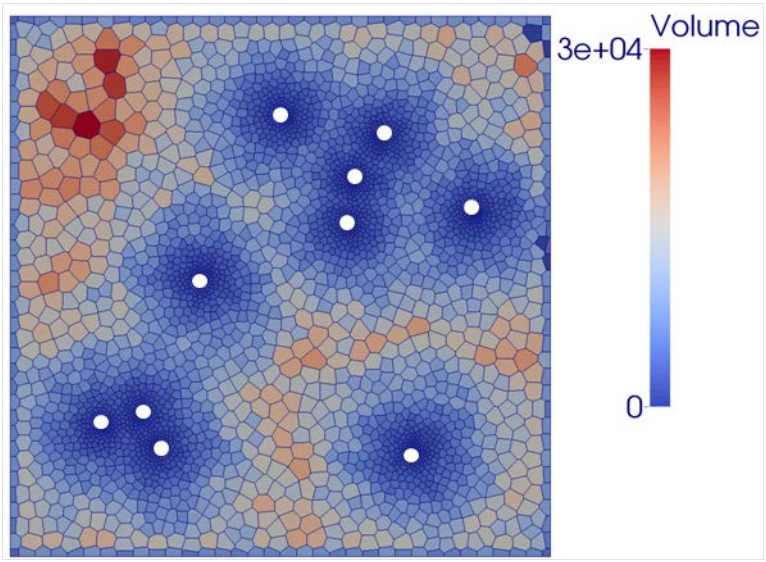


Figure 3-1. The volume distribution of the studied grid. 4 blocks near the right border were excluded from computations; their volume is set to 0. White points indicate the wells locations.

A simulation of a lognormal variable $Z(x) = \varphi(Y(x))$ with logarithmic parameters $(0, 1)$ on the block support is considered using DGM 1, DGM 2 as well as conventional simulation on a fine regular grid with upscaling approaches. The covariance $\rho(\bar{h})$ of the normal score transform variable $Y(x)$ is considered to be known. For this test the isotropic spherical covariance with a small range of 250 m was used in order to test the accuracy of DGM 1 and DGM 2 for a variety of ratios between covariance range and block size. The covariance $C(\bar{h})$ of $Z(x)$ is computed numerically through formula (3.2).

Realization both of a conventional simulation and of a simulation with a DGM are illustrated on Figure 3-2(a-b). SGS was used for simulating the multivariate Gaussian random vectors. The results for DGM 1 and DGM 2 are visually similar, only the result for DGM 2 is provided. Based on 50,000 unconditional simulations, the observed distributions of the block values as well as the descriptive statistics of the simulated values were analyzed. The results obtained demonstrate satisfactory accuracy in approximating the real distribution of the average block values with DGM

1 and DGM 2 predicted densities. For small blocks the densities provided by DGM 1 and DGM 2 are undistinguishable on the plot. For the largest block in the model the comparison is given on Figure 3-3.

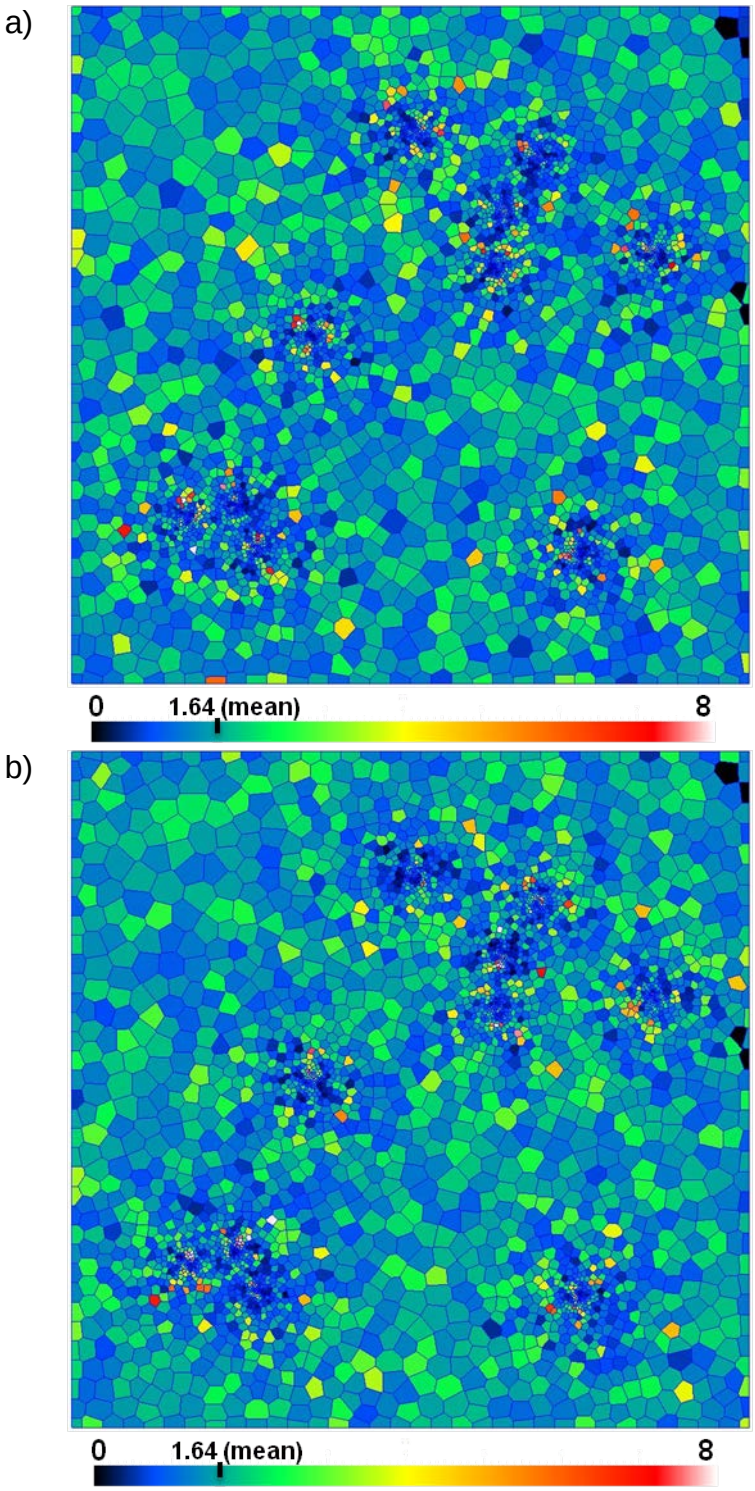


Figure 3-2. Simulation of a lognormal variable on a Voronoï polygon grid for spherical $\rho(\overline{h})$ with range 250m. a) using fine scale regular grid and upscaling. b) using DGM 2

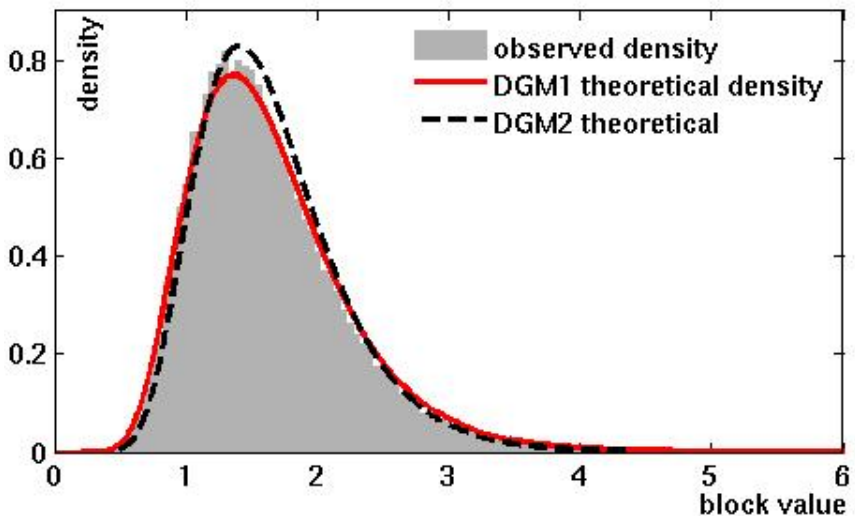


Figure 3-3. Observed density for the value of the largest block in the grid, compared to theoretical predictions of DGM 1 and DGM 2, based on 50,000 simulations

Investigating the descriptive statistics of the block values enables to highlight the practical difference between application of DGM 1 and DGM 2. The variance of every block in the model was estimated from the sample and compared to the theoretical value of the block variance. Figure 3-4 demonstrates the mismatch $\left[C(v,v) - \hat{C}(v,v)\right]$ of the estimated variance from the theoretical variance depending on the block volume.

As can be seen on Figure 3-4, the difference between the estimated and theoretical variance for the realizations produced by DGM 1 approach oscillates around 0, whereas DGM 2 approach demonstrates a bias in the variance which is derived analytically in Appendix A. It should be noted that the relative value of the bias does not exceed 5% of the a priori variance of $Z(x)$. When the covariance range exceeds significantly the block dimensions, which is often encountered in practical applications, the difference between DGM 1 and DGM 2 can be neglected. A realization of the previous test with a covariance function range of 8 km is given on Figure 3-5(a-b).

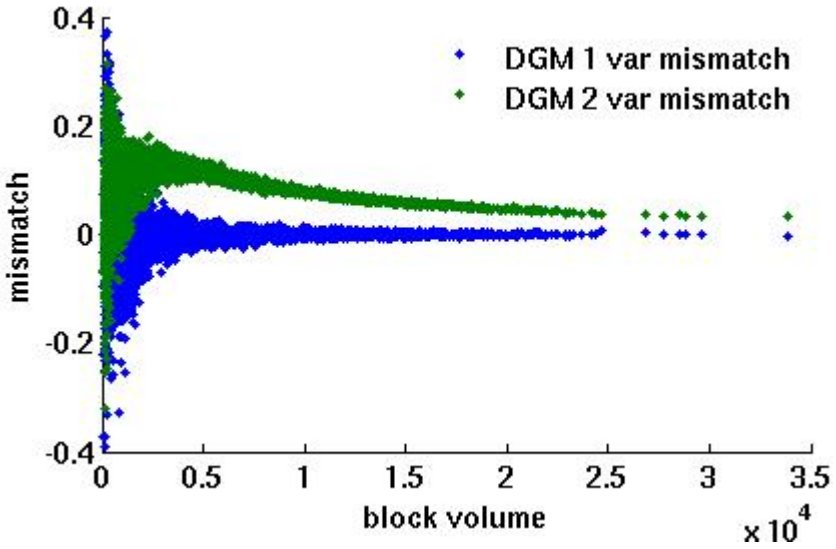
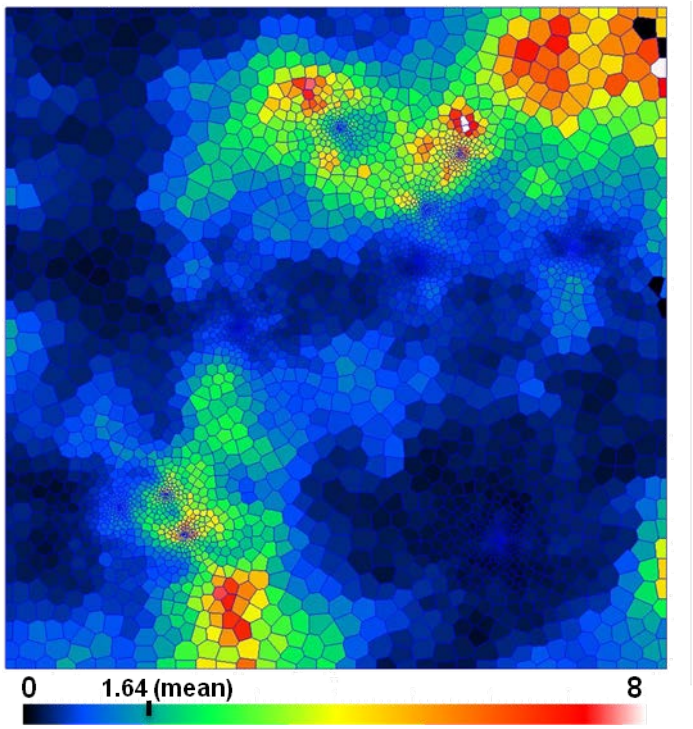


Figure 3-4. Mismatch of the estimated variance from the theoretical variance of the block defined by $C(\bar{h})$

a)



b)

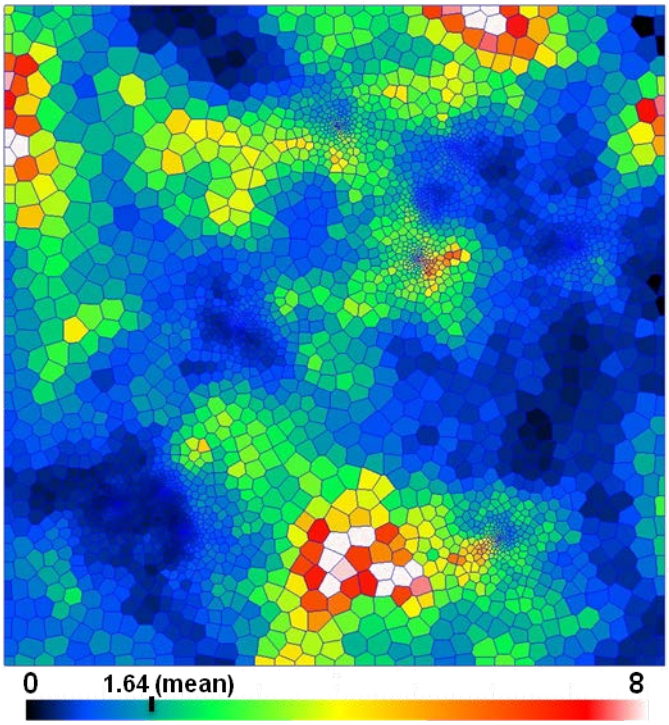


Figure 3-5. Simulation with a spherical covariance $\rho(\bar{h})$ with range 8 km using a) fine scale regular grid and upscaling b) DGM 2

Let us consider a more rigorous test for unconditional simulations on unstructured grids with DGM. Let us consider that the grid on Figure 3-1 is a 3D grid with the thickness of 40 meters (in order to introduce complexity into covariance computations), see Figure 3-6.

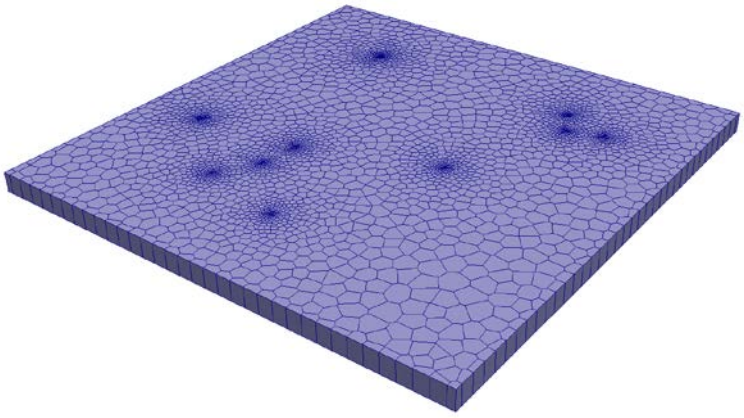


Figure 3-6. 3D grid with 10 local refinement zones. Vertical zoom factor 20.

The reproduction of all block to block theoretical covariances $C(v_p, v_q)$ can be tested. We use 3 different input covariance models $\rho(\bar{h})$ in 3D for the Gaussian variable $Y(x)$ which is a normal score transform of $Z(x)$:

1) Short range $\rho_1(\bar{h})$ - spherical covariance with ranges (250m, 250m, 10m);

2) Long range $\rho_2(\bar{h})$ - spherical covariance with ranges (5km, 5km, 0.2km);
3) Double structure $\rho_3(\bar{h})$ – 0.75 Sph (250m, 250m, 10m) + 0.25 Sph(5km, 5km, 0.2km).

Case 3 represents a mixture of cases 1 and 2. The normalized to unit covariance functions $\rho_1(h)$ and $\rho_3(h)$ as well as the corresponding $C_1(h)$ and $C_3(h)$ are depicted on Figure 3-7.

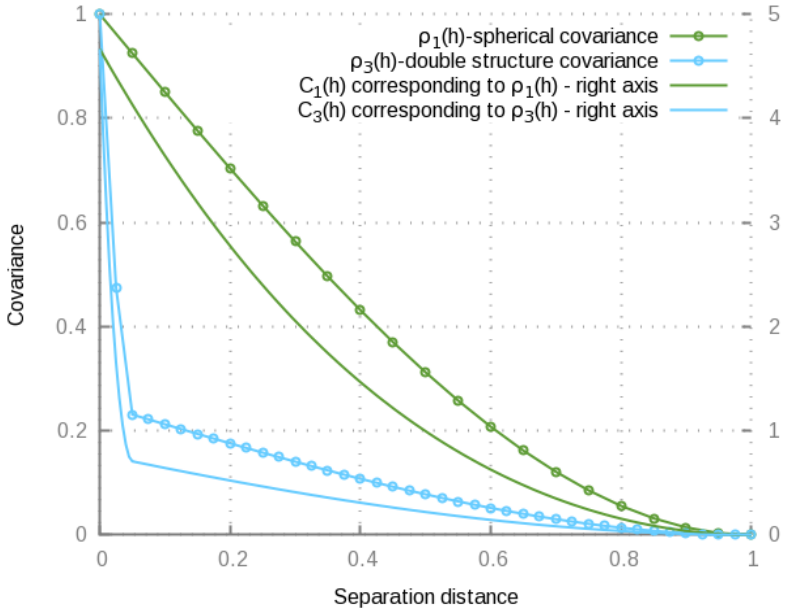


Figure 3-7. Normalized covariance functions $\rho_1(\bar{h})$, $\rho_3(\bar{h})$ (left axis) and $C_1(h)$,$C_3(h)$ (right axis) used for the test..

For each input covariance $\rho_i(\bar{h}), i = 1,2,3$ of $Y(x)$, the corresponding covariance $C_i(h)$ of $Z(x)$ is computed. Given $C_i(h)$ the full matrix of the block to block covariance $C(v_p, v_q), p = 1 \dots N_b, q = 1 \dots N_b$ is computed and stored (since the matrix is symmetric and sparse, only pertinent values are stored). Multiple simulations are produced with DGM 1 and the covariance between each pair of blocks is estimated. The Gaussian random vector $\{Y_p, p = 1 \dots N_b\}$ is produced with SGS, taking 50 closest neighbors for the small range covariance $C_1(x, x')$ and 200 neighbors for the other two cases. For case 1 – small range covariance 50000 simulations were produced, for cases 2 and 3, due to use of a significantly larger simulation neighborhood the number of produced realizations was reduced to 20000. Three different realizations produced with DGM 1 for small range, long range and double structure input covariance model are demonstrated on Figure 3-8.

The expected spatial behavior of the model is visually reproduced on Figure 3-8(a-c). Figure 3-8a,c demonstrate more spatial variation in refinement zones and less variation in the zones of large blocks. Spatial features of the expected size (5×5

km) are visible on Figure 3-8b,c. Figure 3-8c reproduces both the fine scale variations in the refinement zones and the large scale heterogeneities.

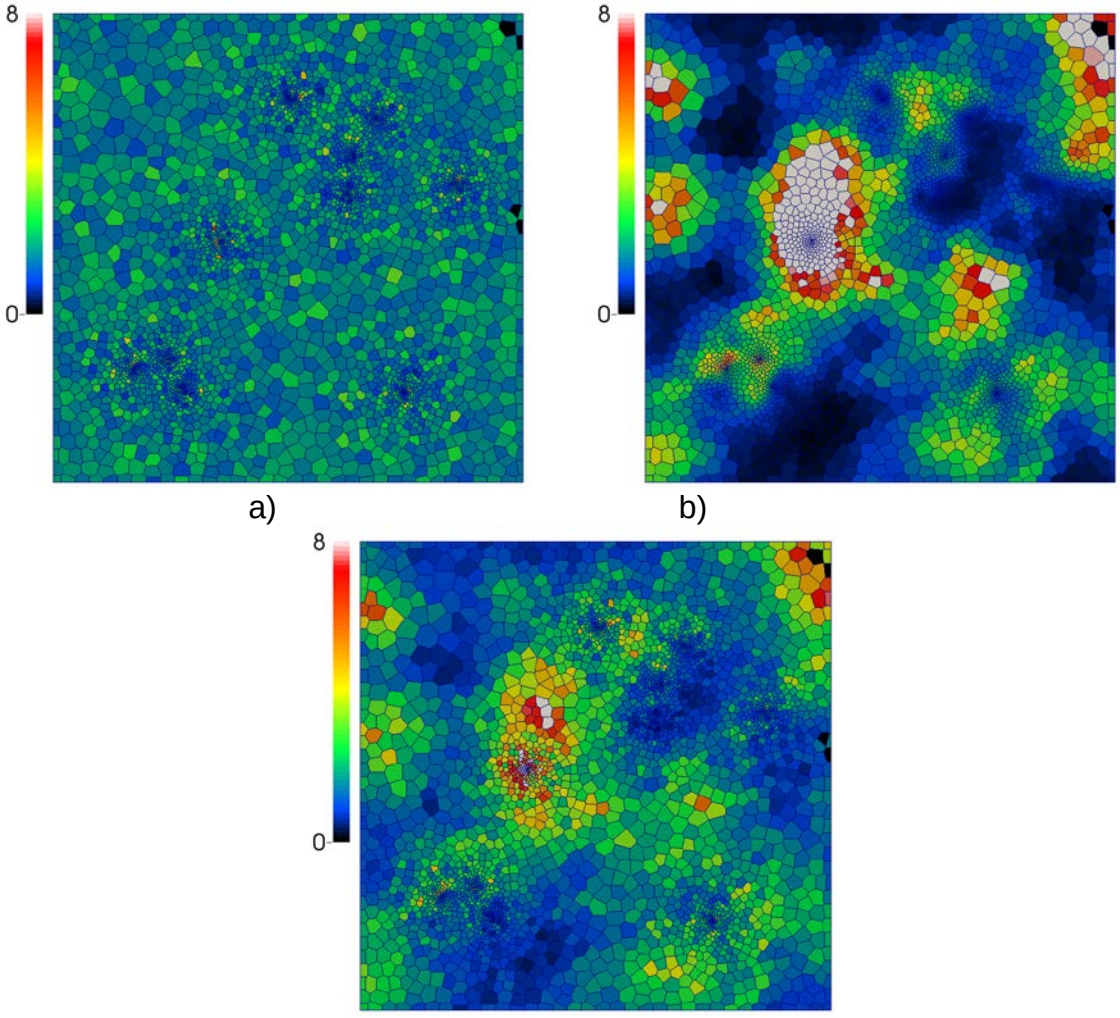


Figure 3-8. Simulations on an unstructured 3D grid for a) covariance with small range $C_1(h)$ b) covariance with long range $C_2(h)$ and c) double structure covariance $C_3(h)$

The scatter plots of the estimated covariance values versus the theoretical are demonstrated on Figure 3-9a-c (zero theoretical covariances are ignored). It is visible that the DGM 1 – based simulation method reproduces the block to block covariance for all 3 types of input covariance functions. For the purpose of this test, all block to block covariances were computed through approximating each block of the cell with 50 Sobol' quasi-random points (see Chapter 5). Figure 3-9b,c shows that the dispersion covariance of the estimates increases with the increase of the theoretical covariance value. This behavior is in line with the theoretical model used – the most correlated blocks on the grid are the smallest ones, for which the simulated values are more dispersed.

The different neighborhood sizes (20 for small range and 200 for other cases) for this test were chosen by a trial method. It was observed that for a small range

covariance $C_1(\bar{h})$ using 20 neighbors is sufficient in order to reproduce block to block covariances, whereas significantly larger neighborhood is required for the long range covariance functions. Figure 3-10 illustrates the scatter plot of the observed block variances versus theoretical block variances for $C_3(\bar{h})$. The results are obtained from 20,000 unconditional simulations. Figure 3-10a shows that when 20 neighbors are used for the double structure covariance, the block variance is systematically underestimated. Since by construction of the theoretical model DGM 1 respects the block to block covariance, the source of underestimation is the SGS procedure which was used to simulate multivariate Gaussian random vectors. This underestimation can be corrected with increasing the neighborhood size; indeed, Figure 3-10b shows that when 200 neighbors are used the underestimation diminishes. However, it remains noticeable, especially for large values of block variance.

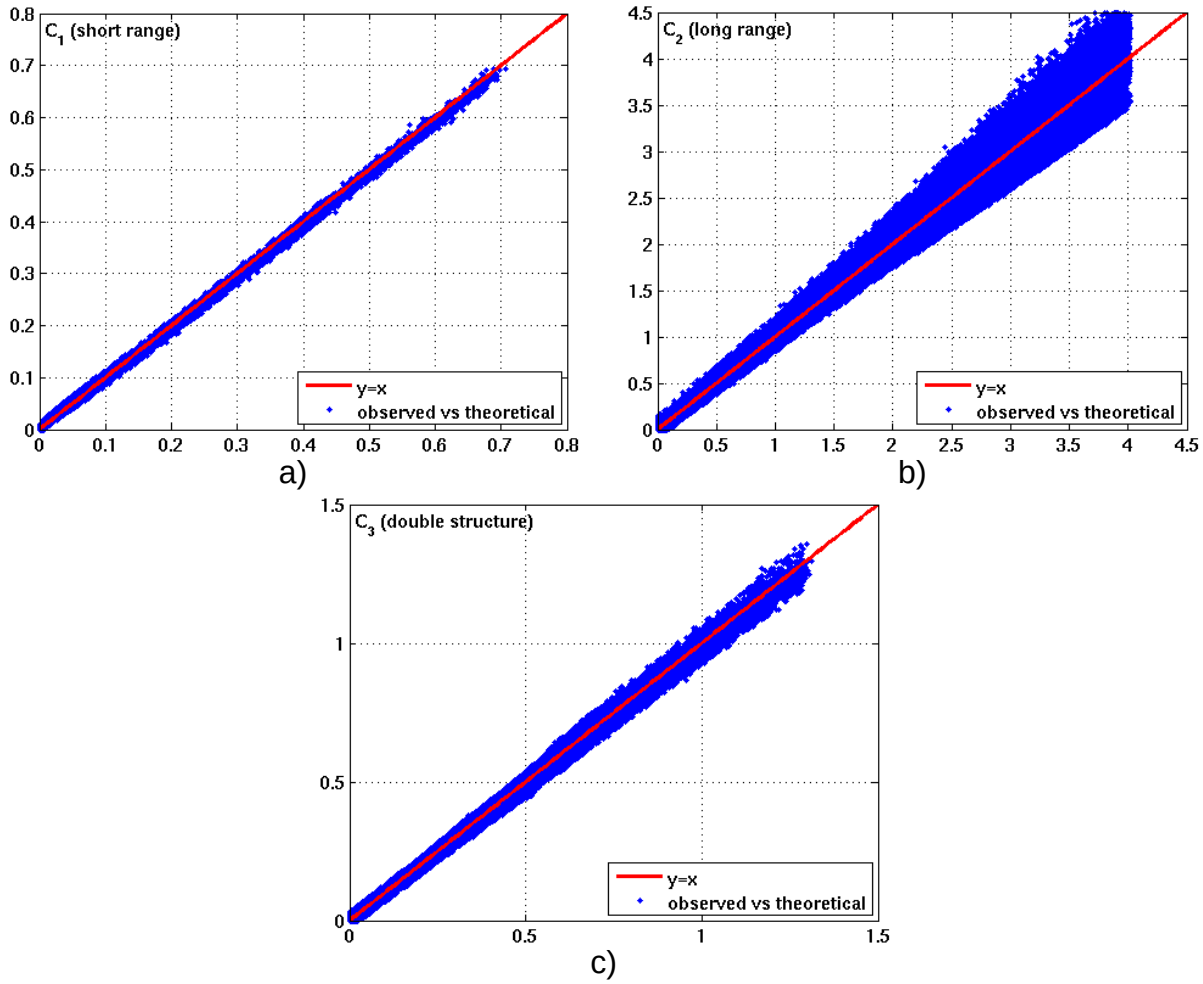


Figure 3-9. DGM 1. Scatter plots of observed covariance versus theoretical covariance for a) covariance with small range $C_1(\bar{h})$b) covariance with long range $C_2(\bar{h})$ and c) double structure covariance $C_3(\bar{h})$

This problem arises due to the fact that only a limited number of neighbors were used in SGS. Although the number of neighbors used (200) is relatively large for this type of problems (it is common to use less than 50 neighbors in practical

applications of petroleum industry, and sometimes only the cells which share a common side are used as neighbors), a systematic deviation from the theoretical values can be observed. The destructive effect for the model statistics due to the limited SGS neighborhood was noted in Emery (2004b) for the case of spherical variogram in 2D. As demonstrated in the same paper, increasing the neighborhood size improves the quality of the simulation. Except for special cases, such as simulating with exponential covariance function (Chilès & Delfiner 2012), there does not exist a theoretical relationship between the size of the neighborhood which should be used in SGS and the type of the simulated covariance model. The impact of the limited simulation neighborhood can be assessed as proposed by Emery and Peláez (2011).

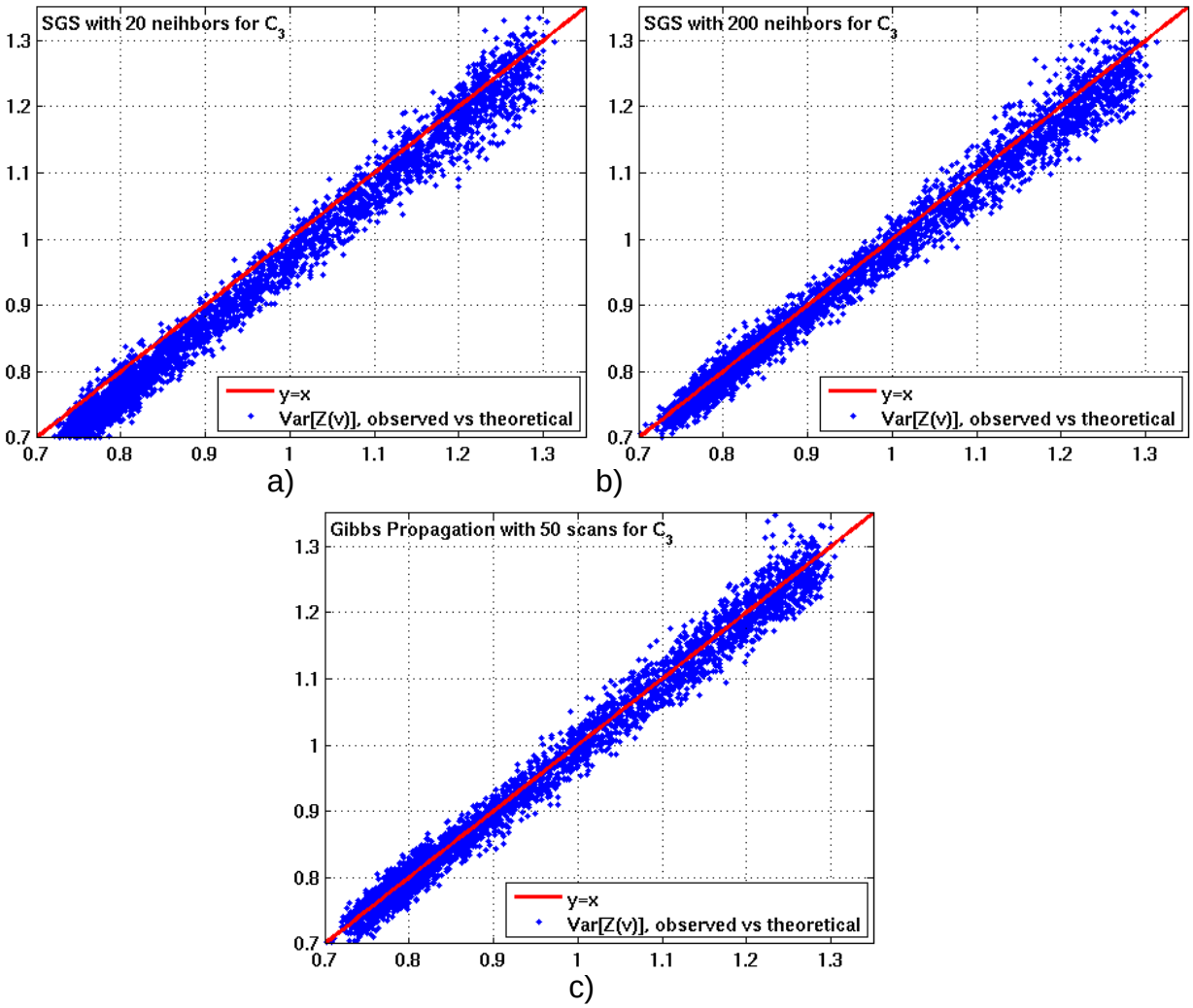


Figure 3-10. Scatter plots of sample variance versus theoretical for $C_3(\bar{h})$ for a) SGS with 20 neighbors b) SGS with 200 neighbors c) Gibbs Propagation algorithm with 50 scans.

This deviation from the theoretical values of block variance can be eliminated if a more reliable method for generating Gaussian random vector $\{Y_p, p = 1 \dots N_b\}$ is used. One of the alternatives of the SGS for simulating multivariate Gaussian random

vectors with a given correlation matrix is using iterative algorithms based on Markov chains, such as Gibbs Sampler (Geman & Geman 1984). A particularly useful implementation of a Gibbs Sampler was proposed by Lantuéjoul and Desassis (2012) – the Gibbs Propagation algorithm. This implementation of Gibbs Sampler does not require any matrix inversions and is very optimal in terms of memory consumption. The Gibbs Propagation algorithm performs a number M of scans, at each scan iterating through all N_b components of the simulated Gaussian random vector. For our tests we use $M = 50$, which gives a good quality of statistics reproduction. A discussion of the required number of scans can be found in (Lantuéjoul & Desassis 2012). As demonstrated by the authors, the number of scans M can be further significantly reduced (up to a factor 4) by grouping the components of the simulated vector in blocks and updating the resulting blocks instead of individual components. Use of the Gibbs Propagation algorithm enables to abandon the problem of the simulation neighborhood of the SGS algorithms, since all components of the simulated vector are taken into account. A simulation produced with Gibbs Propagation algorithm is presented on Figure 3-11. In this test both for the Gibbs Propagation algorithm and for the SGS the computation time required to generate a single realization is around 1 minute.

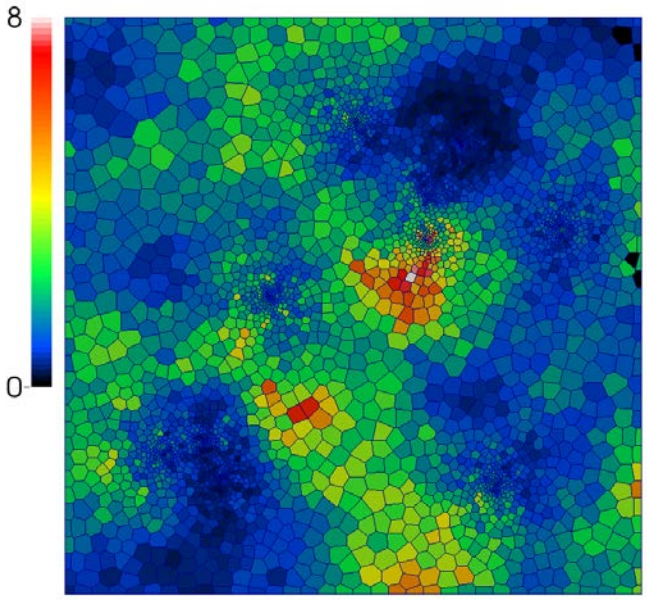


Figure 3-11. Simulation for double structure covariance $C_3(x, x')$ produced with $M = 50$ scans of Gibbs Propagation algorithm.

As it is seen on Figure 3-10c, using the Gibbs Propagation algorithm with 50 scans provides robust reproduction of the block to block covariance. Practical application of Gibbs Propagation algorithm for industrial size models ($N_b > 10^6$) is a challenging task. However, geostatistics is often applied in industries where the computational resources are abundant, as in mining, petroleum, weather forecasting and other industries. A decent programming implementation of the Gibbs

Propagation algorithm coupled with sufficient computational resources can lead to a robust and practical method for simulating random fields of large sizes. In addition, the fact that any suitable method can be applied for generating a multivariate Gaussian random vector when performing geostatistical simulations with DGM demonstrates the universality of the DGM-based simulation approach since it is not linked to the sequential simulation paradigm in contrast to DSS algorithms.

In order to demonstrate the reproduction of the marginal block distribution in the case of 3D simulations, a mini-model simulation for a single block is performed. We choose a medium – sized block from the grid with dimensions $[550, 530, 40]$m (see Figure 3-12) and perform mini-models simulations for this block for the case of the double structure covariance $C_3(\bar{h})$.

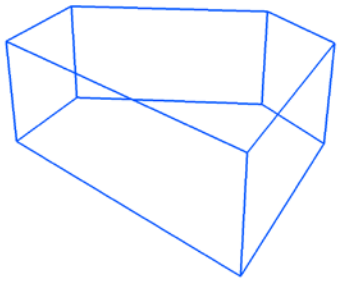
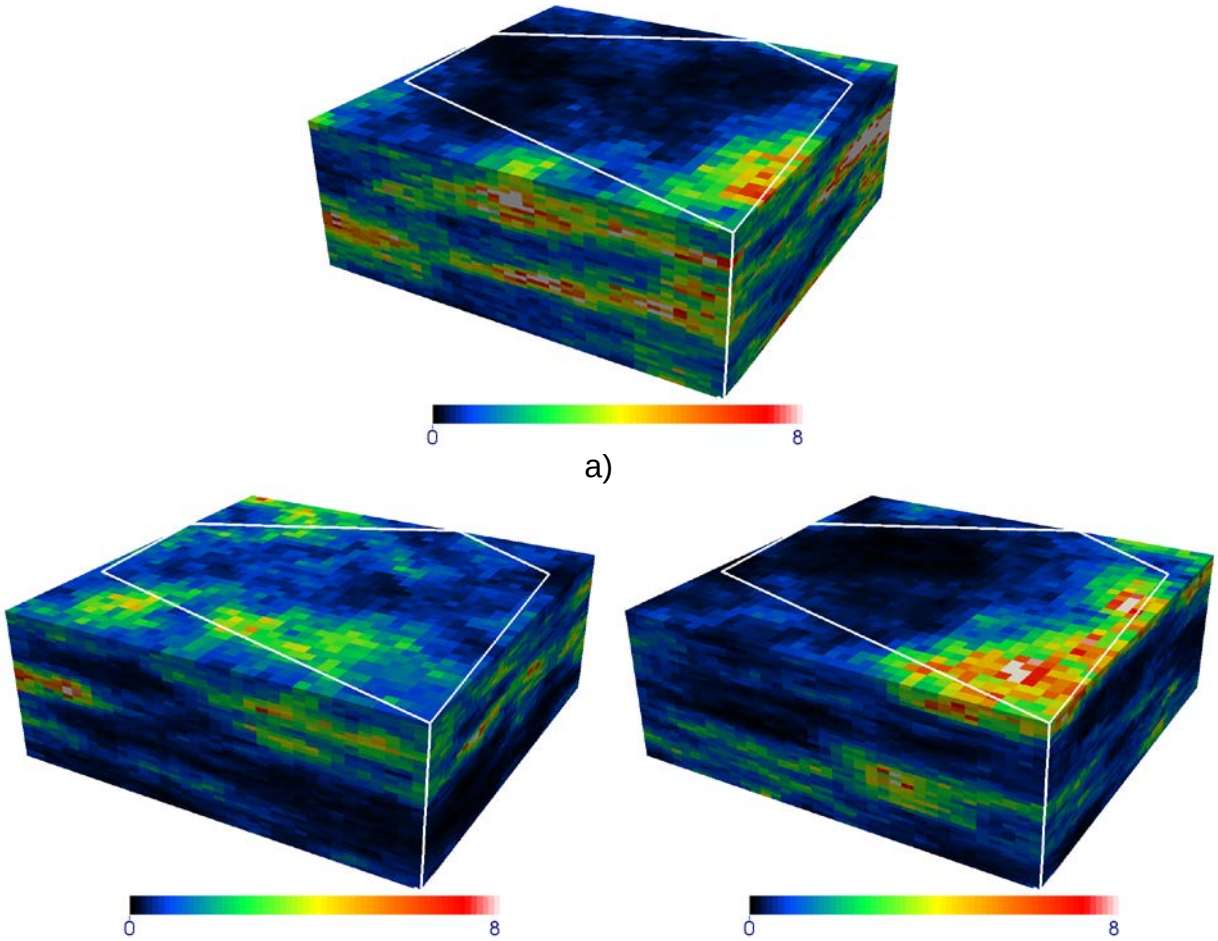


Figure 3-12. Extracted block for mini-model tests. Block dimensions $[550, 530, 40]$m.

The bounding box of the block is discretized with a regular grid with steps $15 \times 15 \times 1m$ and 10000 simulations with SGS are performed on this regular grid with input standard lognormal distribution and covariance function $C_3(\bar{h})$ (see Figure 3-13). The total number of blocks in the fine scale regular grid is 170*230*40 = 1,564,000 blocks. Each of 10000 fine scale simulations is upscaled and the histogram of the observed $Z(v)$ is computed. The theoretical block density is also computed with DGM 1 and DGM 2, where for the purpose of covariance computation the block is approximated with 50 Sobol' quasi-random points (see Chapter 5: 5). The difference between the DGM1 and DGM2 results is visually negligible, so only the results for DGM 1 are shown. The empirical distribution of $Z(v)$ produced by the mini-model simulations is compared with the distribution obtained in the previous test when simulating with DGM 1 on the full grid with covariance $C_3(\bar{h})$. For DGM 1 simulations the Gaussian random field is generated with SGS with 200 closest neighbors. In order to have the same number of simulated values in both approaches, only first 10,000 of previously simulated 20,000 DGM1 realizations are considered.



a) b) c)

Figure 3-13. Different mini-model realizations for the selected block. The discretization steps are $15 \times 15 \times 1m$. The white line indicates the contour of the Voronoï polygon block.

The observed empirical density of the $Z(v)$ produced with mini models and with DGM 1 are compared on Figure 3-14. A perfect reproduction of the block distribution with DGM is observed; both mini-model and DGM 1 honor the theoretically predicted density function (which is undistinguishable for DGM 1 and DGM 2).

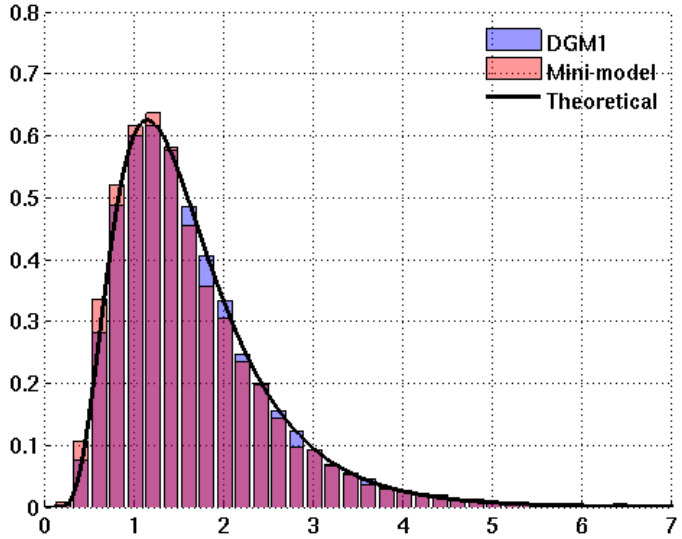


Figure 3-14. Observed empirical density for mini-model and DGM1 simulations. Black line indicates the theoretical (DGM1 and DGM 2) density function.

Closer analysis of results on Figure 3-14 reveals some fluctuations in the observed statistics. The mean and variance of $Z(v)$ produced with mini-models are equal 1.61 and 0.84 respectively. Mean and variance observed for DGM 1 simulations are 1.64 and 0.76 respectively, the change of support coefficient for the block is $r = 0.5$. The theoretical mean and variance when approximating this block with 50 Sobol' points is 1.64 and 0.79. In case of approximating with 1000 Sobol' points, these values are 1.64 and 0.74 respectively; $r = 0.49$. The observed small difference of the variance produced with mini-models is explained with the low quality of approximating the block through a regular mesh (see Chapter 5) and imperfectness of the SGS simulation algorithm.

3.5 Conditioning the simulations on unstructured grids

Although it is possible to develop a DGM-based theoretical model for conditional simulations, separating the problem into unconditional simulation followed by conditioning to data procedure enables to give simpler workflow presentation. Unconditional simulations generated with DGM1 and DGM2 can be conditioned with kriging. Conditioning with kriging is well known for the regular grids (Chilès & Delfiner 2012; de Fouquet 1994; Journel & Huijbregts 1978) and can be used for unstructured grids without significant modifications.

Formally, a geostatistical simulation on an unstructured grid $\{Z(v_p), p = 1 \dots N_b\}$ is just a random vector with N_b components which have predefined marginal distributions and covariance. In order to speak about conditioning of the simulation on point support, one should assume the existence of some point support RF $Z(x) = \varphi(Y(x))$, which corresponds to this random vector i.e. $Z(v_p) = \frac{1}{|v_p|}\int_{v_p} Z(x)dx\,, p = 1 \dots N_b$. In that case, conditioning of $\{Z(v_p), p = 1 \dots N_b\}$ on points $\{(d_i, z_i), i = 1 \dots N_{data}\}$ can be understood as conditioning of $Z(x)$ and re-computing $Z(v_p) = \frac{1}{|v_p|}\int_{v_p} Z(x)dx\,, p = 1 \dots N_b$ from the conditioned $Z(x)$. In practice, however, no re-computing is needed, since all operations can be performed directly on block support since the kriging predictions are averaged linearly over volume (Goovaerts 1997). Given an unconditional simulation $\{Z(v_p), p = 1 \dots N_b\}$ on blocks of an unstructured grid $\{v_p, p = 1 \dots N_b\}$ and the conditioning data $\{(d_i, z_i), i = 1 \dots N_{data}\}$, a conditional simulation can be generated. For that, following the conditioning by kriging approach, the values of the unconditional simulation at the data points $\{Z(d_i), i = 1 \dots N_{data}\}$ should be known. It is clear that given only the simulated average values of the blocks $\left\{Z(v_p) = \frac{1}{|v_p|}\int_{v_p} Z(x)dx\,, p = 1 \dots N_b\right\}$, one cannot exactly derive the values of $\{Z(d_i), i = 1 \dots N_{data}\}$, and approximation techniques should be used.

One way to obtain the values $\{Z(d_i), i = 1 \dots N_{data}\}$ from the simulated on block support unconditional random field is using the assumption (ii) of the DGM 1 and DGM 2. Consider a block v_p which contains one conditioning point (d_1, z_1). Let the corresponding normal score transform for this conditioning point be (d_1, y_1). Following assumption (ii), one can derive for a randomized point $Y(\underline{x}) = Y_p$ within block v_p:

$$Y_p = r_p Y_{v_p} + \sqrt{1 - r_p^2}\,\xi_p, \tag{3.16}$$

where ξ_p is a standard Gaussian RV with mean 0 and variance 1, since the joint distribution of Y_p and Y_{v_p} is bivariate Gaussian. The value Y_p can be then identified

with the value $Y(d_1)$ of the Gaussian RF $Y(x) = \varphi^{-1}(Z(x))$ at point d_1. In this approximation, the value at the fixed location d_1 is considered to be equivalent to the value at the randomized location $\underline{x}$ within block v_p (which is an approximation). The value $Z(d_1)$ can then be obtained: $Z(d_1) = \varphi\big(Y(d_1)\big)$.

Obtaining the values of the point-support random field through Eqn. (3.16) is an approximation, since the value $Y(d_1)$ at fixed location d_1 is approximated with the value at a randomized point. The quality of this approximation increases when the size of the block v_p containing the conditioning point (d_1, z_1) is small relative to the covariance range. For the applications of petroleum industry, the hard conditioning data are represented by well cores and core plugs and it is common to make a local grid refinement in the areas of wells. In that case, identifying the fixed and the randomized locations for conditioning data can give sufficient accuracy for obtaining a reservoir model conditioned to wells data. It should be noted that when a block v_p contains several samples, the variables ξ_p in (3.16) can be drawn independently for each sample. This corresponds to a conditional independence hypothesis: given the average value of the block Y_{v_p}, the sample values are independent. It should be noted that the conditional independence does not imply independence of the data – the samples remain correlated. The hypothesis of conditional independence is, certainly, a strong approximation and discussing its accuracy is out of the scope of this thesis. An extended discussion on the conditional independence can be found in Rue and Held (2005).

Let us consider that the value Y_p obtained through Eqn.(3.16) corresponds to a value of RV $Y(d_1)$. This value can be transformed from the Gaussian scale to the scale of the variable Z: $Z(d_1) = \varphi\big(Y(d_1)\big)$. The value $Z(d_1)$ can be seen as consistent with the unconditional simulation $\{Z(v_p), p = 1 \dots N_b\}$. The consistency in this context means that both $Z(d_1)$ and $\{Z(v_p), p = 1 \dots N_b\}$ originate from the same realization of a point-support RF $Z(x)$ in D, for which $Z(v_p) = \int_{v_p} Z(x)dx$.

Having the conditioning data $(d_i, z_i), i = i \dots N_{data}$, with consistent unconditional block simulation $\{Z(v_p), p = 1 \dots N_b\}$ and point-support values $\{Z(d_i), i = 1 \dots N_{data}\}$, one can use simple kriging to obtain a conditional simulation $\{T(v_p), p = 1 \dots N_b\}$ in the following manner:

$$T(v_p) = Z(v_p) + \Sigma_{i=1}^{N(v_p)} \lambda_i \big(z_i - Z(d_i)\big), \tag{3.17}$$

where $N(v_p)$ is the number of the closest data is points to the block v_p given by the neighborhood selection strategy, the weights $\{\lambda_i, i = 1 \dots N(v_p)\}$ are the solution of the block-kriging system:

$$\begin{pmatrix} C(d_1,d_1) & \cdots & C\left(d_1,d_{N(v_p)}\right) \\ \vdots & \ddots & \vdots \\ C\left(d_1,d_{N(v_p)}\right) & \cdots & C\left(d_{N(v_p)},d_{N(v_p)}\right) \end{pmatrix}\begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_{N(v_p)} \end{pmatrix} = \begin{pmatrix} C(v_p,d_1) \\ \vdots \\ C\left(v_p,d_{N(v_p)}\right) \end{pmatrix} \quad (3.18)$$

This approach for conditioning guarantees that the covariance of the conditional simulation $\{T(v_p), p = 1 \dots N_b\}$ coincides with the covariance of the unconditional RF, although, the spatial distribution is not preserved in the general case (de Fouquet 1994). The marginal distribution of the blocks is corrected only locally within the range of covariance from the conditioning data. It should be noted that this conditioning method does not require any additional assumptions to those of the DGM 1, however, it can rarely be applied in practice since conditioning through (3.17) can lead to values of $Z(v_p)$ which are not compatible with the marginal distribution of $Z(x)$. For instance, negative values can be obtained for a simulated positive parameter. A more practical solution is to perform the conditioning to data for random vector $\{Y_{v_p}, p = 1 \dots N_b\}$ instead of $\{Z(v_p), p = 1 \dots N_b\}$. In this case the resulting conditioned random field is always consistent with the marginal distribution of $Z(x)$. This approach is detailed below.

For conditioning purpose only, one can assume that the assumption (iii) of DGM 2 holds for $Y(x)$. In this case, it is possible to use Eqn. (3.12) in order to link RV Y_{v_p} with the volumetric averages of the underlying Gaussian random field: $Y(v_p) = r_p Y_{v_p}$. Note that using assumption (iii) only at conditioning step is different from using it from the beginning, since the conditioning procedure modifies the unconditional simulation only locally.

Eqn. (3.16) enables us obtaining a set of Gaussian point-support values $\{Y(d_i), i = 1 \dots N_{data}\}$ which can be seen as consistent with the set of Gaussian averages $\left\{Y(v_p) = \frac{1}{|v_p|}\int_{v_p} Y(x)dx\,, p = 1 \dots N_b\right\}$ that correspond to the unconditional simulation $\{Z(v_p), p = 1 \dots N_b\}$. The consistency, as mentioned previously, indicates that both the Gaussian averages and the point-support values can be seen as originating from the same unconditional point-support RF $Y(x)$. In this case the conditioning by kriging can be done directly in the Gaussian scale for the variable $Y(x)$ using $(d_i, y_i) = \left(d_i, \varphi^{-1}(z_i)\right), i = i \dots N_{data}$ - the normal score transform of the given hard data:

$$S(v_p) = Y(v_p) + \Sigma_{i=1}^{N(v_p)} \lambda_i \left(y_i - Y(d_i)\right), \quad (3.19)$$

where $N(v_p)$ is the number of the closest data points to the block v_p given by the neighborhood selection strategy, the weights $\{\lambda_i, i = 1 \dots N(v_p)\}$ are the solution of the block-kriging system in the Gaussian scale:

$$\begin{pmatrix} \rho(d_1,d_1) & \cdots & \rho\left(d_1,d_{N(v_p)}\right) \\ \vdots & \ddots & \vdots \\ \rho\left(d_1,d_{N(v_p)}\right) & \cdots & \rho\left(d_{N(v_p)},d_{N(v_p)}\right) \end{pmatrix}\begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_{N(v_p)} \end{pmatrix} = \begin{pmatrix} \rho(v_p,d_1) \\ \vdots \\ \rho\left(v_p,d_{N(v_p)}\right) \end{pmatrix}. \quad (3.20)$$

Since the RF $Y(x)$ has a multivariate Gaussian distribution, and $\{Y(v_p), p = 1 \dots N_b\}$ and $\{Y(d_i), i = 1 \dots N_{data}\}$ are considered to originate from this field (due to the consistency assumption), the spatial distribution of $\{S(v_p), p = 1 \dots N_{data}\}$ remains multivariate Gaussian. This is a strong property for the result, but it is based on assumption (iii) of DGM 2 as well as on identification of the fixed and randomized locations for the conditioning data. The final result of the conditional simulation $T(v_p)$ is obtained through the transformation to the scale of the variable Z:

$$T(v_p) = \varphi_{v_p}\left(\frac{S(v_p)}{r_p}\right). \quad (3.21)$$

It should be noted that dividing by r_p in (3.21) does not pose a problem since only values $r_p > 0$ are considered. The two methods described above for conditioning call for a strong assumption of identifying the value of the RF $Y(x)$ at a randomized location with the value at a fixed location. It is possible to avoid this assumption in the direct form, considering that the conditioning data $(d_i, z_i), i = 1 \dots N_{data}$ are given directly at randomized locations within the blocks. This assumption seems to make little difference from the previous one, but it has certain theoretical implications for the conditioning procedure. Instead of identifying the randomized sample Y_p in (3.16) with fixed location sample $Y(d_1)$, one can consider that all the points $\{d_i, i = 1 \dots N_{data}\}$ of the conditioning dataset are randomized within the blocks, which contain these data. In this case, it is considered that the conditioning data from the very beginning were defined on randomized locations. This method for obtaining the values on point support given the values on block support is used in (Emery 2009; Emery & Ortiz 2011).

The difference of considering the conditioning data to be defined on randomized locations from the fixed locations approximation is that the data-to-data and the data-to-block covariance is now defined through averaging the point-support covariance over the blocks which contain the data. Consider two randomized locations $\underline{x}$ and $\underline{x}'$ within blocks v_p and v_q. In that case

$$cov\left(Y(\underline{x}), Y(\underline{x}')\right) = cov\left(Y(\underline{x}), Y(v_q)\right) = \frac{1}{|v_p||v_q|}\int_{v_p}\int_{v_q}\rho(x,x')dxdx' = \rho(v_p,v_q), \quad (3.22)$$

which is different from $cov(Y(x), Y(x')) = \rho(x,x')$ in the case of using the fixed locations. The variance of a randomized sample $Y(\underline{x})$ is

$$Var\left(Y(\underline{x})\right) = cov\left(Y(\underline{x}), Y(\underline{x})\right) = \frac{1}{|v_p|}\int_{v_p} \rho(x,x)dx = 1. \tag{3.23}$$

By analogy, for the randomized samples in the scale of RV $Z(x)$, $cov\left(Z(\underline{x}), Z(\underline{x}')\right) = C(v_p, v_q)$ and $Var\left(Z(\underline{x})\right) = \frac{1}{|v_p|}\int_{v_p} C(x,x)dx = C(0)$. Using the randomized locations leads to following two methods of conditioning – one in the scale of the variable $Z(x)$ and the second in the Gaussian scale.

The point-support data at randomized locations $\{Z(d_i), i = 1 \dots N_{data}\}$ corresponding to the unconditional simulation $\{Z(v_p), p = 1 \dots N_b\}$ can be obtained through Eqn. (3.16), transforming the Gaussian values $Y(\underline{x})$ to the scale of variable Z: $Z(\underline{x}) = \varphi(Y(\underline{x}))$. Given the conditioning data $(d_i, z_i), i = 1 \dots N_{data}$ defined on randomized locations, the unconditional block simulation can be conditioned through Eqn.(3.17), but the simple kriging weights $\{\lambda_i, i = 1 \dots N(v_p)\}$ in this case are a solution to the system of equations (3.24). Let us denote by $v(d_\alpha)$ the block of the unstructured grid which contains the conditioning point (d_α, z_α). Then the simple kriging system for the weights $\{\lambda_i, i = 1 \dots N(v_p)\}$ is the following:

$$\begin{pmatrix} C(0) & \cdots & C\left(v(d_1), v\left(d_{N(v_p)}\right)\right) \\ \vdots & \ddots & \vdots \\ C\left(v(d_1), v(d_{N(v_p)})\right) & \cdots & C(0) \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_{N(v_p)} \end{pmatrix} = \begin{pmatrix} C\left(v_p, v(d_1)\right) \\ \vdots \\ C\left(v_p, v\left(d_{N(v_p)}\right)\right) \end{pmatrix} \tag{3.24}$$

This approach to conditioning has the advantage of not using the additional assumption that the value at randomized location $Y(\underline{x})$ obtained through Eqn. (3.16) coincides with the value at a fixed location, but considers that the data were originally defined on the randomized locations, so that the information about their exact locations is ignored from the beginning. It is visible, that the systems of equations (3.18) and (3.24) are similar and the system (3.24) tends to system (3.18) when the size of the blocks that contain the data values tend to 0. Indeed, $C(v(d_\alpha), v_q) = \frac{1}{|v_p||v_q|}\int_{v_p}\int_{v_q} C(x,x')dxdx' \to C(d_\alpha, v_q)$ when $|v_p| \to 0$. Thus, the difference between using the randomized locations and the fixed locations approaches for conditioning diminishes with the decrease of the sizes of the blocks that contain the conditioning data.

The randomized location approach can also be used for conditioning the realizations of DGM 2 in the Gaussian scale. Conditioning is done with Eqn. (3.19),

and the kriging weights $\{\lambda_i, i = 1 \dots N(v_p)\}$ originate from the following system of equations:

$$\begin{pmatrix} 1 & \cdots & \rho\left(v(d_1), v\left(d_{N(v_p)}\right)\right) \\ \vdots & \ddots & \vdots \\ \rho\left(v(d_1), v(d_{N(v_p)})\right) & \cdots & 1 \end{pmatrix} \begin{pmatrix} \lambda_1 \\ \vdots \\ \lambda_{N(v_p)} \end{pmatrix} = \begin{pmatrix} \rho\left(v_p, v(d_1)\right) \\ \vdots \\ \rho\left(v_p, v\left(d_{N(v_p)}\right)\right) \end{pmatrix} \tag{3.25}$$

The final result of the conditional simulation on the unstructured grid is obtained through Eqn. (3.21). In the similar way for the conditioning in the scale of RV Z, the results of conditioning on randomized locations in the Gaussian scale tend to the results of conditioning on fixed locations when the sizes of the blocks that contain the conditioning data decrease.

3.6 Properties of conditional simulations

Let us study the mean value of the conditional simulations introduced in the previous chapter. We start from the case when the conditioning is performed in the scale of RV $Z(x)$. Let us reformulate Eqn. (3.17) in a different way:

$$T(v_p) = \Sigma_{i=1}^{N(v_p)} \lambda_i z_i + \left(Z(v_p) - \Sigma_{i=1}^{N(v_p)} \lambda_i Z(d_i)\right) \tag{3.26}$$

In this form it is easy to see that the conditional expectation of $T(v_p)$ given the data $(d_i, z_i), i = 1 \dots N(v_p)$ is equal to the simple kriging prediction of block v_p given these data:

$$\begin{aligned} E\left[T(v_p) \middle| Z(d_i) = z_i, i = 1 \dots N(v_p)\right] \\ = E\left[\Sigma_{i=1}^{N(v_p)} \lambda_i z_i \middle| Z(d_i)\right] \\ + E\left[\left(Z(v_p) - \Sigma_{i=1}^{N(v_p)} \lambda_i Z(d_i)\right) \middle| Z(d_i)\right] = \Sigma_{i=1}^{N(v_p)} \lambda_i z_i \end{aligned} \tag{3.27}$$

The same situation holds when the conditioning is done on randomized locations, the only difference is that the kriging weights in that case originate from system (3.24). In case if conditioning is done in the Gaussian scale either on fixed or randomized locations, the expected mean value of the conditional simulations is different. The following theorem provides an analytical expression for the mean of simulations, when the conditioning is done in the Gaussian scale.

Theorem

The block mean value of the conditional simulation when the conditioning is performed in the Gaussian scale is given with the following formula

$$E[T(v)|\{z_i\}] = \sum_{i=0}^{\infty} \varphi_i r^i \left(1 - \frac{\sigma_{sk}^2}{r^2}\right)^i \chi_i \left(\frac{Y^*(v)}{r\sqrt{1-\frac{\sigma_{sk}^2}{r^2}}} \right), \quad (3.28)$$

where r is the change of support coefficient for block v_p, $Y^*(v)$ is the simple kriging prediction for the block v in the Gaussian scale, σ_{sk}^2 is the variance of the simple kriging prediction.

Proof

For simplicity of notations, we will consider that the conditioning of the unconditional value $Y(v)$ for block v is done with respect to prior information $Y_1 = y_1, \dots, Y_{N(v)} = y_n$ which can be considered either on fixed or on randomized locations $d_1 \dots d_{N(v)}$.

$$E[T(v)|\{z_i\}] = E\left[\varphi_v\left(\frac{S(v)}{r}\right)\middle|\{y_i\}\right] = E\left[\sum_{i=0}^{\infty}\varphi_i r^i \chi_i\left(\frac{S(v)}{r}\right)\middle|\{y_i\}\right]$$

$$= \sum_{i=0}^{\infty}\varphi_i r^i E\left[\chi_i\left(\frac{S(v)}{r}\right)\middle|\{y_i\}\right]$$

In the general case, computing $E\left[\chi_i\left(\frac{S(v)}{r}\right)\middle|\{y_i\}\right]$ is problematic. However, by the cost of introducing an additional assumption about joint multigaussianity of the vector $(Y(v), Y_1, \dots Y_{N(v)})$ for a given block v, one can derive an analytical formula for $E[T(v)|\{z_i\}]$. In case if the conditioning is done on fixed locations, this assumption is valid since the RF $Y(x)$ has a multivariate Gaussian distribution. For conditioning on randomized locations, this assumption is an approximation and it is not obvious, how restrictive it is.

In the theoretical model considered and under the given assumptions, the conditional distribution of $\frac{S(v)}{r}$ given $\{Y_i, i = 1 \dots N(v)\}$ is Gaussian with parameters $\left(\frac{Y^*(v)}{r}, \frac{\sigma_{sk}^2}{r^2}\right)$. Indeed,

$$S(v) = Y(v) + \Sigma_{i=1}^{N(v)}\lambda_i\left(y_i - Y(d_i)\right) = \Sigma_{i=1}^{N(v)}\lambda_i y_i + \left(Y(v) - \Sigma_{i=1}^{N(v)}\lambda_i Y(d_i)\right) \Rightarrow$$

$$E[S(v)|y_i] = \Sigma_{i=1}^{N(v)}\lambda_i y_i = Y^*(v)$$

$$Var(S(v)|y_i) = E\left[\left(Y(v) - \Sigma_{i=1}^{N(v)}\lambda_i Y(d_i)\right)\right]^2 = \sigma_{sk}^2$$

Thus using a formula from the Appendix of (Chilès & Delfiner 2012), one can derive

$$E\left[\chi_i\left(\frac{S(v)}{r}\right)\middle|\{y_i\}\right] = \int_{-\infty}^{+\infty} \chi_i\left(\frac{Y^*(v)}{r} + \frac{\sigma_{sk}}{r}x\right) g(x)dx = \left(1 - \frac{\sigma_{sk}^2}{r^2}\right)^i \chi_i\left(\frac{Y^*(v)}{r\sqrt{1 - \frac{\sigma_{sk}^2}{r^2}}}\right),$$

which gives the desired result. ■

It is clear that the mean of conditional simulations when conditioning in the Gaussian and in the Z scales is different. Formulas (3.27) and (3.28) can be used to perform the quality check of the simulation procedures.

3.7 Discussion

Conditional simulations with both DGM 1 and DGM 2 require the conditional independence assumption in order to deal with the multiple samples inside a block. As noted by Xavier Emery in personal correspondence, the conditional independence assumption leads to the same DGM 2 equation as assumption (iii). Indeed, for a block v with change of support coefficient r and two randomized locations $\underline{x}$ and $\underline{x}' \neq \underline{x}$ within v, from the conditional independence (3.16) follows

$$cov\left(Y(\underline{x}), Y(\underline{x}')\right) = cov\left(rY_v + \sqrt{1 - r^2}\xi_1, rY_v + \sqrt{1 - r^2}\xi_2\right) = r^2 var(Y_v), \quad (3.29)$$

where ξ_1 and ξ_2 are two independent standard Gaussian random variables. On the other hand, due to linearity of covariance

$$cov\left(Y(\underline{x}), Y(\underline{x}')\right) = var\left(Y(v)\right). \quad (3.30)$$

From (3.29) and (3.30) using $var(Y_v) = 1$ and since Y_v is standard Gaussian, one derives Eqn. (3.12) $Y(v) = rY_v$, which is the basis of DGM 2. As a consequence, it follows that the conditional independence assumption and assumption (iii) are closely related, and applying the conditional independence assumption for conditioning the DGM 1 realizations is similar to applying locally the assumption (iii) for the blocks containing the data samples.

Table 3.1 summarizes the assumptions and properties of DGM 1 and DGM 2 for generating unconditional simulations. Table 3.2 summarizes the properties of various conditioning methods.

Table 3.1 Summary of DGM 1 and DGM 2 for unconditional simulations

Model	DGM 1	DGM 2
Assumptions	(i), (ii)	(i), (ii), (iii)
Advantages	Correct reproduction of covariance in the Z scale for all blocks, internal consistency of the model.	Simplified formulas and computations (no need to solve 3.9 and 3.10), the covariance matrix of the Gaussian vector is always positive semi-definite (if a valid covariance function was used).
Disadvantages	Additional computations required (solving 3.9 and 3.10), the covariance matrix of the Gaussian random vector can be not positive semi-definite.	Covariance between the block values in the Z scale is biased.

Table 3.2 Summary of conditioning methods

Method		Required Assumptions	Advantages	Disadvantages
Randomized locations	**Z scale**	Conditional independence, input samples are given on randomized locations.	The least restricting set of assumptions.	Computing covariance between samples requires computing the block to block covariance, obtained after conditioning values can be out of range of validity for RV Z.
	Y scale	Conditional independence, input samples are given on randomized locations, (iii) for blocks correlated with the data samples.	Assumption (iii) is used only locally, gives a consistent model for simulations, the obtained values are always in range of validity for Z.	Computing covariance between samples requires computing the block to block covariance, using (iii), although locally.
Fixed locations	**Z scale**	Conditional independence, correspondence between a randomized sample and a fixed location sample.	Fixed locations do not require computing integrals over blocks, the input samples are treated "as is", information about their locations is not lost, requires less assumptions than Y scale conditioning.	The obtained after conditioning values can be out of range for RV Z, reduced accuracy when the blocks with conditioning data are large relative to covariance range.
	Y scale	Conditional independence, correspondence between a randomized sample and a fixed location sample, (iii) for the blocks correlated with the data samples.	Fixed locations do not require computing integrals over blocks, the input samples are treated "as is", information about their locations is not lost, Obtained values always in range.	Requires (iii) locally, reduced accuracy when the blocks with conditioning data are large relative to covariance range.

3.8 Testing the conditioning methods

Conditioning of the DGM-based simulations for a standard lognormal variable is considered for the unstructured grid in 3D with 10 local grid refinements from Figure 3-6. The grid dimensions are 20km x 20km x 40m, it is composed of 3,546 Voronoï polygon cells and includes 10 local grid refinement regions. The bounding box size for the smallest and the biggest blocks sizes are 36x42x8 m and 1,035 x 1,052 x 8 m respectively. Since the values of the lognormal random variable are always positive, the method of conditioning the simulations with block kriging in the scale of the RV $Z(x)$ is not applicable – although it preserves the block to block covariance, negative values can be produced, which is unacceptable for a lognormal distribution. In this series of tests we compare the conditioning in the Gaussian scale with fixed and with randomized locations for the support point samples. We start from a simple dataset of 10 samples on standard lognormal distribution placed in the centers of the local grid refinement zones Figure 3-15.

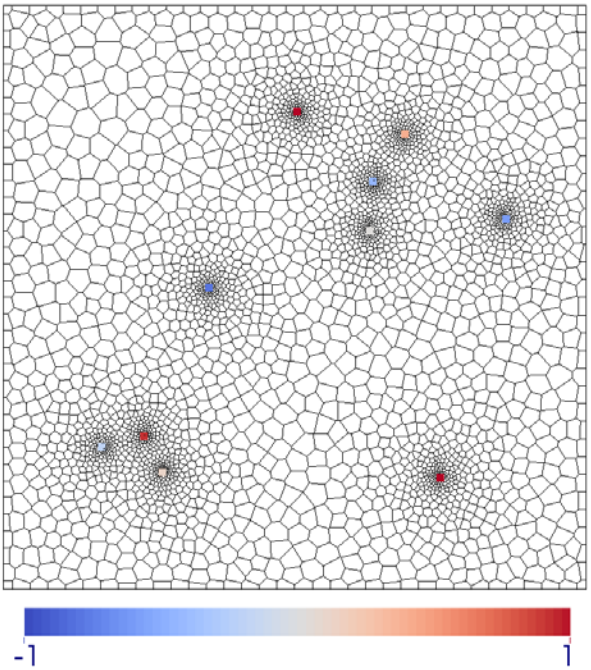


Figure 3-15. Point support samples placed in the well refinement zones. Values after the normal score transform shown.

For the first test a long range spherical covariance function ρ with ranges $(5{,}5{,}0.2)km$ was used. Figure 3-16a demonstrates an unconditional simulation generated with DGM 1 for the given inputs and simple block kriging prediction obtained from the sampled data in the scale of RV Z (Figure 3-16b). The corresponding conditional realizations obtained with conditioning on fixed and randomized locations in the Gaussian scale are depicted on Figure 3-16c,d. It is visible, that local patches were applied to the unconditional realization in order to honor the data samples. The results obtained with the two conditioning methods are

visually close, although some difference can be noticed and the origin of this difference will be discussed below.

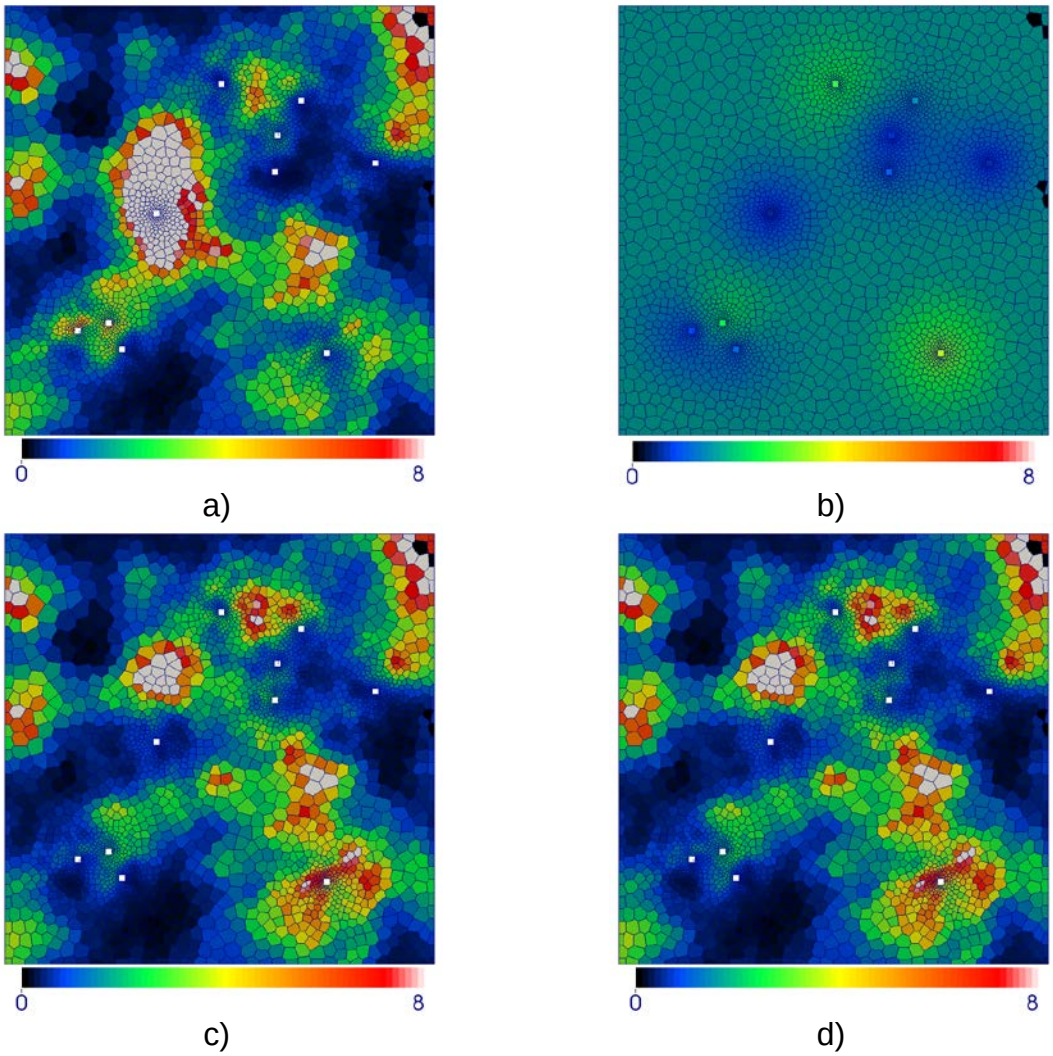


Figure 3-16. Conditioning for spherical covariance function ρ with ranges $(5,5,0.2)km$. a) unconditional realization b) simple kriging block predictions from sample data in Z scale c) conditioning on fixed locations d) conditioning on randomized locations.

The simple kriging predictions for the block values $Y(v)$ in the Gaussian scale for fixed and randomized locations as well as the simple kriging variance are depicted on Figure 3-17a-b. Although visually the kriging predictions and variance are equal, one can still observe a small decrease of the kriging variance when fixed locations are used. The origin of this behavior is in the kriging system used for conditioning. For the given inputs the covariance values $\rho\left(v_p, v(d_\alpha)\right)$ in the right side of (3.24) are lower than the corresponding covariance $\rho(v_p, d_\alpha)$ in (3.20), which leads to higher prediction variance in the case of randomized locations and, as a consequence, a weaker impact of samples on the block predictions. This effect will be better demonstrated in the next test.

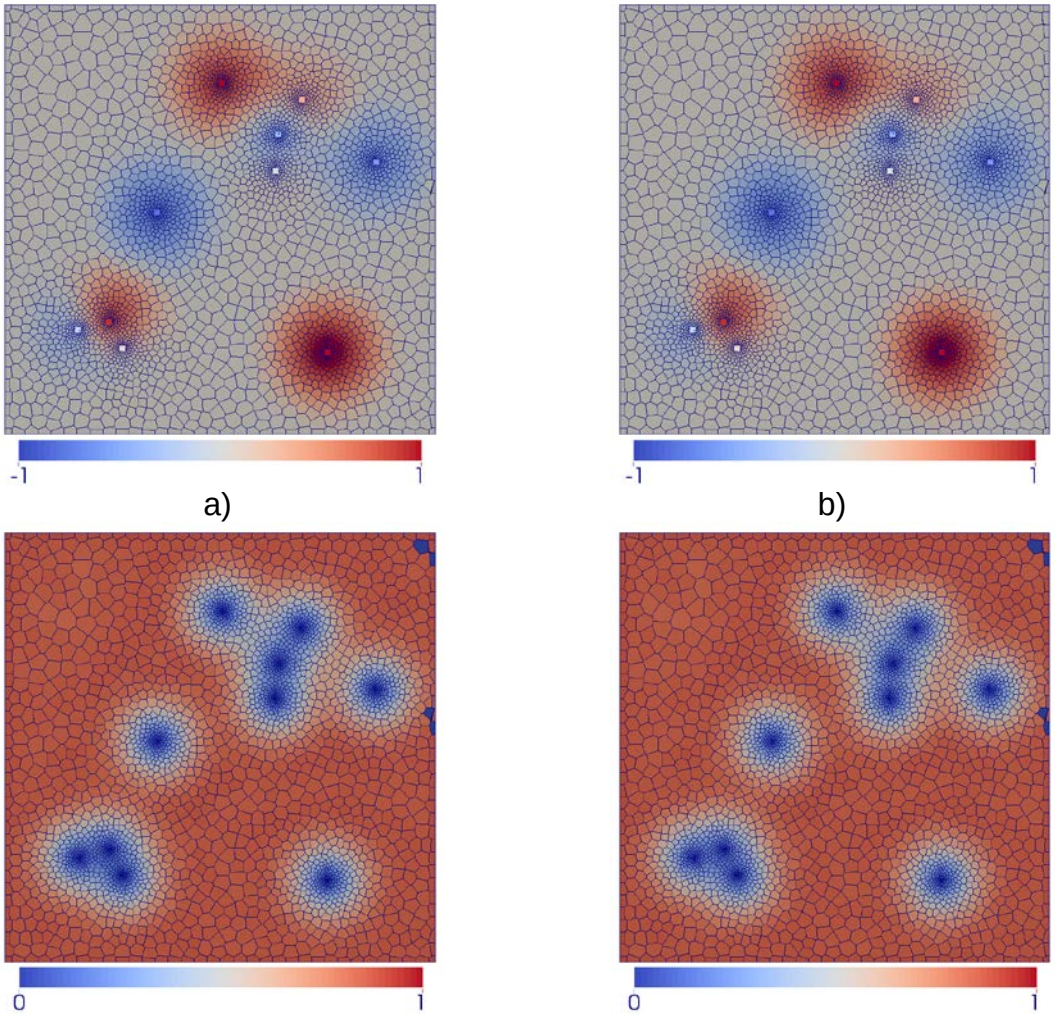


Figure 3-17. Block kriging predictions and kriging variance in the Gaussian scale a) kriging with fixed locations b) kriging with randomized locations. c) kriging variance for fixed locations d) kriging variance for randomized locations.

Since the range of the covariance function is significantly larger than the block sizes in the grid, the block kriging variance on Figure 3-17b,c visually depends only on the distance from the samples, so the impact of the size of the block on the prediction variance is not clearly observed. In order to demonstrate the block volume impact, a covariance function ρ with shorter range $(2km, 2km, 40m)$ is used for simulation with conditioning on the same dataset of samples. This covariance choice provides visual spatial continuity in horizontal direction coupled with noticeable change of support effect. The unconditional and conditioned realizations with fixed and randomized location realizations are depicted on Figure 3-18, the corresponding kriging predictions in the Gaussian scale and kriging variance are depicted on Figure 3-19.

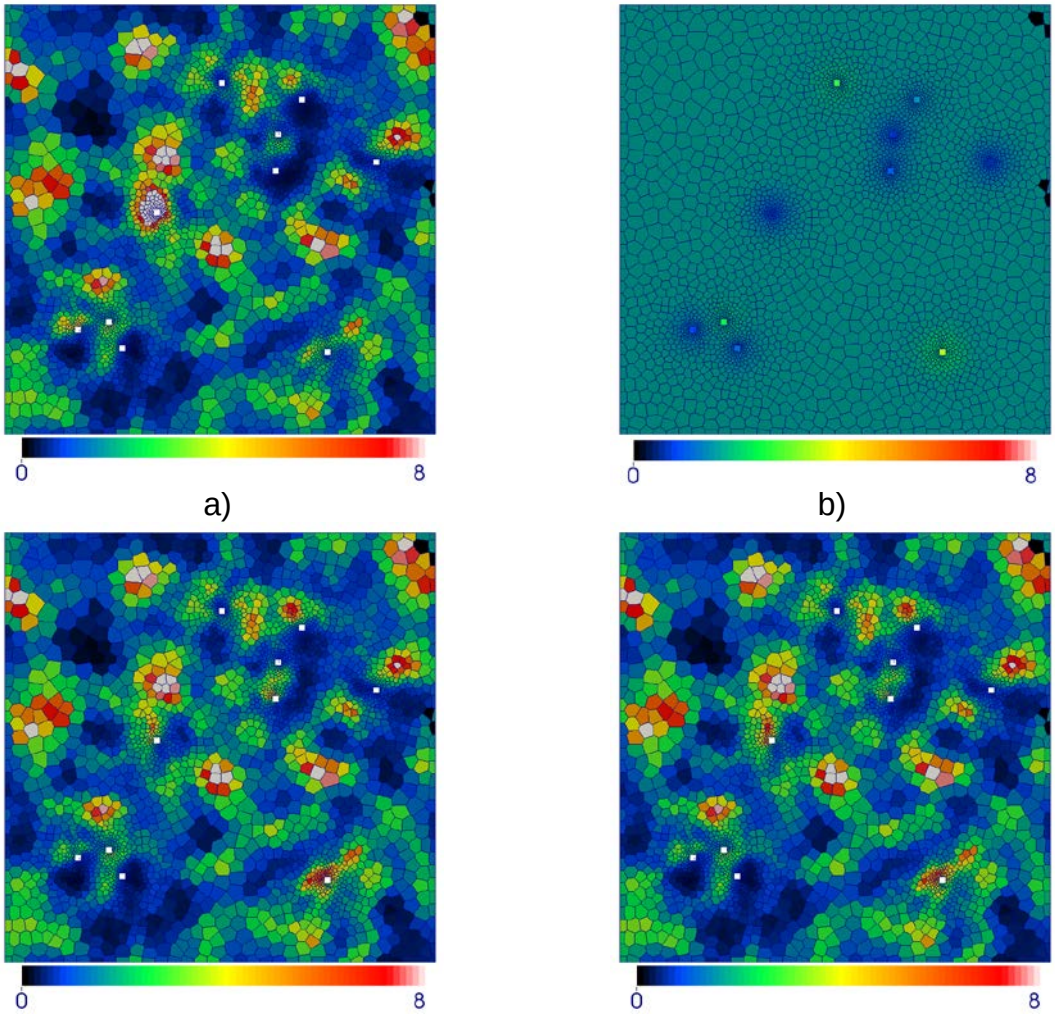


Figure 3-18. Conditioning for spherical covariance function ρ with ranges $(2,2,0.04)km$. a) unconditional realization b) simple kriging block predictions from sample data in Z scale c) conditioning on fixed locations d) conditioning on randomized locations.

The observed behavior of the kriging variance for the short range covariance (Figure 3-19c,d) is different from those in the previous test. It is visible on Figure 3-19c,d that the prediction variance decreases not only in the vicinity of the data samples, but also in the regions of large blocks even though those are distanced from the samples (as in the left top corner on Figure 3-19c). In this case the impact of the block size on the kriging variance map is observed. Also, the kriging variance maps on Figure 3-19c,d demonstrate more clearly the difference between using fixed and randomized locations for kriging – in the vicinity of the data samples prediction variance for fixed locations is lower.

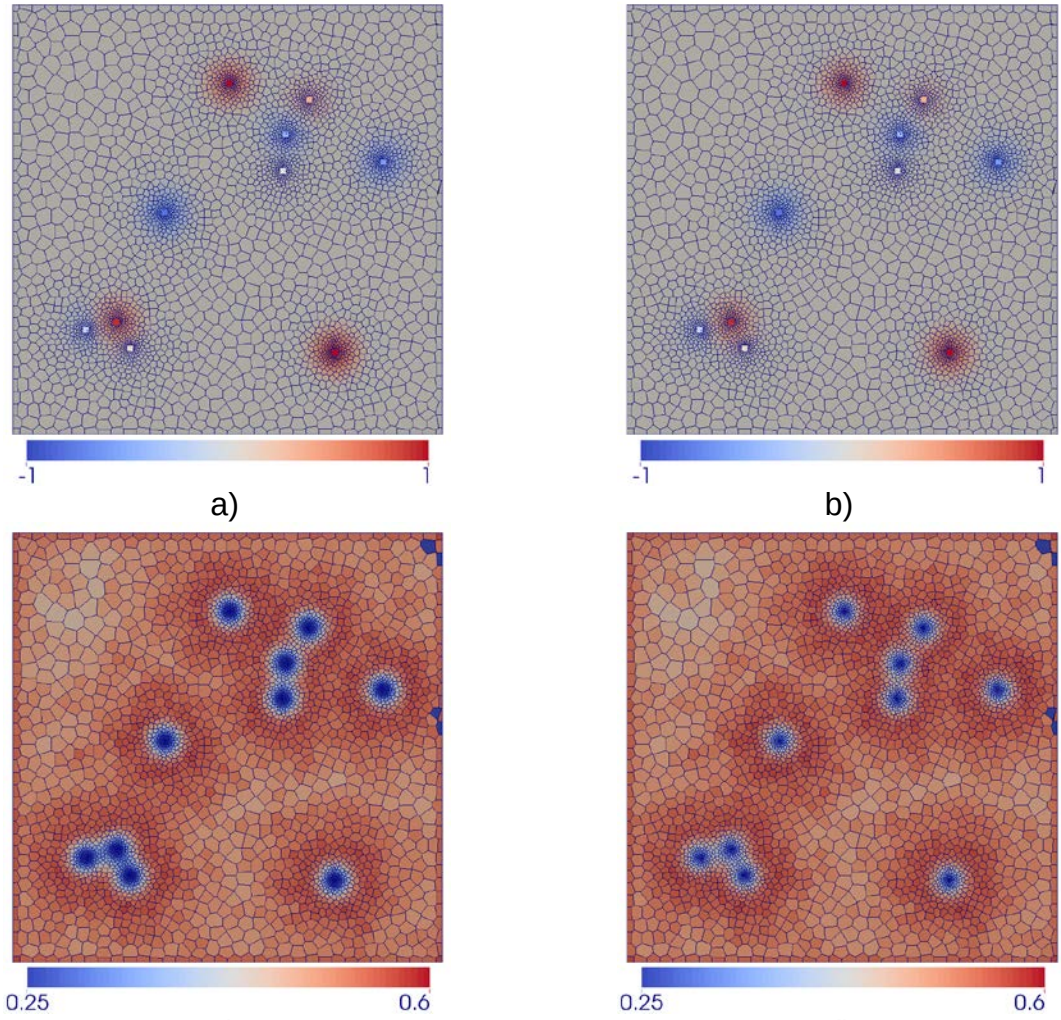


Figure 3-19. Block kriging predictions and kriging variance in the Gaussian scale a) kriging with fixed locations b) kriging with randomized locations. c) kriging variance for fixed locations d) kriging variance for randomized locations.

Let us consider a test in 3D on a grid with multiple layers for conditioning a DGM-based simulation with multiple data samples inside each block. The same grid geometry is used for the test, but now the grid contains 5 layers of thickness $8m$ each (see Figure 3-21). The dataset depicted on Figure 3-20 consists of 10 vertical wells, with samples located every $50cm$. The sample data was simulated with standard lognormal distribution and covariance function ρ with ranges $(2km, 2km, 10m)$ using the Cholesky decomposition approach. As in previous tests, we start from demonstrating an unconditional simulation, kriging in the scale of RV Z and conditioned to data simulations on Figure 3-22.

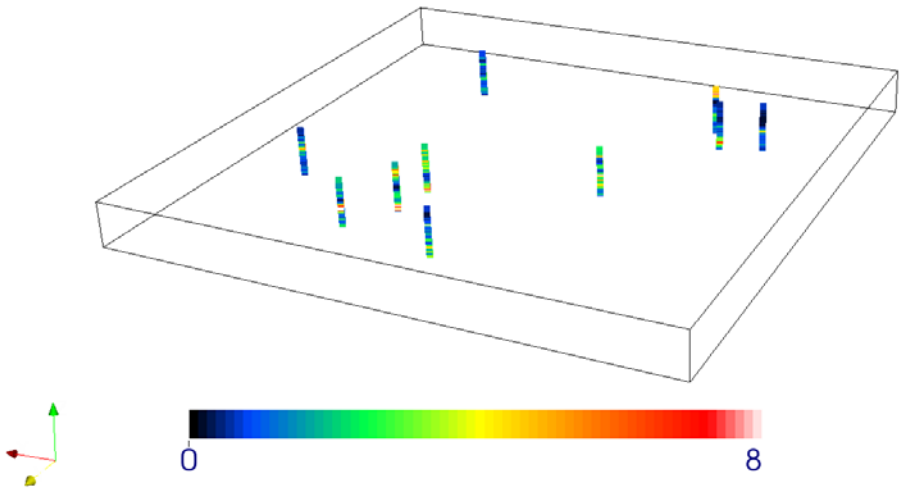


Figure 3-20. Samples in 3D.

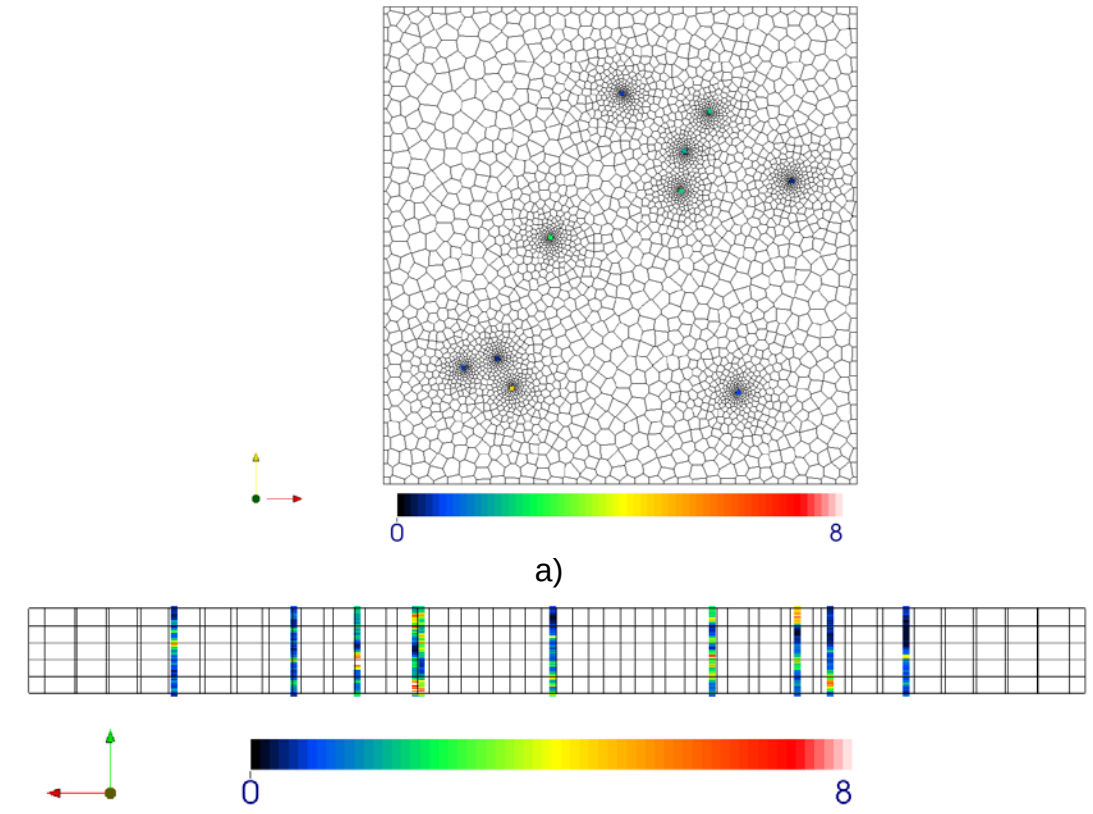
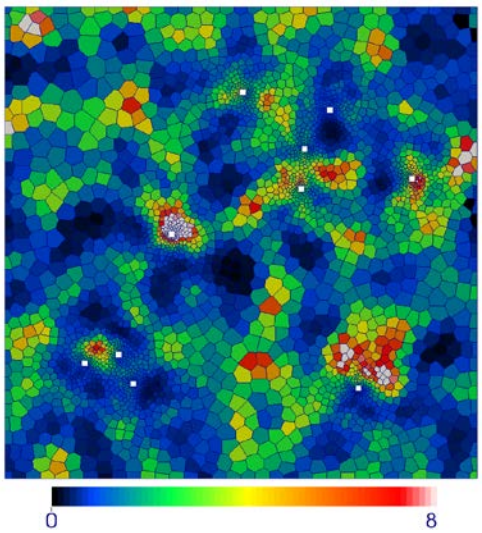


Figure 3-21. Samples locations relative to the grid. a) top view b) lateral view.

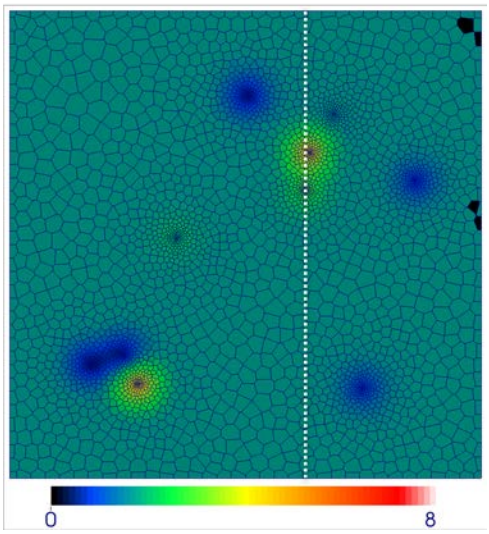
The considered dataset has a total number 800 of samples and kriging can still be performed in a unique neighborhood. Unfortunately, this approach is not general, since in the industrial applications it is not uncommon to have many thousands of conditioning data points (or blocks) and in that case only a limited number of closest samples are usually taken into account. The prediction quality when local neighborhood is used can remain high due to the screening effect – the

closest samples “screen out” other samples which justifies this method. A fundamental work on the screening effect can be found in (Stein 2002) where the author proves the existence of the screening effect and provides the conditions under which it is observed.

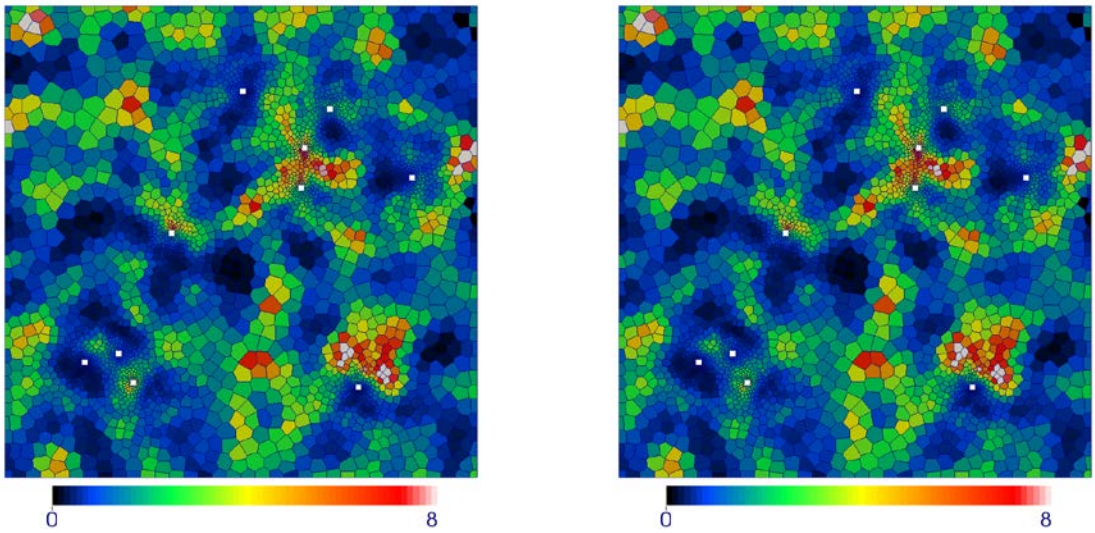
Taking a limited number of neighboring samples however leads to discontinuities in the kriging prediction maps and abnormal kriging weights, the effect of which was extensively discussed in literature (Babak & Deutsch 2008; Chilès & Delfiner 2012). Numerous solutions were proposed for producing continuous kriging maps, such as successive augmentation of the kriging neighborhood (Babak & Deutsch 2008), applying kernel smoothing functions on the kriging weights (Gribov & Krivoruchko 2004) and considering the outermost data samples in the neighborhood spoiled with a random error (Rivoirard & Romary 2011). In our work we apply the approach proposed by Rivoirard and Romary (2011) - the samples in the local neighborhood of each block are selected for kriging (the size of the neighborhood is linked to the covariance ranges) and the kriging weights of the outermost samples in the local neighborhood are forced to tend to zero. This is achieved by adding to the diagonal elements of the simple kriging matrix (3.20) or (3.22) penalty terms which vary from 0 for the close samples to infinity for the samples approaching the boundary of the neighborhood, the details can be found in Rivoirard and Romary (2011). Comparison of kriging in unique neighborhood with kriging in local neighborhoods in this case does not demonstrate any significant differences.



a)



b)



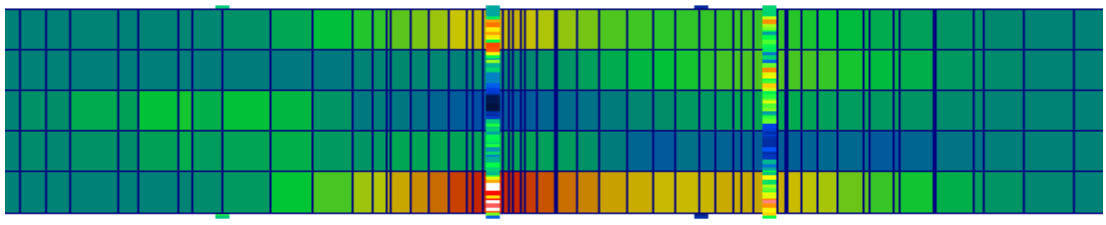
c) d)

Figure 3-22. Conditioning on a multiple layers for spherical covariance function ρ with ranges $(2km, 2km, 10m)$: a) unconditional realization b) simple kriging in the Z scale c) conditioning on fixed locations d) conditioned on randomized locations.

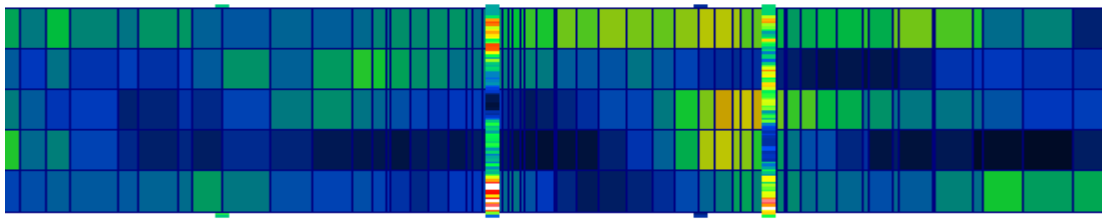
Careful analysis of Figure 3-22 demonstrates that the unconditional simulation was corrected in order to fit the sample data. Since the modifications were done by simple kriging, the covariance structure of the simulated random field is preserved. Conditioning on fixed and randomized locations is almost undistinguishable visually.

In order to demonstrate the effect of conditioning in the vicinity of the wells, a vertical cross-section of the model was done (see Figure 3-23). The position of the cross-section in the reservoir is marked on Figure 3-22b with white dashed line. Figure 3-23 demonstrates the simple kriging prediction results in the vicinity of the wells, unconditional and conditioned simulations. It is visible, that the wells data are respected in both variants of conditioning.

a)



b)



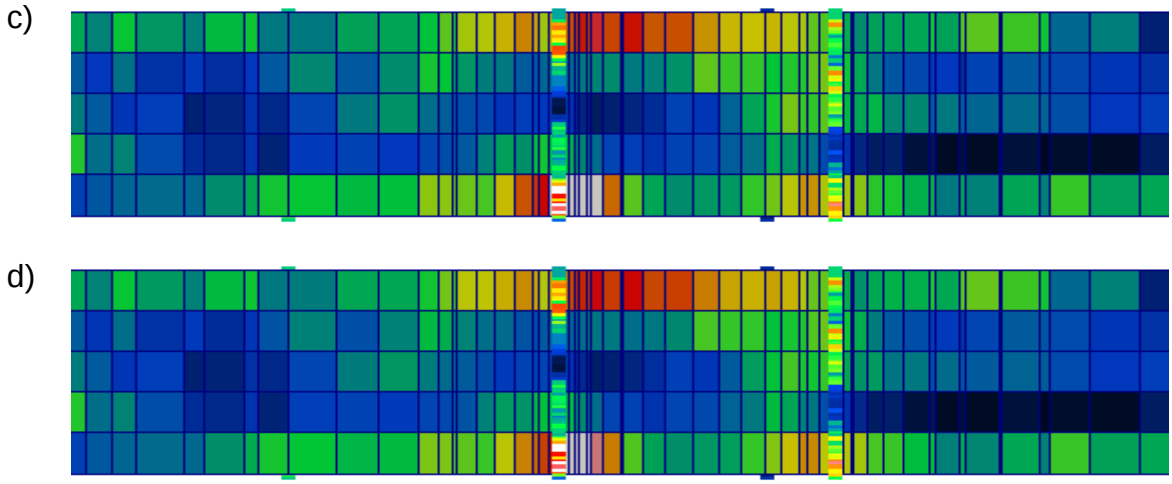


Figure 3-23. Grid cross-section a) kriging in Z scale b) unconditional simulation c) conditioning on fixed locations d) conditioning on randomized locations.

As demonstrated in this chapter, DGM-based simulations can be effectively conditioned through simple kriging in order to respect sample data defined on quasi-point support. When conditioning on quasi-point support, a choice should be done between conditioning in the Gaussian scale or in the scale of RV Z, as well as between conditioning on fixed or randomized samples. Conditioning in the scale of RV Z, although this preserves the covariance of the simulated random field, possibly leads to negative values which is often incompatible with the simulated distribution, which makes utilization of conditioning in Gaussian scale a more general solution. From the point of view of DGM, the randomized location approach for conditioning is theoretically better justified, since the method of obtaining the point-support values of the unconditional block-support random field through (3.16) is coherent with samples defined on randomized locations. On the other hand, for a practitioner it might be unacceptable to lose the information about the exact location of the sample data in order to ensure the theoretical integrity of the model. In addition, conditioning on randomized locations requires more extensive numerical computations – covariance between two randomized samples requires computing the covariance between the blocks, containing these samples, which is a problem of computing multidimensional integrals.

The main difficulty of conditioning the block-support simulations to quasi – point sample data is the derivation of the coherent point-support values at the data locations for the unconditional block support simulation. In case of DGM-based simulations, this problem is solved through the theory of randomized samples in the blocks and assumption of conditional independence of the samples within a single block given the average value of this block.

An alternative solution to the problem of conditioning a block support simulation on quasi-point samples was proposed by Journel and Huijbregts (1978) for simulations on regular grid. In their approach instead of deriving a coherent point-

support data, the authors propose deriving the conditioning block support samples from the given point-support data in the following form:

$$z_v'(d_\alpha) = z_{vK}^*(d_\alpha) + \varepsilon_v \quad (3.31)$$

Where $z_{vK}^*(x_\alpha)$ is the kriging value of block v given the sample data $\{z(d_\alpha), \alpha = 1 \dots N_{data}\}$ and ε_v is the variance correction terms. This approach can also be used for conditioning of DGM-based simulations.

As it can be seen from numerous tests in this section (Figure 3-19 for instance), conditioning on quasi-point samples is not “exact” in the sense that the value of the block, containing a single sample is not equal to the value of this sample. This behavior of conditioning procedure is reasonable since in the general case correlation between a fixed or randomized quasi-point sample and the average value of the block in the general case remains inferior 1. In case if the values at the blocks, containing sample data should not have any variability (which appears sometimes in practical applications), conditioning should be performed directly on block support. For that the sample data should first be upscaled to the block support in order to be used for conditioning the simulation. Conditioning can be performed with simple block kriging in Gaussian scale or in the Z scale, the implementation is straightforward.

The final choice of the conditioning strategy depends on the case study and on the choice of the software user. If the user considers that the blocks which contain the well data should not exhibit any variation but reproduce exactly the known average values, the strategy of conditioning on blocks can be selected. However, even having a large number of sample points in a block does not imply that the average value of the block is known exactly and considering this uncertainty effect can increase the overall quality of the modeling workflow. In the case when the size of the block with well data is small relative to the covariance range one can expect that different conditioning methods provide similar results.

3.9 Simulating non-additive variables

Geostatistical simulations on unstructured grids considered in this thesis assume that the equivalent block value $Z(v)$ for a point – support field $\{Z(x), x \in v\}$ is expressed as a volumetric average of this field: $Z(v) = \frac{1}{|v|}\int_v Z(x)dx$, so that the simulated variable averages linearly over the volume. In practice this property does not hold for an important range of physical parameters.

One of such parameters, being of exceptional importance in petroleum applications is permeability, which does not average linearly over the blocks of the reservoir models (following the convention, we denote permeability as $K(x)$). Indeed, the equivalent permeability of the block can be defined as such a value of permeability for the block, which enables correct reproduction of the fine scale flow behavior (see Wu et al. (2002) for a similar definition). A more formal definition of the equivalent block permeability can be derived from (Farmer 2002). According to this author, upscaling is approximating one system of partial differential equations with another system, often of the same form, which can be solved with fewer resources. Following this definition of upscaling, the equivalent block permeability for a point – support field $\{K(x), x \in v\}$ is the corresponding permeability K_v in the approximating flow equation on the scale of blocks. It should be noted that this equivalent value depends not only on the fine scale permeability field, but also on the flow equations considered, the classical example of this dependence being the type of the boundary conditions used when dealing with the steady state flow equation.

The equivalent permeability of a block K_v generally cannot be expressed analytically from the fine scale values $K(x)$. However, approximate solutions exist. One of the commonly used approximations is the power transform, which assumes that the equivalent permeability K_v can be derived from the fine scale values through the following relationship for some fixed factor ω

$$K_v = \left(\frac{1}{|v|}\int_v K(x)^\omega dx\right)^{\frac{1}{\omega}}. \tag{3.32}$$

This approach was justified through the theory of the stochastic differential equations by Noetinger and Haas (1996). In their paper, the authors also give an explicit formula for the power ω which should be used for the transform. An important implication of their paper is that the power ω can be considered to depend only on the global anisotropy ratio $\lambda = \sqrt{\frac{k_v}{k_h}\frac{L_h}{L_v}}$, where $\frac{k_v}{k_h}$ is the vertical to horizontal permeability ratio in the domain of investigation, L_h and L_v denote the horizontal and vertical covariance ranges respectively. This, in turn, implies that the parameter ω in transformation (3.32) does not depend on the block v considered. Certainly, this assumption is a rough approximation which, following the authors, is based on the

hypothesis that the permeability variance is small and the domain of investigation is large with respect to the correlation lengths of the medium.

A fundamental double inequality bounding K_v is known: K_v is bounded by the harmonic average of $K(x)$ on the left and arithmetic average of $K(x)$ on the right (Matheron 1984). The homogenization method through formula (3.32) is coherent with this fundamental inequality, since harmonic and arithmetic mean correspond to choices $\omega = -1$ and $\omega = 1$ respectively. Three different analytical formulas for computing equivalent permeability, including that of form (3.32) were proposed in (Matheron 1967; Matheron 1993). In practice, it is not uncommon in the petroleum industry to use for upscaling permeability in horizontal plane $\omega = 1$ (arithmetic average) and in the vertical plane $\omega = -1$ (harmonic average). In the general case, when formula (3.32) is used, selection of ω should be done for every case study independently.

Another argument for applying upscaling formula (3.32) for geostatistical simulations can be found in the work (De Lucia et al. 2009). These authors compare several analytical and numerical methods for upscaling absolute permeability on a 2D Voronoï polygon grid populated with stochastic medium. Although power averaging in the form (3.32) was not tested in this paper, a similar analytical method proposed by Matheron (1967) was considered

$$K_v = \sqrt{\mu_a \mu_h}, \tag{3.33}$$

where μ_a and μ_h denote arithmetic and harmonic mean of $K(x)$ respectively. An important conclusion of (De Lucia et al. 2009) is that the upscaling method has significantly less impact on the flow behavior than the structure of the hydrodynamic grid used for flow simulation and the spatial variability of the medium.

The advantage of formula (3.32) for determining the equivalent permeability is that it enables one to shift from the problem of simulating a vector of equivalent block permeability K_v on a grid $\{v_i, i = 1 \dots N_b\}$ to problem of simulating of a vector of block averages $\left\{\frac{1}{|v_p|}\int_{v_p} K(x)^\omega dx\,, p = 1 \dots N_b\right\}$ for a new RV $K(x)^\omega$ followed by application of a post – processing procedure. The simulation can proceed in the following manner:

1) Define a new RF $S(x) = K^\omega(x)$ and derive the density and covariance function for $S(x)$
2) Simulate with a DGM – based method a random vector $\left\{S(v_i) = \frac{1}{|v_p|}\int_{v_p} S(x)dx\,, p = 1 \dots N_b\right\}$ - the average values of $S(x)$ over the blocks.
3) Apply a post-simulation transformation to derive the vector $\{K_{v_p}, p = 1 \dots N_b\}$: $K_{v_p} = S(v_p)^{\frac{1}{\omega}}$

Let us consider the first step of this algorithm. Let the fine scale permeability $K(x)$ be defined by the Gaussian anamorphosis $K(x) = \varphi(Y(x))$, and let $C(x, x')$ be the covariance function of $K(x)$ and $\rho(x, x')$ be the covariance function of $Y(x)$. In that case, $S(x) = K(x)^{\omega} = \varphi^{\omega}(Y(x)) = \psi(Y(x))$ defines the Gaussian anamorphosis function for $S(x)$. From the said above, one derives $\psi(y) = \varphi^{\omega}(y)$ and $S(x)$ has the same covariance $\rho(x, x')$ in the Gaussian scale as $K(x)$.The covariance $C_{\omega}(x, x')$ of $S(x)$ can be derived from $\rho(x, x')$ and from the decomposition of the transformation function $\psi(y)$ into Hermite polynomials. Given the Gaussian anamorphosis $\psi(y)$ and covariance function $\rho(x, x')$, DGM-based simulation on an unstructured grid can be performed with a DGM 1 or DGM 2 based algorithm.

It is convenient when a simulation algorithm on an unstructured grid is designed as a “black box” which produces simulations given the inputs and hides the implementation details from the user. It is more preferable when the “black box” takes as input the CDF or PDF of the simulated RV and not the Gaussian anamorphosis (users of the geostatistical software are rarely used to working with the anamorphosis functions). In order to match the input format of the simulation “black box”, let us derive the CDF and PDF of $S(x)$. The distribution of $S(x)$ can be found analytically from those of $K(x)$. For positive ω one has

$$H(z) = P(S(x) \le z) = P(K^{\omega} \le z) = P\left(K \le z^{\frac{1}{\omega}}\right) = F\left(z^{\frac{1}{\omega}}\right), \tag{3.34}$$

where $F(z)$ is the CDF of $K(x)$. For $\omega < 0$:

$$H(z) = P(S(x) \le z) = P(K^{\omega} \le z) = P\left(K > z^{\frac{1}{\omega}}\right) = 1 - F\left(z^{\frac{1}{\omega}}\right), \tag{3.35}$$

then the PDF of $S(x)$ can be derived, denoting $sign(\omega)$ the sign of ω:

$$\begin{aligned} \frac{\partial H(z)}{\partial z} &= \frac{sign\,(\omega)\,\partial F\left(z^{\frac{1}{\omega}}\right)}{\partial z} = sign\,(\omega)\frac{\partial F\left(z^{\frac{1}{\omega}}\right)}{\partial z^{\frac{1}{\omega}}}\frac{\partial z^{\frac{1}{\omega}}}{\partial z} \\ &= sign\,(\omega)\frac{1}{\omega}z^{\frac{1-\omega}{\omega}}F'\left(z^{\frac{1}{\omega}}\right) \end{aligned} \tag{3.36}$$

Formula (3.36) enables preparing easily the standard inputs for a DGM – based simulation algorithm when applying the power transform averaging hypothesis for the simulated variable.

3.10 Simulation of co-regionalization

The DGM-based simulation algorithms can easily be generalized for the case of simulating co-regionalized variables (Emery & Ortiz 2011). Generalization of DGM 2 approach seems to be more practical and is presented below for the case of co-simulating two variables. Consider a pair of RV $Z_1(x)$ and $Z_2(x)$ defined through their Gaussian anamorphosis functions $\varphi_1(y)$ and $\varphi_2(y)$ respectively

$$\begin{pmatrix} Z_1(x) \\ Z_2(x) \end{pmatrix} = \begin{pmatrix} \varphi_1\big(Y_1(x)\big) \\ \varphi_2\big(Y_2(x)\big) \end{pmatrix}. \tag{3.37}$$

The random variables $Y_1(x)$ and $Y_2(x)$ are dependent in the case of co-regionalization and one of the approaches for simulating these dependent variables is decomposing them into a composition of independent variables. This can be achieved in several manners, including principal component analysis (Davis 1987; Wackernagel et al. 1989), min/max autocorrelation factors (Desbarats & Dimitrakopoulos 2000) or applying a linear model of co-regionalization (Chilès & Delfiner 2012). Here we consider the linear model of co-regionalization approach (LMC).

For the random vector $\big(Y_1(x), Y_2(x)\big)$ a linear model of co-regionalization (LMC) can be fitted (Chilès & Delfiner 2012)

$$\begin{pmatrix} Y_1(x) \\ Y_2(x) \end{pmatrix} = \sum_{m=1}^{M} A_k \begin{pmatrix} Y_1^m(x) \\ Y_2^m(x) \end{pmatrix}. \tag{3.38}$$

Where A_k is a 2×2 matrix with known coefficients, M is the number of structures in LMC, the covariance function for each Gaussian RF $Y_i^m(x), i = 1{,}2\ m = 1 \dots M$ is $\rho_{(m)}(x, x')$. The random fields for different sets of indices $Y_i^m(x)$ are independent. In that case the covariance matrix of random vector $\big(Y_1(x), Y_2(x)\big)$ is expressed as follows

$$R(x, x') = \begin{pmatrix} \rho_{11}(x, x') & \rho_{12}(x, x') \\ \rho_{21}(x, x') & \rho_{22}(x, x') \end{pmatrix} = \sum_{m=1}^{M} A_k\, A_k^T\, \rho_{(m)}(x, x'). \tag{3.39}$$

Fitting LMC for the data is often a challenging task, fortunately, it is thoroughly discussed in literature, see Chilès and Delfiner (2012), among others. Once LMC is fitted for the given covariance matrix $R(x, x')$, independent univariate Gaussian random fields $Y_i^m(x), i = 1{,}2\ m = 1 \dots M$ with covariance functions $\rho_{(m)}(x, x')$ can be simulated on a given grid with any suitable method (SGS, Gibbs Propagation and others), then random fields $Y_1(x)$ and $Y_2(x)$ can be derived through the composition of the previously simulated independent fields through (3.38). The unconditional simulation of $(Y_1(x), Y_2(x))$ can then be conditioned to data by co-kriging with covariance matrix $R(x, x')$. In case of simulation on an unstructured grid composed of blocks $\{v_i, i = 1 \dots N_b\}$, the algorithm proceeds as follows:

1) Derive the Gaussian anamorphosis functions $\varphi_1(y)$, $\varphi_2(y)$ for $Z_1(x)$, $Z_2(x)$ repsectively.
2) Fit LMC for the Gaussian random vector $\left(Y_1(x), Y_2(x)\right)$ as in (3.38)
3) Compute the covariance matrix $R(x, x')$ which corresponds to fitted LMC on step 2.
4) On grid $\{v_p, p = 1 \dots N_b\}$ simulate independently for each component $Y_i^m(x), i = 1,2\ m = 1 \dots . M$ with covariance function $\rho_{(m)}(x, x')$ Gaussian random vector $\{Y_i^m(v_p) = \frac{1}{|v_p|}\int_{v_p} Y_i^m(x)dx\,, p = 1 \dots N_b\}$ with any appropriate simulation method for multivariate Gaussian random vectors (such as SGS or Gibbs Propagation algorithm).
5) Derive unconditional simulation results for $\left(Y_1(v_p), Y_2(v_p)\right), p = 1 \dots N_b$ from the random fields simulated on step 4 through (3.38).
6) For each block v_p of the model derive change of support coefficients r_{1p} and r_{2p} for RV $Z_1(x)$ and $Z_2(x)$ through (3.13) and block-dependent transform functions $\varphi_1(v, y)$ and $\varphi_2(v, y)$.
7) Derive the quasi-point support values for $Y_1(x)$ and $Y_2(x)$ through (3.16).
8) Condition the unconditional realization of $\left(Y_1(v_p), Y_2(v_p)\right)$ on the samples using cross-covariance matrix $R(x, x')$ through block co-kriging using fixed or randomized samples approach.
9) Derive the final result $\left(Z_1(v_p), Z_2(v_p)\right) = \left(\varphi_1\left(v_p, \frac{Y_1(v_p)}{r_{1p}}\right), \varphi_2\left(v_p, \frac{Y_2(v_p)}{r_{2p}}\right)\right), p = 1 \dots N_b$.

Chapter 4: Facies simulation

Résumé

Le problème de simulation de variables discrètes, comme des faciès géologiques est discuté. Une généralisation de ce problème pour les maillages non-structurés est proposée. L'auteur propose de simuler les vecteurs de proportion de faciès pour chaque cellule du maillage. L'avantage de cette approche par rapport à la simulation directe des faciès est démontré. Une généralisation de simulation pluri-Gaussienne avec le modèle Gaussien discret pour le problème de simulations des faciès sur les maillages non-structurés est présentée.

Applications of geostatistics to natural resources estimation and simulation often require dealing with discrete (categorical) variables. Discrete variables can be used to distinguish ore and waste in the mining pit, or the distribution of different facies in the reservoir, such as different types of sands and shales. Various methods for simulating discrete variables are described in the literature including sequential indicator simulation (SIS, see Alabert (1989)), iterative methods based on Markov chains (Chilès & Delfiner 2012), truncated Gaussian simulation (TGS) see (Matheron et al. 1987; Matheron et al. 1988), and pluri-Gaussian simulations (PGS), see (Galli et al. 1994).

When the volume support effect is taken into account for unstructured grids, simulating discrete variables can lead to a loss of important information. Consider for instance a simulation of ore/waste parameter over the blocks of an unstructured grid. Assigning every block a discrete variable ore/waste is less informative than attributing a proportion vector $(ore\ proportion, waste\ proportion)$. The same logic applies directly to a simulation of facies – a vector of facies proportions within the block provides more information than a single categorical variable "facies". In this work we consider that "facies simulation" on unstructured grids corresponds to a simulation of facies proportion vector for every block of the grid. In order to keep the terminology consistent with the conventionally used, we still refer to the process as "facies simulation".

Building a theoretical model for facies simulations on unstructured grids is a challenging task. A natural approach to this problem would be constructing a generalization of one of the algorithms used for regular grids – SIS, TGS or PGS. Although widely used in the petroleum industry for simulating reservoir facies, the SIS algorithm is subject to serious critics from the research community. Thus Emery (2004a) demonstrates that the reproduction of the input statistics is ensured only under very restrictive conditions. The covariance is reproduced only for the case of pure nugget effect in n-dimensional space and exponential input covariance function in 1D. The author demonstrates that even if all previously simulated nodes were taken into account for sequential simulation, the resulting multivariate distribution depends on the visiting sequence, which makes the visiting sequence of the nodes

itself a parameter of the multivariate distribution simulated. Emery (2004a) derives a conclusion that the multivariate distributions of the SIS are “baseless” and should be considered as “undefined”. Another characteristics of SIS (as noted by Chilès and Delfiner (2012)) is the ability to perform simulations even if the inputs are inconsistent and do not correspond to any random field. Another family of algorithms share this property – DSS (Oz et al. 2003) algorithms, for which there is exists no practical way to check the model consistency.

Considering the strong critics from the scientific community, attempting to generalize the SIS algorithm for simulations on unstructured grids does not seem reasonable. It is not clear how a random field model without any underlying multivariate distribution can be transferred from a regular to an irregular grid. Fortunately the situation is different for TGS and PGS where the simulated categorical RV has underlying multivariate Gaussian distribution. Another advantage of TGS and PGS is that in addition to the reproduction of covariance between the blocks, the simulation algorithms enable simulating contacts between different facies, which is used for modeling realistic geological features. This chapter provides an attempt to generalize the PGS simulation model for regular grids to geostatistical simulations on unstructured grids which consider the change of support effect. The chapter starts by describing the formalism of the pluri-Gaussian model and formulating the corresponding problem statement for unstructured grids. Generalization of TGS, which is a particular case of PGS, is straightforward.

4.1 Pluri-Gaussian model for facies

A model of K facies in the compact region D is considered. Let $Y_1(x) \sim G(0,1)$ be a standard multivariate Gaussian random field with covariance function $\rho_1(x,x')$, $Y_2(x) \sim G(0,1)$ be standard multivariate Gaussian RF with covariance function $\rho_2(x,x')$. The random fields $Y_1(x)$ and $Y_2(x)$ are independent: $Y_1(x) \perp Y_2(y)\ \forall (x,y) \in D^2 = D \times D$.

Consider also a partition of the plane R^2 into non – intersecting rectangles (with possible infinite boundaries):

$$R^2 = \cup_{i=1\dots K}\ (\Delta t_i^1 \times \Delta t_i^2), \qquad \mu\left((\Delta t_i^1 \times \Delta t_i^2) \cap \left(\Delta t_j^1 \times \Delta t_j^2\right)\right) = 0 \ \ for\ i \neq j$$

Where μ is the Lebesgue measure on the plane. Let $Z(x)$ be a discrete RF defined in region D as “facies at point x”. Then $Z(x)$, following the pluri-Gaussian model can be defined through $Y_1(x)$ and $Y_2(x)$ and the truncation diagram $\{(\Delta t_i^1 \times \Delta t_i^2), i = 1 \dots K\}$.

Definition 1
The facies at point x, $Z(x)$, is defined as follows:

$$Z(x) = \{\, i \;\; if \; Y_1(x) \in \Delta t_i^1, Y_2(x) \in \Delta t_i^2 \; i = 1 \dots K\} \tag{4.1}$$

Corollary 1

The indicators of facies i coincides with the indicators of the random vector $\big(Y_1(x), Y_2(x)\big)$

$$I_i(x) =^{def} \; I\{Z(x) = i\} = \; I\{Y_1(x) \in \Delta t_i^1, Y_2(x) \in \Delta t_i^2\} \tag{4.2}$$

Corollary 2

The (cross-) covariance of the facies indicators coincides with the (cross-) covariance of the indicators of $\big(Y_1(x), Y_2(x)\big)$:

$$\begin{aligned} C_{ij}(x, x') =^{def} \; & cov(I\{Z(x) = i\}, I\{Z(x') = j\}) = \\ & cov\big(\, I\{Y_1(x) \in \Delta t_i^1, Y_2(x) \in \Delta t_i^2\}, I\big\{Y_1(x') \in \Delta t_j^1, Y_2(x') \in \Delta t_j^2\big\}\big) \end{aligned} \tag{4.3}$$

So, basically, the PGM on regular grids reduces the problem of facies simulation to a problem of simulating two multivariate Gaussian random fields $Y_1(x)$ and $Y_2(x)$ and applying a 2D truncation diagram defined by a partition of the plane $R^2 = \cup_{i=1\dots K} (\Delta t_i^1 \times \Delta t_i^2)$.

4.2 Problem statement for unstructured grids

For an unstructured grid $\{v_p, p = 1 \dots N_b\}$ the volumes of the blocks can vary significantly, and characterizing a given block v_p with a categorical variable "facies type" may be too simplistic. Indeed, given a large block, e.g. $1 \times 1km^2$, characterizing it as either shales or sands seems unrealistic. A better solution would be describing the compound of the block with proportions of the different facies contained in this block. This consideration leads to the problem statement of simulating proportions of facies on unstructured grids.

Definition 2
The proportion of facies i in block v is a continuous random variable defined by the equation

$$P_i(v) = \frac{1}{|v|}\int_v I\{Y_1(x) \in \Delta t_i^1, Y_2(x) \in \Delta t_i^2\}dx \tag{4.4}$$

Remark
Obviously, from the definition $P_i(v)$ satisfies the constraint $\Sigma_{i=1}^{K} P_i(v) = 1$.

Definition 2 can be interpreted in the following way: the proportion of facies i in block v is the frequency of occurrence of facies i in this block. Given the definition of $P_i(v)$, the problem of PGS on unstructured grids can be formulated. For given inputs:

- Unstructured grid $\{v_p,\ p = 1 \dots N_b\}$
- Covariance function $\rho_1(x, x')$ of $Y_1(x) \sim G(0,1)$
- Covariance function $\rho_2(x, x')$ of $Y_2(x) \sim G(0,1)$
- Truncation diagram $\{(\Delta t_i^1 \times \Delta t_i^2), i = 1 \dots K\}$, which links $Y_1(x)$ and $Y_2(x)$ to facies $Z(x)$ through formula (4.1).
- Conditioning data $\{(d_i, z_i), i = 1 \dots N_{data}\}$

Simulate a set of N_b random vectors $\left\{\left(P_1(v_p), \dots P_K(v_p)\right), p = 1 \dots N_b\right\}$, which respect the theoretical covariance and cross-covariance between proportions of different facies in blocks and the marginal distributions. The simulations should also respect the conditioning data.

Let us make clear what respecting the (cross-) covariance between proportions means. For any pair of blocks v_p and v_q and facies i and j in case of unconditional simulation, the covariance between simulated $P_i(v_p)$ and $P_j(v_q)$ should be

$$cov\left(P_i(v_p), P_j(v_q)\right) = C_{ij}(v_p, v_q), \tag{4.5}$$

where C_{ij} is defined in (4.3). The marginal distribution of $P_i(v_p)$ should coincide with the marginal distribution of $\frac{1}{|v|}\int_v I\{Y_1(x) \in \Delta t_i^1, Y_2(x) \in \Delta t_i^2\}dx$. For conditional simulation the same definition holds, but the facies indicators $I\{Y_1(x) \in \Delta t_i^1, Y_2(x) \in \Delta t_i^2\}$ should be considered conditional to the data.

The problem of simulating proportions of facies on unstructured grid is significantly more complicated than the simulation of a continuous variable, since for each block a random vector of proportions satisfying (4.5) should be simulated, and in addition a linear constraint is applied to simulation – the sum of facies proportions in every block is equal to 1.

A non-linear geostatistics solution based on DGM for this problem is studied below. Our results demonstrate that the DGM-based approach can be applied for an approximate reproduction of the block-to-block covariance, but the reproduction of marginal distributions is ensured only for a limited number of configurations of a truncation diagram $\{(\Delta t_i^1 \times \Delta t_i^2), i = 1 \dots K\}$.

4.3 PG-DGM: generalization of PGM for unstructured grids

Let us assume, that $P_i(v)$ - proportion of facies i in block v is a block-dependent transform of a block-dependent pair of standard Gaussian RVs Y_v^1 and Y_v^2, for which $Y_v^1 \perp Y_v^2$

$$P_i(v) = \psi_{v,i}(Y_v^1, Y_v^2) \tag{4.6}$$

Let, as previously, $\underline{x}$ denote a uniformly distributed random point within block v. The following assumptions are done for pairs $\left(Y_1(\underline{x}), Y_v^1\right)$ and $\left(Y_2(\underline{x}), Y_v^2\right)$

$$\begin{aligned} \left(Y_1(\underline{x}), Y_v^1\right) &\sim B(r_1) \\ \left(Y_2(\underline{x}), Y_v^2\right) &\sim B(r_2) \end{aligned} \tag{4.7}$$

-joint distributions are bivariate Gaussian with some correlation coefficients r_1 and r_2. Let us at the moment consider the correlation coefficients r_1 and r_2 to be known (the problem of determining these coefficients will be considered later). We demonstrate here that assumption (4.7) is sufficient to determine the block-dependent transformation function $\psi_{v,i}(y_1, y_2)$ in (4.6).

As stated in (4.2), the indicator of facies i can be expressed through indicators of $Y_1(x)$ and $Y_2(x)$. Let $\varphi_i\,(y_1, y_2)$ denote the bivariate indicator of two Gaussian random fields $Y_1(x)$ and $Y_2(x)$, i.e. $I\{Y_1(x) \in \Delta t_i^1, Y_2(x) \in \Delta t_i^2\} = \varphi_i\left(Y_1(x), Y_2(x)\right)$. This function can be decomposed in the basis of bivariate normalized Hermite polynomials $\left\{\chi_{n,m}(y1, y2), n = 0 \ldots \infty, m = 0 \ldots \infty\right\}$, see Appendix B for details:

$$\begin{aligned} I_i(x) = \; & I\{Y_1(x) \in \Delta t_i^1, Y_2(x) \in \Delta t_i^2\} = \varphi_i\left(Y_1(x), Y_2(x)\right) \\ & = \Sigma_{n=0}^{+\infty}\Sigma_{m=0}^{n}\varphi(i,n,m)\chi_{n,m}\left(Y_1(x), Y_2(x)\right), \end{aligned} \tag{4.8}$$

where $\varphi(i,n,m)$ are the coefficients of decomposition of $\varphi_i\,(y_1, y_2)$. The block transformation function $\psi_{v,i}(y_1, y_2)$ can also be expressed in the same basis of normalized bivariate Hermite polynomials:

$$\psi_{v,i}(y_1, y_2) \; = \Sigma_{n=0}^{+\infty}\Sigma_{m=0}^{n}\psi(v,i,n,m)\chi_{n,m}(y_1, y_2) \tag{4.9}$$

Applying Cartier's relation (Chilès & Delfiner 2012)

$$E\left[\varphi_i\left(Y_1(\underline{x}), Y_2(\underline{x})\right)\middle| P_i(v)\right] = P_i(v), \tag{4.10}$$

leads to an explicit expression for $\psi_{v,i}\,(y_1, y_2)$ through $\varphi_i(y_1, y_2)$ in the basis of bivariate Hermite polynomials. The coefficients of the decomposition of $\psi_{v,i}(y_1, y_2)$ before the basis element $\chi_{n,m}$ are

$$\psi(v,i,n,m) = \; \varphi(i,n,m) r_1^{n-m} r_2^{m}. \tag{4.11}$$

Complete derivation of (4.11) is given in Appendix B. To conclude, given the decomposition of the bivariate Gaussian indicator $\varphi_i(y_1, y_2)$ decomposed in the basis of bivariate normalized Hermite polynomials $\{\chi_{n,m}(y_1, y_2), n = 0 \dots \infty, m = 0 \dots \infty\}$ and correlation coefficients (r_1, r_2), one is able to derive the block transformation function $\psi_{v,i}(y_1, y_2)$ for any proportion of facies $i = 1 \dots k$.

In practice it is not possible to use the infinite number of Hermite polynomials to decompose the indicator function – the decomposition should be truncated on some finite number of polynomials N_{basis}. A comparison of indicator decompositions for various values of N_{basis} can be found in Chilès and Delfiner (2012). Fortunately, as demonstrated in Appendix B.4, modeling the transformation function $\psi(v, i, n, m)$ can be done in an analytical manner, avoiding the Hermite polynomials.

It should be noted that the (cross-) covariance between two facies indicators $I_i(x)$ and $I_j(x')$ can also be decomposed into a series:

$$\begin{aligned} C_{ij}(x, x') &= cov\left(I\{Y_1(x) \in \Delta t_i^1, Y_2(x) \in \Delta t_i^2\}, I\{Y_1(x') \in \Delta t_j^1, Y_2(x') \in \Delta t_j^2\}\right) \\ &= \Sigma_{n=1}^{+\infty} \Sigma_{m=0}^{n-1} \varphi(i, n, m) \varphi(j, n, m) \rho_1^{n-m}(x, x') \rho_2^{m}(x, x') \end{aligned} \quad (4.12)$$

Proposition 1
This method of modeling the proportions guarantees that the sum of proportions in every block is 1.

The proof of this proposition is given in Appendix B.

4.4 Derivation of correlation coefficients

As in the case of classical DGM we refer to (r_1, r_2) as "change of support coefficients". Similar to the use of DGM 1 and DGM 2 for simulations of continuous variables, there exist two ways for determining the change of support coefficients (r_1, r_2) for a block. Let us start from the simpler approach of adding an additional assumption in the theoretical model.

Adding assumption (4.13), leads to a simple formula for determining the change of support coefficients r_1 and r_2. Let $\underline{x}$ and $\underline{x'}$ be two independent uniformly distributed in block v random points. Then assume that the following distributions are bivariate Gaussian

$$\begin{aligned} \left(Y_1(\underline{x}), Y_1(\underline{x'})\right) &\sim B \\ \left(Y_2(\underline{x}), Y_2(\underline{x'})\right) &\sim B \end{aligned} \quad (4.13)$$

As in case of DGM 2, the bivariate distributions of the randomized points in the studied domain are approximated with bivariate Gaussian distribution. In this case, exactly as in DGM 2, $Y_v^1 = \frac{1}{r_1}\frac{1}{|v|}\int_v Y_1(x)dx = \frac{1}{r_1}Y_1(v)$ and $Y_v^2 = \frac{1}{r_2}\frac{1}{|v|}\int_v Y_2(x)dx =$

$\frac{1}{r_2}Y_2(v)$ which leads directly to formula (4.14) for the change of support coefficients. The proof can be found in (Chilès & Delfiner 2012; Emery 2007). Let $\overline{the\ bar}$ on top of the function denote a double volumetric integral over the given volumes normalized by the multiplication of these volumes.

$$\begin{cases} r_1^2 = \overline{\rho_1}(v,v) \\ r_2^2 = \overline{\rho_2}(v,v) \end{cases}. \tag{4.14}$$

In a similar manner, for two blocks v_p and v_q with change of support coefficients $(r_{1,p}, r_{2,p})$ and $(r_{1,q}, r_{2,q})$, the covariance between pairs of variables $\left(Y^1_{v_p}, Y^1_{v_q}\right)$ and $\left(Y^2_{v_p}, Y^2_{v_q}\right)$ can be found:

$$\begin{cases} r_{1,p}r_{1,q}cov\left(Y^1_{v_p}, Y^1_{v_q}\right) = \overline{\rho_1}(v_p, v_q) \\ r_{2,p}r_{2,q}cov\left(Y^2_{v_p}, Y^2_{v_q}\right) = \overline{\rho_2}(v_p, v_q) \end{cases}, \tag{4.15}$$

the derivation of which is straightforward.

As in the case of DGM2, introducing an additional assumption (4.13) into the model provides a significant simplification for deriving the change of support coefficients. The disadvantage of this approach is that the derivation of (r_1, r_2) does not ensure the correct reproduction of $Var\left(P_i(v)\right)$. However, it is possible to derive the change of support coefficients without assumption (4.13) in a manner similar to DGM1 in order to ensure the reproduction of $Var\left(P_i(v)\right)$. For that one should notice that for block v the variance of the proportion of facies i in this block can be expressed in two forms: through $C_{ii}(x, x')$ - the covariance function of $I_i(x)$ and through the change of support coefficients (r_1, r_2). Indeed:

$$Var\left(P_i(v)\right) =^{def} \bar{C}_{ii}(v,v) = \Sigma_{n=1}^{+\infty}\Sigma_{m=0}^{n}\varphi^2(i,n,m)r_1^{2(n-m)}r_2^{2m}. \tag{4.16}$$

Equation (4.16) should hold for any facies $i = 1 \dots K$. Expressed in this way, finding the change of support coefficients (r_1, r_2) is equivalent to solving an optimization problem for parameters (r_1, r_2):

$$\sum_{k=1}^{K}\left[\bar{C}_{ii}(v,v) - \Sigma_{n=1}^{+\infty}\Sigma_{m=0}^{n}\varphi^2(i,n,m)r_1^{2(n-m)}r_2^{2m}\right]^2 \to min. \tag{4.17}$$

This approach is examined in details in Appendix B. As explained in Appendix B, equations (4.14) and (4.15) can be also derived with this approach as approximate solutions of the optimization problem (4.17).

The proposed theoretical model enables reducing the problem of simulating N_b random vectors $\left\{\left(P_1(v_p), \dots P_K(v_p)\right), p = 1 \dots N\right\}$ with given covariance and cross-covariance to a problem of simulating two appropriate multivariate discrete Gaussian

random vectors $\{Y^1_{v_p}, p = 1 \dots N_b\}$ and $\{Y^2_{v_p}, p = 1 \dots N_b\}$ and applying block-dependent transformation functions. The applicability of the proposed model should generally be tested for given inputs. The tests provided in the following section demonstrate that PG-DGM gives a good approximation for block to block covariance, however, in the general case the obtained marginal distributions differ from those provided by using PGM on a fine scale and upscaling.

4.5 Conditioning of PG-DGM simulations

Conditioning of PG-DGM simulations can be performed with kriging. For that, it is sufficient to condition the simulated unconditional multivariate Gaussian random vectors $\{Y^1_{v_p}, p = 1 \dots N_b\}$ and $\{Y^2_{v_p}, p = 1 \dots N_b\}$ to the given input point-support data $\{Y_1(d_\alpha) = y_1(d_\alpha), Y_2(d_\alpha) = y_2(d_\alpha), \alpha = 1 \dots N_{data}\}$. It should be noted, that the transformed to Gaussian scale conditioning data should be obtained from the discrete data $(z_i, i = 1 \dots N_{data})$ using the Gibbs sampler, see Lantuejoul (2002) for details.

For that, for every conditioning data location d_α, we derive the point support samples coherent with the unconditional block simulation using (4.7). Let $d_\alpha \in v_p$, then

$$\begin{aligned} Y_1\left(\underline{d_\alpha}\right) &= r_{1,p} Y^1_{v_p} + \sqrt{1 - r^2_{1,p}}\,\xi^1_\alpha,\ where\ \xi^1_\alpha \sim N(0,1) \\ Y_2\left(\underline{d_\alpha}\right) &= r_{2,p} Y^2_{v_p} + \sqrt{1 - r^2_{2,p}}\,\xi^2_\alpha,\ where\ \xi^2_\alpha \sim N(0,1). \end{aligned} \tag{4.18}$$

If we identify, for the purpose of conditioning, $\underline{d_\alpha}$ and d_α, then the covariance between $\{Y_1(d_\alpha), \alpha = 1 \dots N_p\}$ if defined by function $\rho_1(h)$. The covariance between $Y^1_{v_q}$ and $Y_1(d_\alpha)$ can be easily found assuming $Y^1_{v_q} = \frac{Y_1(v_q)}{r_{1,q}}$

$$\begin{cases} r_{1,q}\, cov\left(Y_1(d_\alpha), Y^1_{v_q}\right) = \overline{\rho_1}(d_\alpha, v_q) \\ r_{2,q}\, cov\left(Y_2(d_\alpha), Y^2_{v_q}\right) = \overline{\rho_2}(d_\alpha, v_q) \end{cases} \tag{4.19}$$

It should be noted that the proposed approach for conditioning does not require any modifications if several data points belong to the same block of the grid. In this case the conditional independence hypothesis is applied – given the block value the sample values are independent and can be derived through (4.18). After all the sample to sample and sample to block covariances are derived, RF $Y_1(x)$ and $Y_2(x)$ can be independently conditioned to data $\{y_1(d_\alpha), \alpha = 1 \dots N_{data}\}$ and $\{y_2(d_\alpha), \alpha = 1 \dots N_{data}\}$ respectively through conditioning kriging (de Fouquet 1994). Applying transformation (4.6) leads to a conditioned simulation. As in the case of DGM 1 and DGM 2, it is possible to avoid the assumption that $\underline{d_\alpha}$ and d_α are

identical. In this case the input samples should be considered as randomized inside the blocks from the beginning. The resulting covariances in this case are derived in the same way as for DGM 1 and DGM 2.

4.6 Comparing with mini-models

Let us compare the marginal distributions of proportions implied by PG-DGM with the observed distributions produced by Monte Carlo mini-model simulation on regular grids followed by upscaling for a single 3D block of an unstructured grid. One categorical variable “facies” is simulated on the blocks of the mini-model. This can be considered as simulating the proportion vectors for which one of the proportions is equal to 1 and the others are 0. The block selected for the test is a Voronoï polygon-based prism which is depicted on the Figure 4-2 inscribed in its bounding box. The bounding box dimensions are $88 \times 97 \times 4$ m.

For the purpose of convenience instead of defining the truncation diagram on the plane $R^2 = \cup_{i=1\dots K} (\Delta t_i^1 \times \Delta t_i^2)$ it is common to define the truncation diagram on $[0,1]^2$, so that $[0,1]^2 = \cup_{i=1\dots K} (\Delta s_i^1 \times \Delta s_i^2)$. The advantage of this definition is that $EI_i(x)$ is equal to the area of rectangle $\Delta s_i^1 \times \Delta s_i^2$. The conversion between two representations is straightforward. Two truncation diagrams defined on $[0,1]^2$ demonstrated on Figure 4-1 were selected for the test.

A spherical covariance function with geometric anisotropy is used for the marginal distribution reproduction test. For all tests $\rho_1(x,x') = \rho_2(x,x') = \rho(x,x')$. Three sets of parameters are tested

1) Short range of $\rho(x,x')$ is $(50,50,1)m$
2) Medium range of $\rho(x,x')$ is $(100,100,2)m$
3) Long range of $\rho(x,x')$ is $(150,150,3)m$

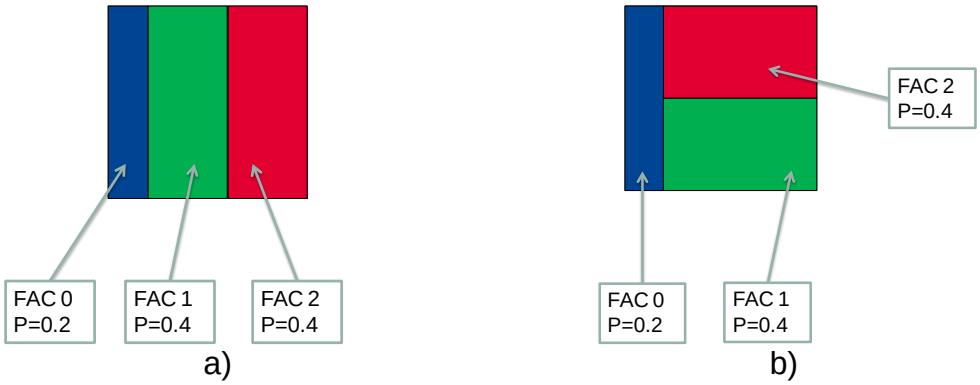
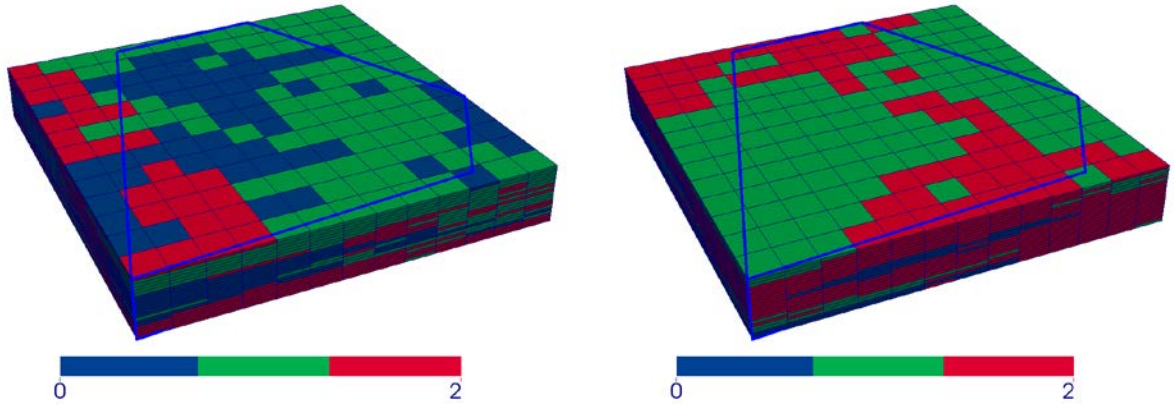


Figure 4-1. Truncation diagrams a) diagram 1 b) diagram 2.

The marginal distribution of $P_i(v), i = 0,1,2$ turns out to be very sensitive to the simulation method – we were not able to obtain decent quality of statistics reproduction with SGS applied for mini-models. For the purpose of this test the mini-model simulation is performed with the Cholesky decomposition simulation method. Figure 4-2 demonstrates two realizations of mini-model for truncation diagram 2 for short and medium ranges of $\rho(x,x')$.



a) b)

Figure 4-2. Mini-model simulation for a single Voronoï polygon block (diagram 2). The boundary of the block is shown with blue line. Vertical zoom factor 2. a) short range b) medium range.

The comparison of the marginal distribution of $P_i(v), i = 0,1,2$ implied by PG-DGM with the value observed from Monte Carlo simulations for a small covariance function range is shown on Figure 4-3 and Figure 4-4; the results for medium and large ranges can be found in Appendix B. It is observed, that for truncation diagram 1, the distributions provided by PG-DGM differ significantly from the distributions obtained with Monte Carlo for facies 1, which is bounded by facies 0 on the left and facies 2 on the right on the truncation diagram. On the other hand, for truncation diagram 2, Figure 4-4 demonstrates a fairly good reproduction of the marginal distribution. Our tests indicate that the marginal distribution of $P_i(v)$ is not reproduced if on the truncation diagram facies i was bounded by other facies on both sides. This effect limits significantly the range of applicability of PG-DGM. On the other hand, for truncation diagrams as on Figure 4-1b, where no facies is bounded from both sides, PG-DGM gives good approximations for marginal distributions.

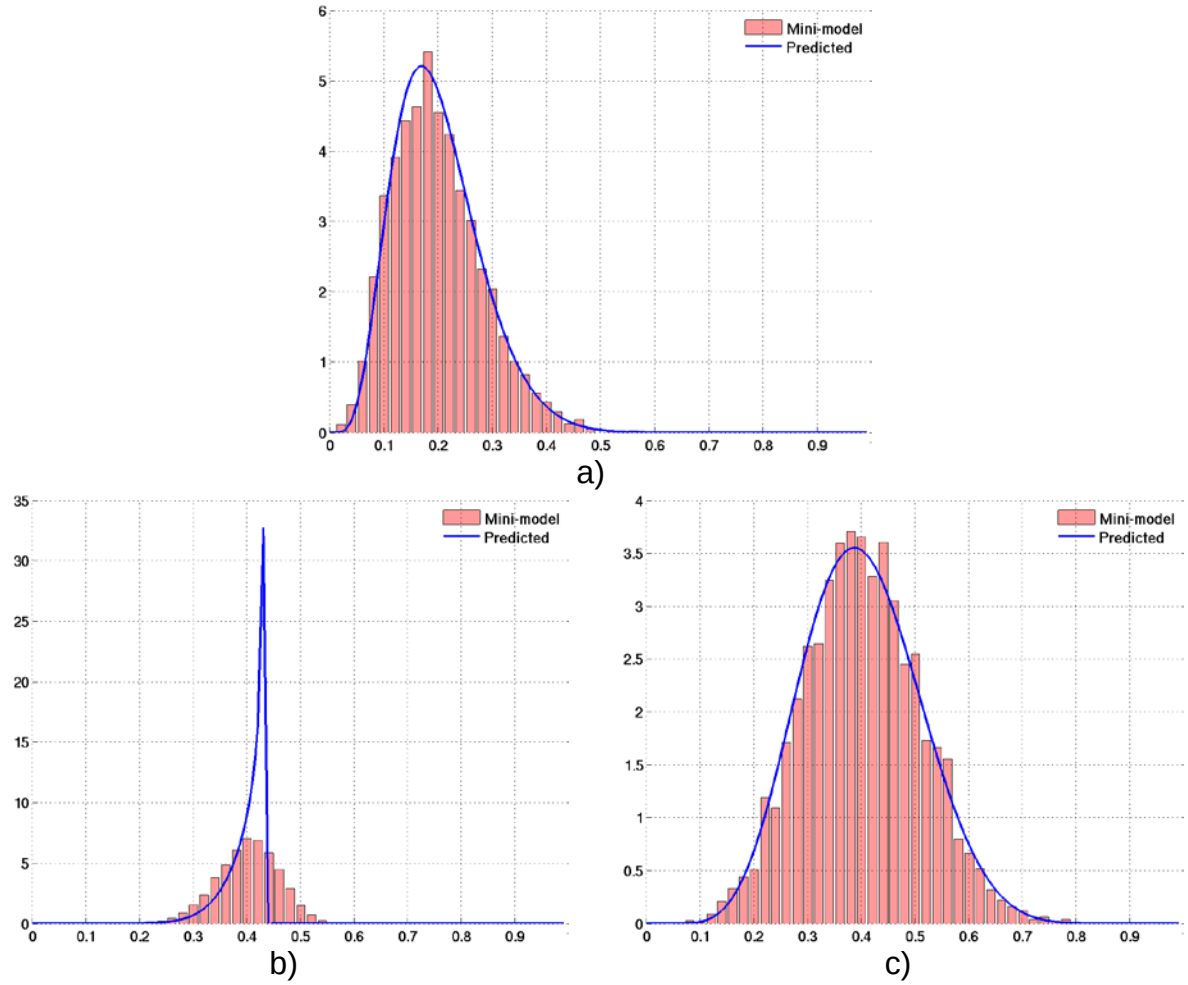


Figure 4-3. Block marginal distribution for truncation diagram 1 (small covariance range case). a) $P_0(v)$ b) $P_1(v)$ c) $P_2(v)$. Histogram demonstrates the observed distribution from Monte Carlo simulations, blue curve indicates the predicted by PG-DGM density.

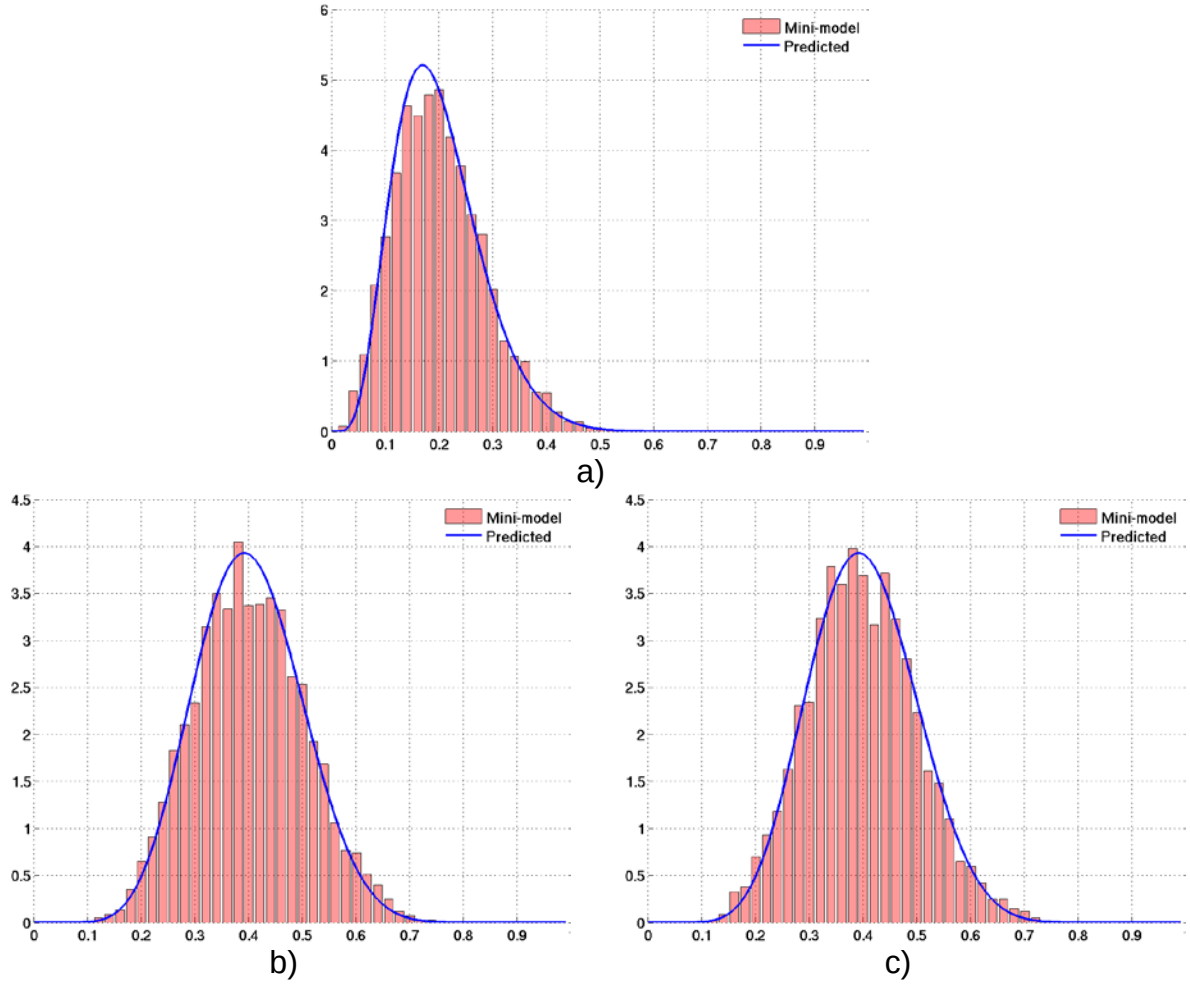


Figure 4-4. Block marginal distribution for truncation diagram 2 (small covariance range case). a) $P_0(v)$ b) $P_1(v)$ c) $P_2(v)$. Histogram demonstrates the observed distribution from Monte Carlo simulations, blue curve indicates the predicted by PG-DGM density.

4.7 Unconditional simulation with PG-DGM

Since the marginal distributions for truncation diagram 1 are not reproduced, only truncation diagram 2 is considered further. For truncation diagram 2 unconditional realizations produced with PG-DGM for a 1-layer 3D grid are demonstrated on Figure 4-5 and Figure 4-6. For Figure 4-5 a spherical covariance with range $(300,300,1)m$ is used, while for Figure 4-6 a spherical covariance with range $(800,400,1)m$ and azimuth 45 degrees is used.

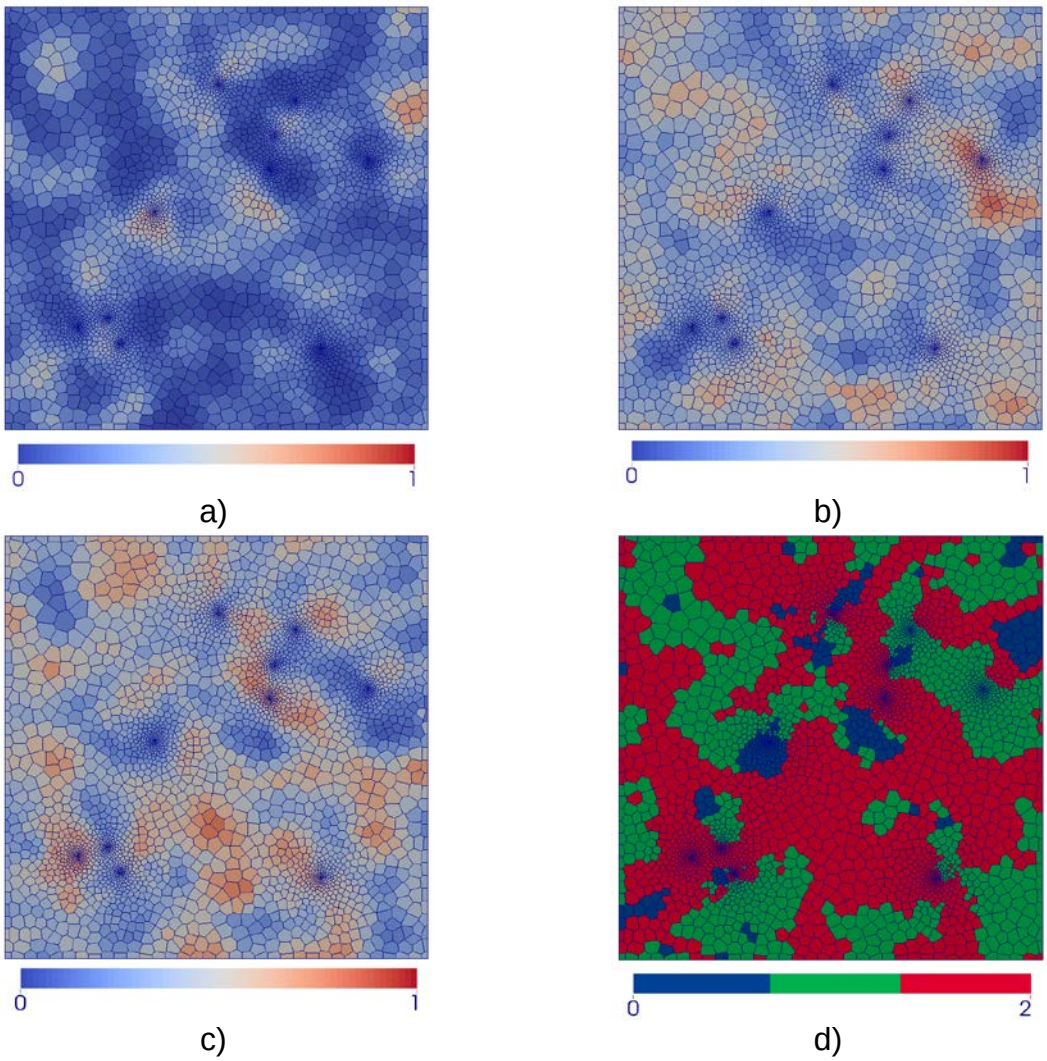


Figure 4-5. Unconditional realizations produced with PG-DGM. a) $P_0(v)$ b) $P_1(v)$ c) $P_2\ (v)$ d) dominant facies map.

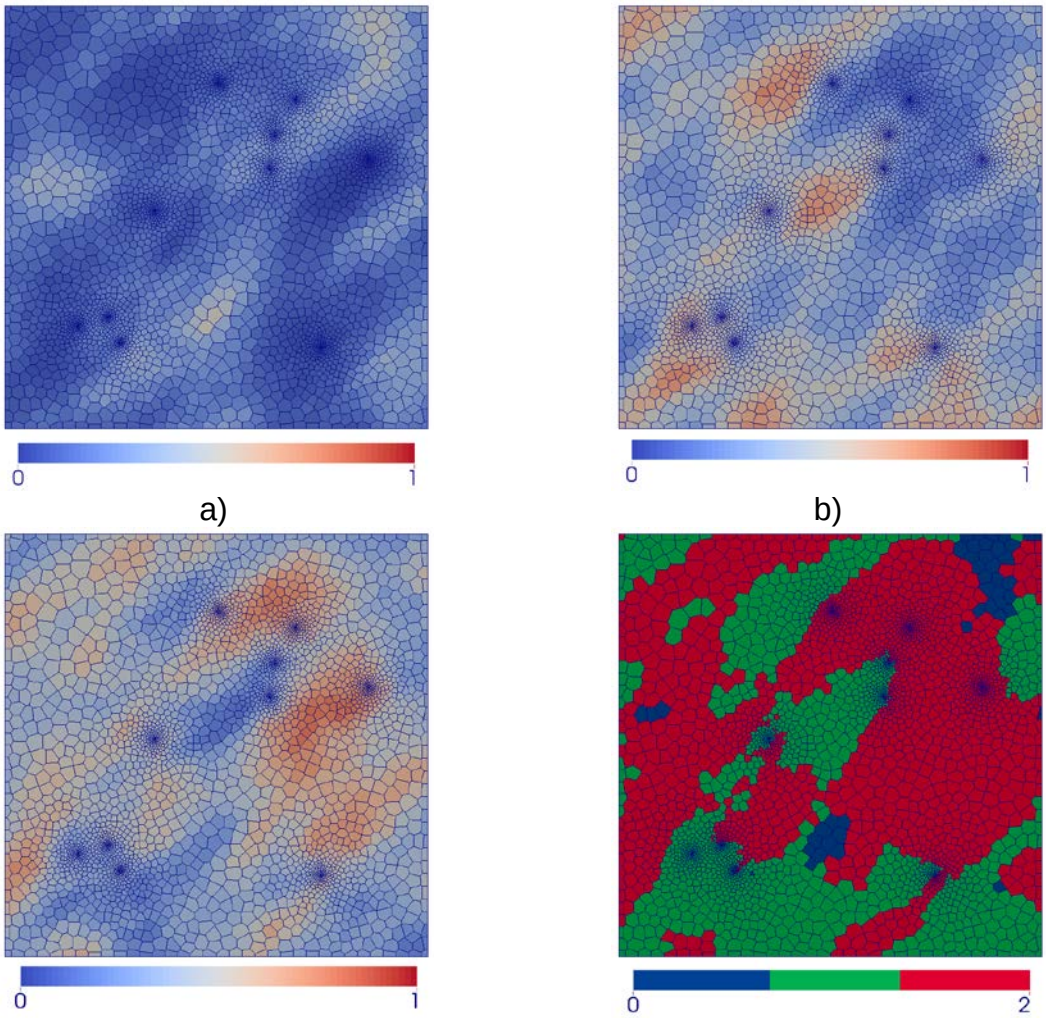


Figure 4-6. Unconditional realizations produced with PG-DGM. a) $P_0(v)$ b) $P_1(v)$ c) $P_2\ (v)$ d) dominant facies map.

We verify if PG-DGM reproduces correctly the block to block covariance. A test for spherical covariance with range $(300,300,1)m$ is provided below. Given the indicator covariance function $C_{00}(x,x')$ for facies 0, the covariance $C_{00}(v_p,v_q)$ between any pair of blocks (v_p,v_q) can be computed. It is the theoretical value of block to block covariance which should be respected by a facies proportion simulation algorithm on unstructured grids, as indicated in (4.5). When PG-DGM is applied, as mentioned in the model derivation section, the values of the block to block covariance are reproduced only approximately. The $cov\left(P_0(v_p),P_0(v_q)\right)$ implied by PG-DGM does not coincide with the theoretical value $C_{00}(v_p,v_q)$, see Eqn. (B.31) in Appendix B. Figure 4-7 gives a cross-plot of implied by PG-DGM covariance $C_{00}(v_p,v_q)$ versus the theoretical values.

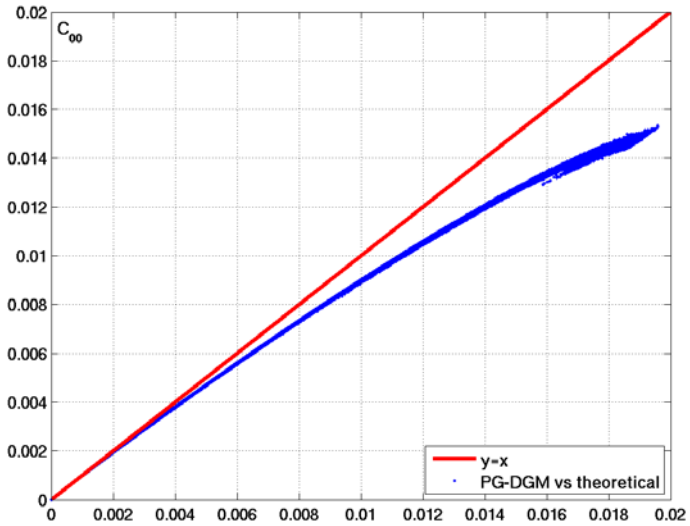


Figure 4-7. PG-DGM block to block covariance versus theoretical.

It is seen from Figure 4-7 that PG-DGM does not respect exactly the block to block covariance, but only approximates it with visible systematic underestimation. The error of this approximation relative to point support variance $\sigma_{00}^2 = C_{00}(0,0)$ is $\delta_c = \frac{C_{00}(v_p,v_q) - C_{00}^{DGM}(v_p,v_q)}{\sigma_{00}^2} * 100\% \leq 3\%$, while the classical relative error $\delta_r = \frac{C_{00}(v_p,v_q) - C_{00}^{DGM}(v_p,v_q)}{C_{00}(v_p,v_q)} * 100\%$ is high and reaches 20% (mean δ_r is 5.5%). Although the relative block to block covariance is high, the general structure of the block to block covariance matrix is reproduced. Analyzing these results, one should take into account that the change of support effect in this test is very high – the maximal change of support coefficients for a block are only $r_1 = r_2 = 0.42$. As demonstrated in the mini-model tests, even for stronger change of support effect PG-DGM provides a good approximation for the marginal distributions. Although the block to block covariance implied by PG-DGM underestimates the theoretical covariance $C_{00}(v_p, v_q)$, it is questionable whether in practice any other simulation method based on a sequential simulation principle is able to produce better results for the same inputs. Application of SGS on a fine regular grid followed by upscaling for these inputs did not lead to any satisfactory results in reproduction of marginal distributions and covariance even if large simulation neighborhood of 400 previously simulated nodes was considered. Moreover, as mentioned before, SGS followed by upscaling does not reproduce the marginal distribution even for mini-model Monte Carlo simulations, which explains the use of Cholesky decomposition.

4.8 Conclusion

The problem of simulating discrete variables such as facies was generalized in order to take into account the change of support effect. As demonstrated in this chapter, when the change of support effect is considered, proportions of facies should be simulated. The problem of simulating proportions of facies on an unstructured grid proves to be significantly more complicated than the problem of simulating without change of support and than the problem of simulating continuous variables on unstructured grids. The above-mentioned complexity arises from the necessity of respecting the cross-covariance relations between the blocks as well as ensuring that the sum of proportions in every block equals to 1.

A generalization of the PGS algorithm based on the DGM formalism was presented. Author is not aware of any previous attempts to generalize the PGS for simulations on unstructured grids with the volume support effect. Previous generalizations of PGS include the work of Emery and Gálvez (2012), where the authors propose a generalization of PGM for simulations of regionalized compositions. Both PG-DGM and the model proposed by Emery and Gálvez (2012) permit simulating a set of random vectors with correlated components under the constraints of non-negativity and fixed sum of the components of each vector. The models have significant theoretical differences, applications of the algorithm proposed by Emery and Gálvez (2012) to the simulations on unstructured grids were not investigated. PG-DGM has a limited scope of application – in particular, the marginal distributions are not reproduced correctly for TGS-style truncation diagrams. On the other hand, as demonstrated through the mini-model Monte Carlo tests, for the “open” flags the PG-DGM reproduced the marginal distributions of proportions.

When PG-DGM is not applicable to the facies simulation on an unstructured grid, another simulation algorithm should be applied. One solution is using fine scale simulations on a regular grid followed by upscaling on the target unstructured grid. This approach has the disadvantage of using very fine regular grids. A more elegant constructive solution to this problem was proposed by Gross and Boucher (2015) which is a generalization of the previously presented algorithm for regular grids (Boucher & Dimitrakopoulos 2009). The authors present a sequential simulation algorithm on unstructured grids, which can be applied for facies simulation, in particular to PGM. The algorithm simulates block after block and uses upscaling inside each block instead of analytical transform functions such as PG-DGM. Simulations inside each block should be performed on a set of discretizing points in the Gaussian scale, conditional to hard data and to the previously simulated block values. In the next chapter we contribute the results of (Gross & Boucher 2015) by demonstrating that a set of Sobol’ quasi-random points is a good choice for discretizing each block in this algorithm. Through the algorithm of Gross and Boucher (2015) is able to reproduce the marginal distributions of facies proportions, a disadvantage of this algorithm is that it is linked to the sequential simulation principle

and is subject to the weakness of all sequential simulation algorithms – necessity of using a limited neighborhood for simulations, which can have a negative impact on resulting statistics. As demonstrated in Chapter 3 in the testing DGM section, it is not always possible to reproduce rigorously statistics with sequential simulation methods.

Chapter 5: Covariance computations

Résumé

Le calcul de covariance entre les blocs est une des parties principales de tous les algorithmes de simulation pour les maillages non-structurés. Ce chapitre compare les différentes approches à ce problème – l'utilisation de discrétisations régulières, les quadratures Gaussiennes et les méthodes de Monte Carlo. L'avantage des méthodes de Monte Carlo, notamment l'utilisation des suites à discrépance faible, sur les méthodes déterministes est illustré.

5.1 Introduction

One of the crucial problems when performing geostatistics on unstructured grids is the problem of computing the block to block covariance.

$$C(v_p, v_q) = \frac{1}{|v_p||v_q|}\int_{v_p}\int_{v_q} C(x, x')dxdx', \tag{5.1}$$

The computation of the block to block covariance is essential for determining the change of support coefficients for the blocks as well as the covariance between the Gaussian variables Y_v corresponding to the block average values $Z(v_p)$. When the blocks on the unstructured grid are defined in 3 dimensions, which is often the case in practice, the above-mentioned double volumetric integral leads to a computation of a sextuple integral, and its computation becomes indeed a challenging task.

According to Press et al. (1996), computation of multidimensional integrals is "not easy", and in the general case, facing the problem of such a computation, one should try to avoid it when possible by reducing the dimension of the problem. Computing the integral (5.1) is further complexified by the shape of the boundary of the blocks v_p and v_q. Indeed, when the problem of computing $C(v_p, v_q)$ is considered in one dimension, the blocks are defined simply as closed intervals: $v_p = [l_p, r_p]$, $v_q = [l_q, r_q]$ and the computation of (5.1) reduces to

$$C(v_p, v_q) = \frac{1}{(r_p - l_p)(r_q - l_q)}\int_{l_p}^{r_p}\int_{l_q}^{r_q} C(x, x')dxdx', \tag{5.2}$$

which can be solved with any classical method of numerical integration as trapezoidal quadrature, Simpson's rule, Gaussian quadrature or other, see Press et al. (1996) for an extensive methods review. When the problem is considered is two dimensions, the boundary of the blocks v_p and v_q cannot be represented in a simple form like in (5.2). In order to compute $C(v_p, v_q)$, the cross-sections of v_p should be represented

in a functional form, for instance, one might derive a representation for a cross-section of v_p with a fixed x coordinate: $v_p(x) = [l_p(x), r_p(x)]$, see Figure 5-1.

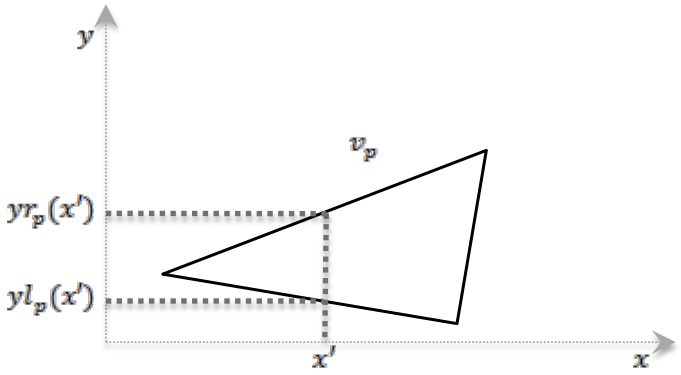


Figure 5-1. Functional representation of the block cross-sections in 2D.

Given that the x boundaries for v_p and v_q are (xl_p, xr_p) and (xl_q, xr_q) respectively, the integral (5.1) can be written in the following form:

$$C(v_p, v_q) = \frac{1}{|v_p||v_q|} \int_{xl_p}^{xr_p} \int_{yl_{p(x)}}^{yr_p(x)} \int_{xl_q}^{xr_q} \int_{yl_q(x\prime)}^{yr_q(x\prime)} C[(x,y),(x',y')]dxdydx'dy' \qquad (5.3)$$

The volume $|v_p|$ in formula (5.3) should also be computed with integration, but we avoid writing this explicitly for the sake of simplicity. In fact finding the cross-section functions $yl_p(x)$ and $yr_p(x)$ is also a complicated task, which hardly has a general solution for arbitrary shaped blocks. This task can be simplified when the integration domain has some “good” properties, for instance if it is convex, or even better, if it is a convex polygon in two dimensions or a convex polyhedron in three dimensions. In that case the cross-section functions can be efficiently found. Following the logic of the 2D case, the integration formula in 3D has the following form:

$$C(v_p, v_q) =$$

$$\frac{1}{|v_p||v_q|} \int_{xl_p}^{xr_p} \int_{yl_{p(x)}}^{yr_p(x)} \int_{zl_{p(x,y)}}^{zr_p(x,y)} \int_{xl_q}^{xr_q} \int_{yl_q(x\prime)}^{yr_q(x\prime)} \int_{zl_q(x',y\prime)}^{zr_q(x\prime,y\prime)} C[(x,y,x),(x',y',z')]dxdydzdx'dy'dz'. \qquad (5.4)$$

Due to the necessity of finding the boundary functions $l_p(x)$, $yr_p(x), zl_p(x,y)$ and $zr_p(x,y)$, we make an assumption that our cells are all convex polyhedrons in 3D. This assumption is very reasonable, since the flow simulation grids in the petroleum industry are often composed of convex polyhedrons, such as tetrahedron, triangular prisms, Voronoï polygon – based prisms, cubes and octahedrons. But even in the case of convex polyhedrons in 3D, the problem of finding the boundary

functions remains rather complicated. The method of modeling convex polygons in 3D is described in the following section.

5.2 Modeling grid blocks

The advantage of using the convex polyhedrons in 3D (and convex polytopes in general case) is that they can be represented as a system of linear inequalities - the intersection of a final number of half – spaces (Grünbaum & Shephard 1969). For any convex polytope v one can write

$$v = \begin{cases} a_{11}x + a_{12}y + a_{13}z \le b_1 \\ \vdots \\ a_{m1}x + a_{m2}y + a_{m3}z \le b_m \end{cases}, \tag{5.5}$$

where m is the number of faces of the block v. Although the representation (5.5) can be easily derived from the set of faces of block v and the directions of the outward normals to these faces, this representation itself is not sufficient to derive the cross-section functions for v, but it can be further converted to a form compatible with Eqn. (5.4). When the block v is a convex polyhedron, the boundary functions are piecewise-linear, and any integral of a continuous function over the block v can be represented as a sum of integrals with linear boundaries. For instance, for the volume of block v, one has

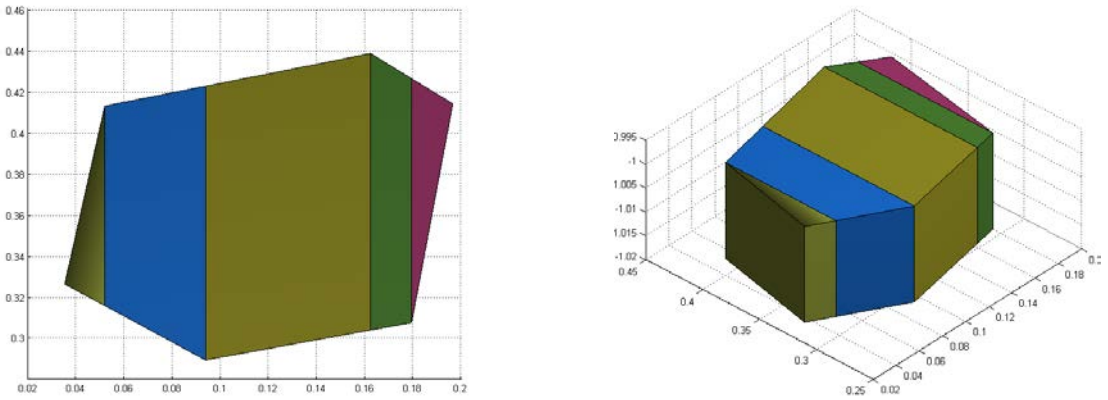
$$|v| = \sum_{i=1}^{NL(v)} \int_{a_x^i}^{b_x^i} \int_{a_y^i + a_{yx}^i x}^{b_y^i + b_{yx}^i x} \int_{a_z^i + a_{zx}^i x + a_{zy}^i y}^{b_z^i + b_{zx}^i x + b_{zy}^i y} 1 dx dy dz, \tag{5.6}$$

where $NL(v)$ is the number of regions in v with linear boundaries. Transforming the system (5.5) to the form suitable for integration is not a simple task, but it can efficiently be solved with the algorithm proposed by Korenblit and Shmerling (2006). The algorithm transforms the system (5.5) into the following form

$$v = \begin{bmatrix} \begin{cases} a_x^1 \le x \le b_x^1 \\ a_y^1 + a_{yx}^1 x \le y \le b_y^1 + b_{yx}^1 x \\ a_z^1 + a_{zx}^1 x + a_{zy}^1 y \le z \le b_z^1 + b_{zx}^1 x + b_{zy}^1 y \end{cases} \\ \vdots \\ \begin{cases} a_x^{NL(v)} \le x \le b_x^{NL(v)} \\ a_y^{NL(v)} + a_{yx}^{NL(v)} x \le y \le b_y^{NL(v)} + b_{yx}^{NL(v)} x \\ a_z^{NL(v)} + a_{zx}^{NL(v)} x + a_{zy}^{NL(v)} y \le z \le b_z^{NL(v)} + b_{zx}^{NL(v)} x + b_{zy}^{NL(v)} y \end{cases} \end{bmatrix}, \tag{5.7}$$

which gives explicit formulas for the boundary functions and enables computing integrals like (5.4) and (5.6) over the blocks. It should be noted, that the algorithm of Korenblit and Shmerling (2006) also guarantees that the linear systems of equations in (5.7) are mutually exclusive, which makes the summation in formula (5.6) correct. The system of equations resulting from the decomposition algorithm can be

effectively stored and used for computations. Figure 5-2 illustrates the result of decomposing a Voronoï polygon-based prism into representation (5.7).



a) b)

Figure 5-2. Decomposition of a convex polyhedron into blocks with linear boundaries. a) top view b) side view.

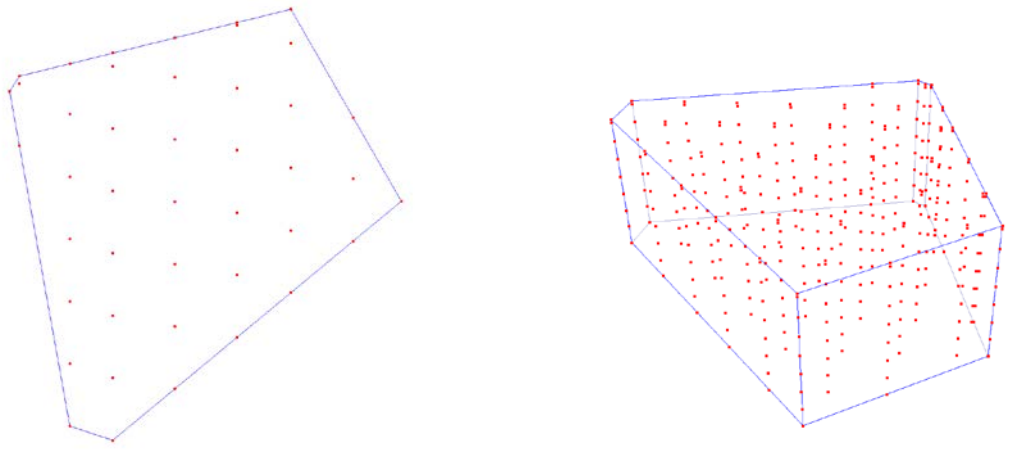
Figure 5-2 demonstrates that the polyhedron is decomposed into a number of non-overlapping SOI (system of inequalities), which determine convex polyhedrons with linear (and not piecewise-linear, as for the original block) boundaries. This decomposition enables one to derive the boundary functions for equation (5.4). Given these boundary functions, various methods can be considered for numerical integration in order to compute the block to block covariance. We further extensively study the problem of computing the variance of a block – its covariance with itself, since this problem is more complicated from the numerical point of view. Indeed, computing the variance of a single block includes summation over both large and small values of covariance function, whereas computing the covariance between two distinct blocks in general only refers to the values corresponding to non-zero separation vectors.

5.3 Computing with regular discretization

As suggested by Goovaerts (1997) for regular grids, in order to compute the block to block covariance, each block should be discretized with some finite number N of points and the block to block covariance should be computed as an arithmetic mean of the point to point covariances. Then for block v one has:

$$Var(v) = C(v,v) \approx \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} C(x_i, x_j) \tag{5.8}$$

In case of regular grids, the blocks are rectangular parallelepipeds and the discretization can be simply derived using regular spacing on the sides of the parallelepiped. On the other hand, creating a discretization for a block of a general form is less obvious. Representation (5.7) enables one to build the same type of discretization as for rectangular parallelepipeds for convex polyhedrons. This type of discretization can be intuitively considered as building a fine scale regular grid that discretizes block v. The discretization is built for each SOI composing the block independently, so the points are aligned only inside each SOI with linear boundaries, in addition, we always include the points on the sides of the polyhedron, in order to keep the summation formula (5.8) closed (the same open-closed notations as for quadratures is used). Figure 5-3 illustrates a regular discretization built for a Voronoï polygon-base block v in 3D.



a) b)

Figure 5-3. Regular discretization of a 3D Voronoï polygon-based block. The points are placed regularly within each SOI, boundary points are always included. a) top view b) side view.

One of the advantages of using a regular discretization is that in order to compute $Var(v)$ it is not necessary to compute independently the integral part of (5.1) and the volume. Indeed, let v be discretized into N equal blocks σ_i, identical up to a translation to a small block σ. Then

$$Var(v) = \frac{1}{|v|^2}\int_v\int_v C(x,x')dxdx'$$

$$= \frac{1}{\left(\lim\limits_{|\sigma|\to 0}\sum_{i=1}^{N}|\sigma_i|\right)^2}\lim_{|\sigma|\to 0}\sum_{i=1}^{N}\sum_{j=1}^{N}C(x_i,x_j)|\sigma_i||\sigma_j| \qquad (5.9)$$

$$= \lim_{|\sigma|\to 0}\frac{\sum_{i=1}^{N}\sum_{j=1}^{N}C(x_i,x_j)|\sigma|^2}{N^2|\sigma|^2} = \lim_{N\to+\infty}\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}C(x_i,x_j).$$

5.4 Computing with Gaussian quadratures

With regards to the representation of the convex polyhedron block v in the form (5.7), multidimensional integrals over v can be computed by applying successive one-dimensional integrations. Let us consider the problem of computing the volume of block v, for which $NL(v) = 1$ in (5.7), which means the block is determined with one SOI in a form suitable for integration (we use this assumption only to simplify the formulas).

$$|v| = \int_{a_x}^{b_x}\int_{a_y+a_{yx}x}^{b_y+b_{yx}x}\int_{a_z+a_{zx}x+a_{zy}y}^{b_z+b_{zx}x+b_{zy}y} 1dxdydz = \int_{a_x}^{b_x}F_x(x)dx, \qquad (5.10)$$

where $F_x(x) = \int_{a_y+a_{yx}x}^{b_y+b_{yx}x}\int_{a_z+a_{zx}x+a_{zy}y}^{b_z+b_{zx}x+b_{zy}y}1dydz$ is only a function of x. In a similar manner one can write

$$F_x(x) = \int_{a_y+a_{yx}x}^{b_y+b_{yx}x}F_y(y)dy, \qquad (5.11)$$

with $F_y(y) = \int_{a_z+a_{zx}x+a_{zy}y}^{b_z+b_{zx}x+b_{zy}y}1dz$. This representation through univariate integrals enables computing $|v|$ by means of classical one-dimension integration techniques. Thus, in order to compute $|v|$, the one-dimensional integral of $F_x(x)$ should be computed; the function $F_x(x)$ can also be computed through one-dimensional integration and so on. We propose applying this principle for computing the variance of block v, and in the general case the covariance between the blocks (5.1).

One of the methods to compute numerically the integral in one dimension is the utilization of the Gauss-Legendre quadrature (Golub & Welsch 1969; Kahaner et al. 1989; Stoer & Bulirsch 2013). With this method, the integral is approximated with a finite sum of n terms (see Stoer and Bulirsch (2013) for details)

$$\int_a^b f(x)dx \approx \sum_{i=1}^{n} w_i f(x_i), \tag{5.12}$$

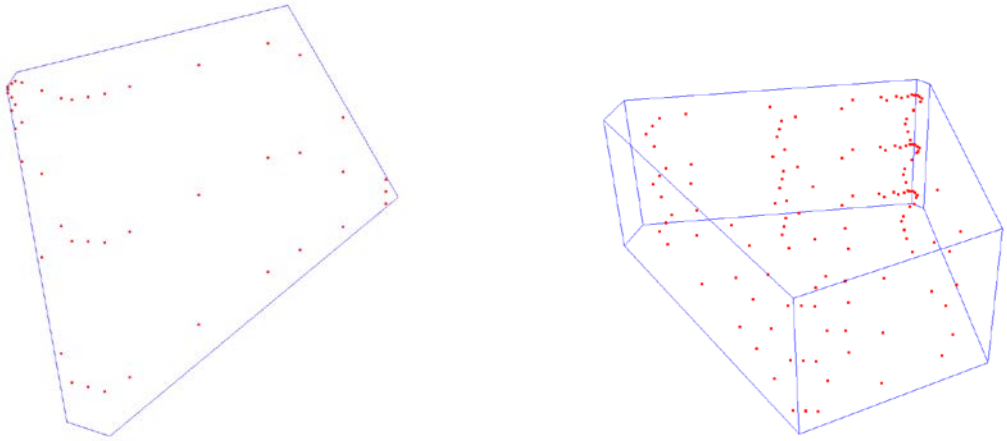
where the points x_i are the roots of the Legendre polynomial of degree n on (a, b) and the values w_i are the associated weights which can be found through Theorem 3.6.21 in (Stoer & Bulirsch 2013). The Gaussian-Legendre quadrature has two additional degrees of freedom relative to the integration through the regularly spaced points such as trapezoidal rule (Press et al. 1996). First, we are not restricted to computing the integrand in the regular spaced points, and second – we are able to find optimal weighting factors for each computed value of the integrand. The number n of terms in (5.12) is called the degree of the quadrature. It is worth mentioning the well-known result that the Gaussian quadrature of degree n is exact for polynomials of order up to $2n - 1$ inclusive. This property can be used for parameter optimization, since some of geostatistics covariance functions, such as the spherical, are defined with polynomials of small degree.

The error of approximation in (5.12) is given by the following formula from (Kahaner et al. 1989)

$$\frac{(b-a)^{2n+1}(n!)^4}{(2n+1)[(2n)!]^3} f^{(2n)}(\xi), \ where\ a < \xi < b. \tag{5.13}$$

Assessing the asymptotic behavior of the error of the Gauss-Legendre quadrature is a complicated task, which refers to complex analysis (Chawla & Jain 1968). The asymptotic behavior was also studied for the case of multiple integration (Elliott et al. 2011). We will limit ourselves to practical comparison of the rate of convergence for different methods for computing $Var\left(Z(v)\right)$.

When using Gauss-Legendre quadrature it is not possible to avoid computing $|v|$ as it was in the case of regular discretization due to irregular spacing of the integration points. Figure 5-4 depicts the placement of Gauss-Legendre integration points for v. Placement of points is performed when recursive integration in one dimension with a Gauss-Legendre quadrature of degree 3 is called successively on x, y and z. The integration is performed one SOI after another.



a) b)

Figure 5-4. Placement of integration points for Gauss-Legendre integration. a) top view b) side view.

The pattern of points is strongly irregular, and it is not obvious to see the three points placed in each direction. The figure below explains the placement inside a selected SOI for integration on variables x and y.

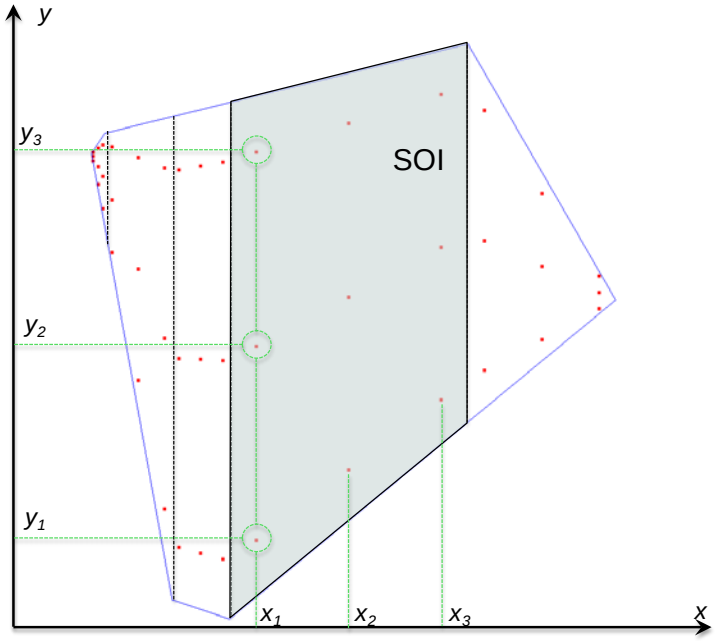


Figure 5-5. Integration points for a Gauss-Legendre quadrature of degree 3. Black dotted lines indicate the boundaries of the SOI; filled area corresponds to the SOI selected for demonstration; green dashed lines indicate the projections of the selected points on the coordinate axis.

Careful inspection of Figure 5-5 shows that each SOI is indeed filled with 9 points placed with the recursive integration procedure first on x and then on y.

5.5 Monte Carlo methods

Instead of applying subsequently integration in one dimension, computation of multidimensional integrals can be effectively performed with Monte Carlo integration techniques. The covariance $C(v_p, v_q)$ in (5.1) can be represented in the following form

$$C(v_p, v_q) = \int_{v_p}\int_{v_q} C(x, x') \frac{dx}{|v_p|}\frac{dx'}{|v_q|} = E[C(X, X')] \quad (5.14)$$

Where X and X' are uniform randomly distributed random points in v_p and v_q respectively. Using (5.14) the value of $C(v_p, v_q)$ can be estimated with a number N of pairs of points $(X_i, X_i') \in v_p \times v_q,\ i = 1 \ldots N$ sampled from the uniform distribution within these blocks

$$\hat{C}(v_p, v_q) = \frac{1}{N}\sum_{i=1}^{N} C(X_i, X_i'). \quad (5.15)$$

It is easy to see that such an estimate is unbiased, and to compute its variance:

$$E[\hat{C}(v_p, v_q)] = \frac{1}{N} N \int_{v_p}\int_{v_q} C(x, x') \frac{dx}{|v_p|}\frac{dx'}{|v_q|} = C(v_p, v_q), \quad (5.16)$$

$$Var[\hat{C}(v_p, v_q)] = \tfrac{1}{N} Var(C(X, X')), \quad (5.17)$$

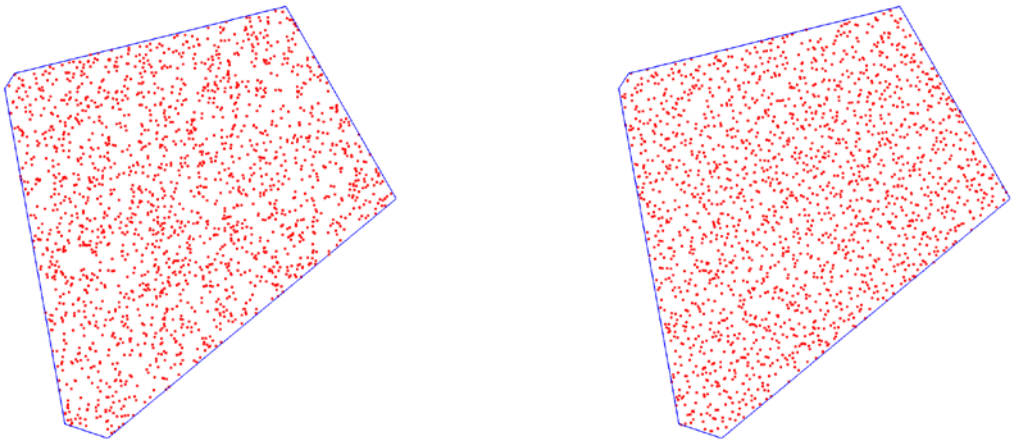
where $Var(C(X, X'))$ is an unknown constant, which can be further expanded into integrals. Following (Press et al. 1996), one can write for a sufficiently large N:

$$C(v_p, v_q) \approx \tfrac{1}{N}\textstyle\sum_{i=1}^{N} C(X_i, X_i') \pm \sqrt{\frac{Var(C(X,X'))}{N}} \quad (5.18)$$

Multidimensional integration Monte Carlo methods enable obtaining the value of $C(v_p, v_q)$ with significantly less computations as compared to subsequent application of deterministic methods. The problem of the classical Monte Carlo is its rate of convergence: from formula (5.18) it is clear that the computation error declines as $N^{-\frac{1}{2}}$ which is significantly slower than for deterministic methods. It turns out that this parameter can be significantly improved up to by means of quasi-random sequences (see Press et al. (1996) for a review).

We apply Sobol' quasi-random sequences (Sobol' 1967) for computations of $C(v_p, v_q)$ in (5.1). Since the integration space $v_p \times v_q$ has 6 dimensions, Sobol's sequences on $[0,1]^6$ are used. Application of Sobol' sequence in a 6 dimensional space provides the rate of decline $\frac{(\ln N)^6}{N}$ for the error which is almost as fast as $\frac{1}{N}$, see Press et al. (1996). These sequences fill the space in a more optimal manner relative to sampling from the normal distribution, avoid clustering and produce successively

finer and finer sets of discretizing points. Fast generation of Sobol' sequences can be performed with the method of Antonov and Saleev (1979). Figure 5-6 below demonstrates the top view on block v sampled with 1000 pseudo-random points with the standard Java random number generator and with Sobol' quasi-random sequence.



a) b)

Figure 5-6. Sampling 1000 pseudo-random points in v, top view with a) standard Java random uniform random number generator and b) Sobol' quasi-random sequences.

Although the difference between two sampling methods is not obvious from the first sight, the Sobol' quasi-random sequence proves its advantage further in the section devoted to the testing results.

The Monte Carlo approach to computing the block to block covariance requires an additional subroutine - sampling random (or quasi-random) sequences inside the block. There are two common approaches to this problem, and both of these use the bounding boxes D_p of v_p and D_q of v_q. These approaches are described below:

1) Extend the integrand function $C(x, x')$ to the region $D_p \times D_q$:

$$C(x, x') = \begin{cases} C(x, x') \; if \; (x, x') \in v_p \times v_q \\ 0 \; otherwise \end{cases} \tag{5.19}$$

and perform Monte Carlo on integration on $D_p \times D_q$.

2) Rejection sampling - the sampling is done in $D_p \times D_q$, and the generated pseudo-random points are accepted only if they belong to $v_p \times v_q$. Otherwise, the experiment is repeated until the desired points in the interior are found.

The first option does not reject the generated pseudo-random points, but increases the variance of the result due to considering the points outside $v_p \times v_q$ (see

Press et al. (1996) for discussion). The second option has an additional computation penalty for finding the points in the interior of $v_p \times v_q$ but enables to reduce the variance of the result. In our computations we use the second option.

Both approaches to sampling described above require testing the condition whether the pseudo-random point belongs to the block. Testing this condition for the arbitrary-shaped blocks is a complex problem, and we use the fact that the blocks are modeled as convex polyhedrons. For a convex polyhedron, representation (5.5) enables to test the desired condition. Indeed, if all m conditions in (5.5) are satisfied for a point, it is in the interior of v. In turns out that a testing algorithm based on representation (5.5) has a non-optimal ("pessimistic") behavior when the point is in the interior of v (which is the most desired situation in the sampling process). Indeed, in order to verify that all the m inequalities are satisfied, one should perform m times 3 multiplications, 2 additions and 1 comparison. Since the number m of faces of a convex polyhedron in practice can be more than 8 as on Figure 5-2b, the testing operation slows down the computation of the integral.

It turns out that the performance of the testing algorithm can be improved with the representation (5.7) of the block v. Suppose that the block v is represented with $NL(v)$ mutually exclusive systems of inequalities. Let $[xmin, xmax]$ be a projection of v onto the coordinate axis x; then $NL(v)$ systems in (5.7) define the following decomposition: $[xmin, xmax] = \cup_{i=1}^{NL(v)} [xmin_i, xmax_i]$, see Figure 5-7 below. Figure 5-7 depicts the block v, its bounding box D_v and the projections of SOI of the block onto the x axis. In order to check if a simulated pseudo-random point $X = (x_r, y_r, z_r)$ belongs to block v it is sufficient to find the interval $[xmin_i, xmax_i]$ which contains x_r and to check if the corresponding SOI from system (5.7) contains X. Later can be done by checking the validity of two double inequalities – one for y_r and one for z_r, which can be done with 6 multiplications, 6 additions and 4 comparisons. It is possible to find the interval $[xmin_i, xmax_i]$ containing x_r with binary search which requires $\lfloor \log_2 NL(v) \rfloor + 1$ comparisons. The total number of operations for checking if the random point X is inside v is then 6 multiplications, 6 additions and $\lfloor \log_2 NL(v) \rfloor + 5$ comparisons versus $3m$ multiplications, $2m$ additions and m comparisons for the previous test. Let us compare the total number of operations required to check if the simulated point is inside the block v, which top view is depicted on Figure 5-7. For this block $m = 8$, $NL(v) = 5$. The first approach then requires $24 + 16 + 8 = 48$ operations, whereas the second test requires $12 + 7 = 19$ operations, which is 2.5 times less.

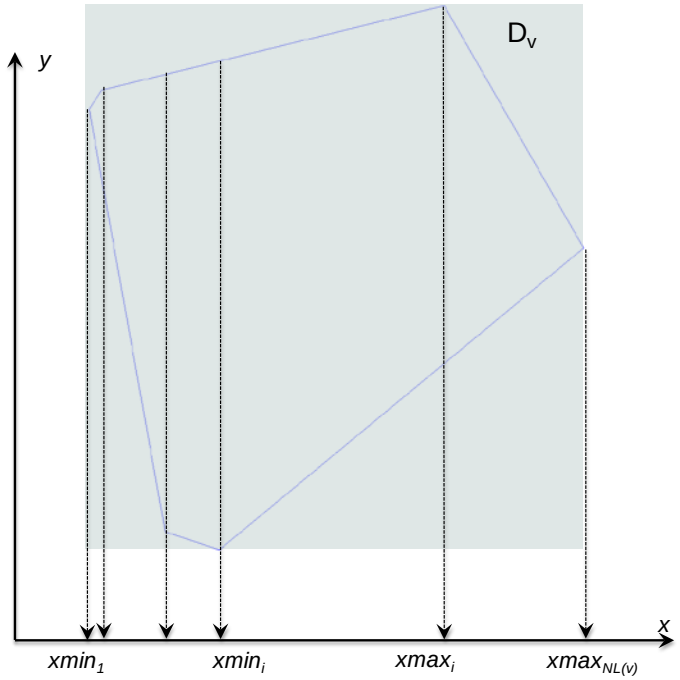


Figure 5-7. Projections of the SOI composing v onto x axis. The filled area corresponds to the bounding box D_v of v.

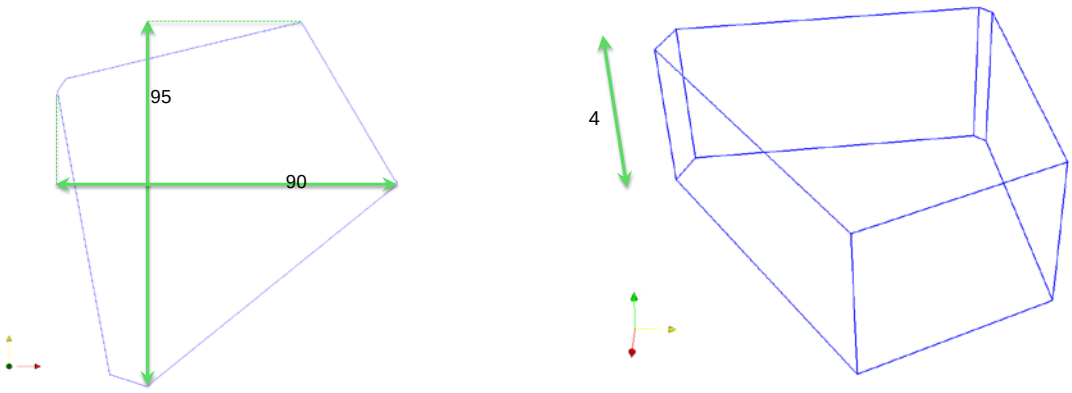
The proposed method for validating the point inside polyhedral condition enables to reduce significantly the time of computation of (5.1) with Monte Carlo methods.

5.6 Testing methodology

The performance of the four computation methods applied to the problem of computing $Var(Z(v))$ is compared:

1) Averaging of regularly placed points (5.8).
2) Successive application of Gaussian quadratures (5.12).
3) Monte Carlo integration with pseudo-random points produced by the standard Java uniform random number generator (5.15).
4) Monte Carlo integration with Sobol' quasi – random sequences using (5.15).

The block v that serves as the integration domain is depicted on Figure 5-8 with its dimensions. Many important applications of geostatistics include strong anisotropy in horizontal and vertical directions. Thus, it is not uncommon that the dimensions of the block v in horizontal and vertical directions differ by factor 10 and more. The exact dimensions of the selected block v for the test are $90 \times 95 \times 4$ units (see Figure 5-8). Due to strong anisotropy, the images of v are done with scaling factor 10 on the z axis.



a) b)

Figure 5-8. Dimensions of the block v. Scaling factor on z is 10.

Let us now discuss the integrand in (5.1) which we use to compute $Var(Z(v))$. The covariance function $C(x, x')$ is modeled through the geometric anisotropy approach (Chilès & Delfiner 2012; Leuangthong et al. 2011; Srivastava & Carter 1983). Following this approach the covariance function $C(x, x')$ is fully determined by three ranges R_x, R_y, R_z, two rotation angles α and θ and a function $g(h), h \in [0,1]$:

$$C(x, x') = g(|A(x' - x)|), \tag{5.20}$$

where A is a matrix defining a composition of rotation on angles α and θ and scaling

$$A = \begin{pmatrix} \frac{1}{R_x} & 0 & 0 \\ 0 & \frac{1}{R_y} & 0 \\ 0 & 0 & \frac{1}{R_y} \end{pmatrix} \begin{pmatrix} cos\,\alpha\,cos\,\theta & sin\,\alpha\,cos\,\theta & sin\,\alpha \\ -sin\,\alpha & cos\,\alpha & 0 \\ -cos\,\alpha\,sin\,\theta & -sin\,\alpha\,sin\,\theta & cos\,\theta \end{pmatrix}. \quad (5.21)$$

In order to perform our tests in a more realistic setting, we assume strong anisotropy ratio between the horizontal and vertical ranges of covariance. We explore 4 basic scenarios for the covariance function $C(x, x')$ and theirs variations.

1) $\rho_1(\vec{h})$ - spherical covariance with ranges $(100, 100, 4)$ with sill 1
2) $\rho_2(\vec{h})$ - exponential covariance with ranges $(100, 100, 4)$ with sill 1
3) $\rho_3(\vec{h})$ - double structure covariance with ranges $(100, 100, 4)$ with sill 1. The composition is $0.25\ \mathrm{Sph}\,(100, 100, 4) + 0.75\ \mathrm{Sph}\left(\frac{100}{3}, \frac{100}{3}, \frac{4}{3}\right)$.
4) $\rho_4(\vec{h})$ - spherical covariance with ranges $(100, 30, 4)$ with sill 1 and azimuth $\alpha = \pi/4$

Figure 5-9 illustrates the covariance functions $g(h)$ scaled to range 1 which correspond to the above listed four scenarios.

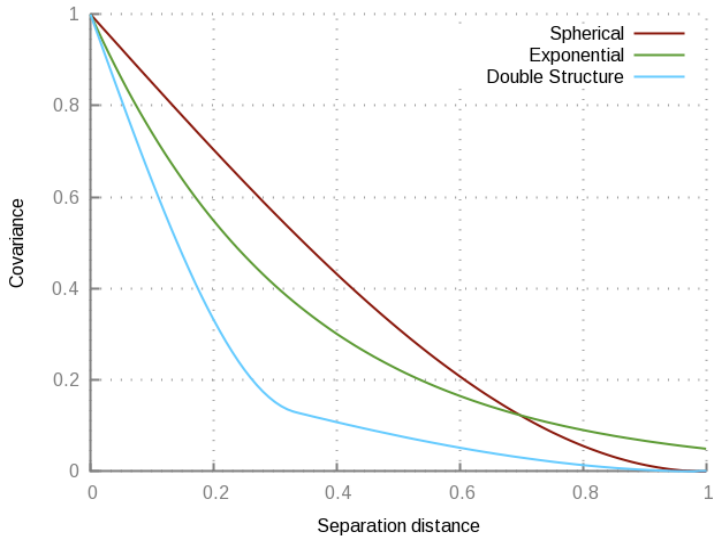


Figure 5-9. Scaled covariance functions $g(h)$ used for the tests.

This four basic scenarios correspond approximately to the situation when the correlation range is equal to the sizes of the block v. For each of these basic scenarios we consider the sub-scenarios when the vector of ranges (R_x, R_y, R_z) is multiplied by a coefficient $k = \frac{1}{5}, \frac{2}{5}, \frac{3}{5}, \frac{4}{5}, 1,2,3,4,5$. This way, for each of the four scenarios we are testing the computation of $Var(Z(v))$ for the covariance ranges varying approximately from $\frac{1}{5}$ to 5 times the size of the block.

The algorithms are advancing in the following manner in order to approach the true value of $Var(Z(v))$:

1) Regular discretization for covariance with ranges $R = (R_x, R_y, R_z)$ starts with discretization steps $S_1 = \frac{1}{2}(100,100,4)$. Due to the strong geometrical anisotropy of v, it is not possible to use equal steps in all directions. Taking the steps proportional to the dimensions of the block seems to be an appropriate solution. At the next iteration the discretization steps S are reduced proportionally: $S_i = S_{i-1}/1.2$. A total of 20 iterations are done.
2) Gaussian quadrature method starts with $N = 2$ integration points allowed for each of the six integration variables. The next step N is increased by 1 for all variables simultaneously. Computation proceeds up to $N = 20$.
3) Monte Carlo methods (both with pseudo-uniform points and Sobol' quasi-random sequence) start from sampling $N_6 = 100$ pairs of points in $v \times v$ on first iteration. At the next iteration this number is increased by 100. Computation proceeds up to $N_6 = 100{,}000$.

5.7 Test results

In this section we present the most essential results of comparing the four presented above computation techniques for determining the value of $Var\,(Z(v))$. For the sake of clarity here we only present the results for scenario 1 (Spherical covariance) and scenario 4 (Spherical covariance with azimuth). The complete results for all four scenarios can be found in Appendix C.

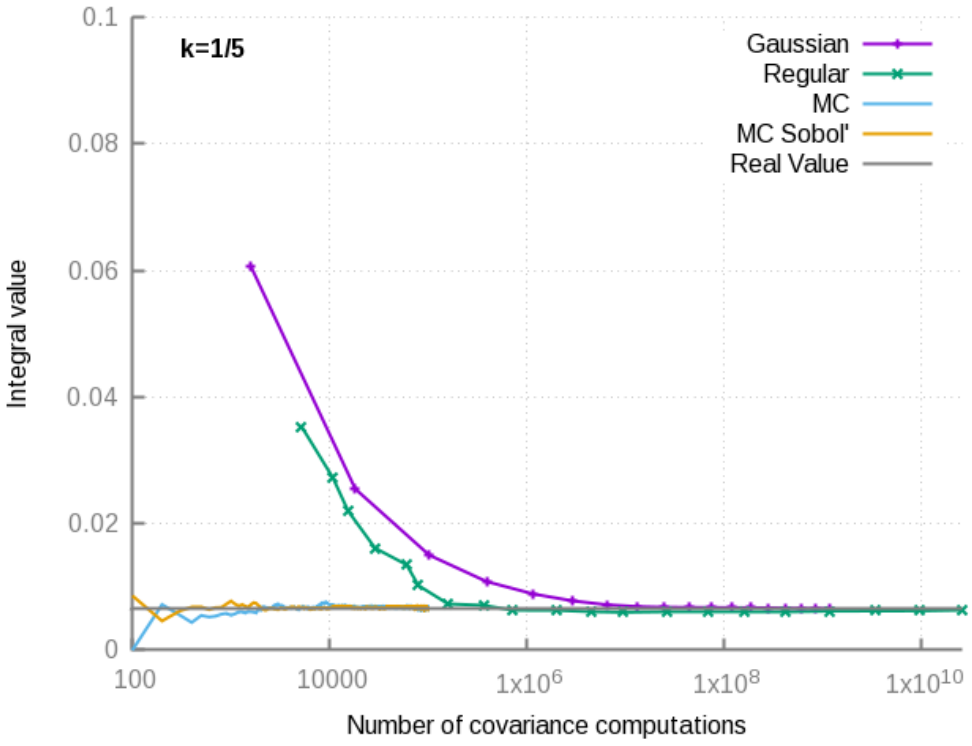
In order to estimate the quality of each computation method, the dependence of the computed value of $Var(Z(v))$ on the number of covariance function computations done by the method is plotted. This approach is useful, since all four methods have different parameters: regular discretization – the steps used, Gaussian quadrature – number of points placed, Monte Carlo method – number of pairs of points simulated. Indeed, Gaussian quadrature with $N = 10$ in 6D makes 10^6 computations of the covariance between points, whereas Monte Carlo with $N_6 = 10$ makes only 10. That explains the usage of the number of covariance computations as a parameter for comparing the method convergence.

Figure 5-10 and Figure 5-11 demonstrate the results for the 3 most representative cases: $k = \frac{1}{5}$ corresponds to a small covariance range relative to the block, $k = 1$ corresponds to covariance range of the size of the block and $k = 5$ corresponds to the covariance range significantly larger than the size of the block. In order to make the graphs comparable, the size of the ordinate axis is always set to 0.1. Each plot contains the true value of the covariance which is indicated by a thick gray horizontal line. We consider the final value computed by the Gaussian quadrature method to be the true value of $Var(Z(v))$. As it is seen from Figure 5-10

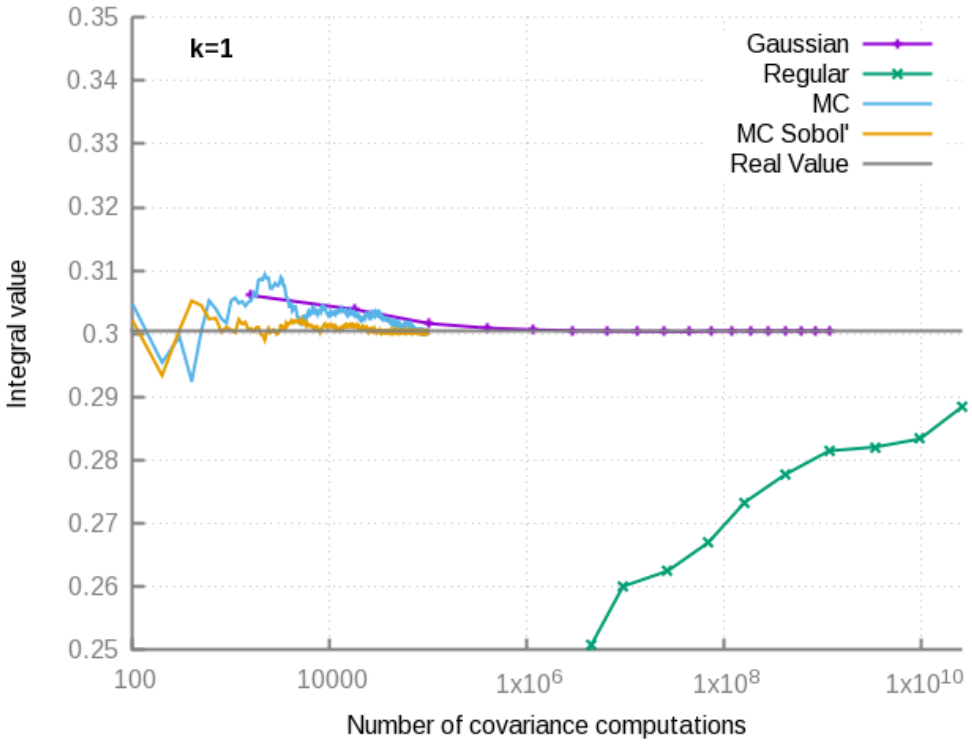
and Figure 5-11, the Monte Carlo methods require significantly less computations in order to converge to the correct answer. Amongst the Monte Carlo methods, Sobol' quasi-random sequence demonstrates the fastest convergence rate and the lowest dispersion as compared to independent uniform sampling.

Scenario 1 – Spherical covariance

a)



b)



c)

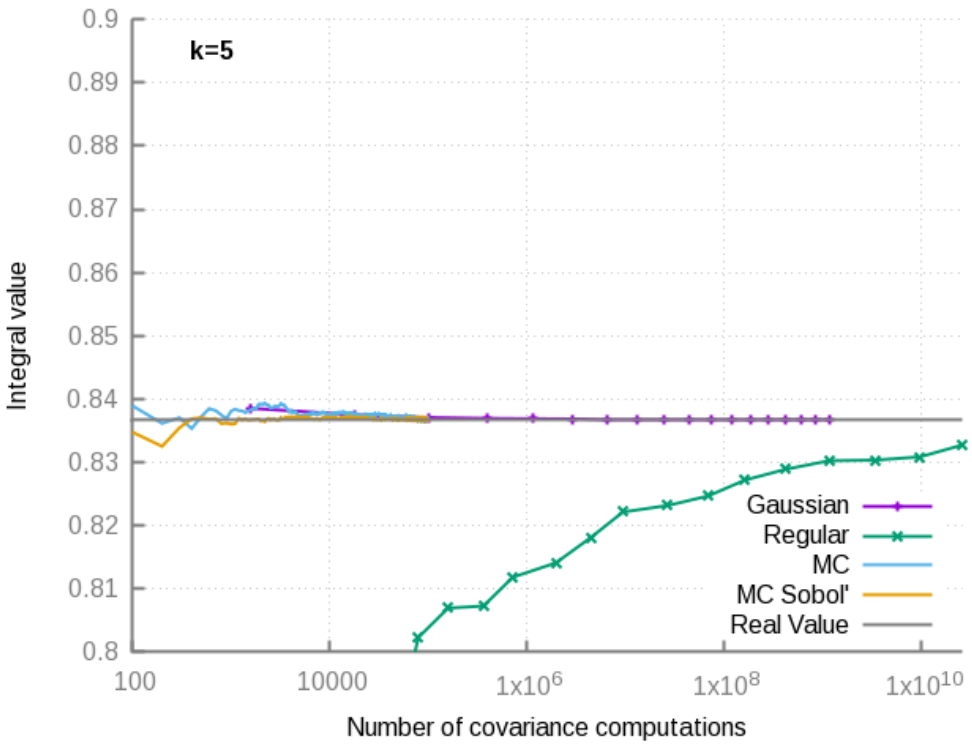
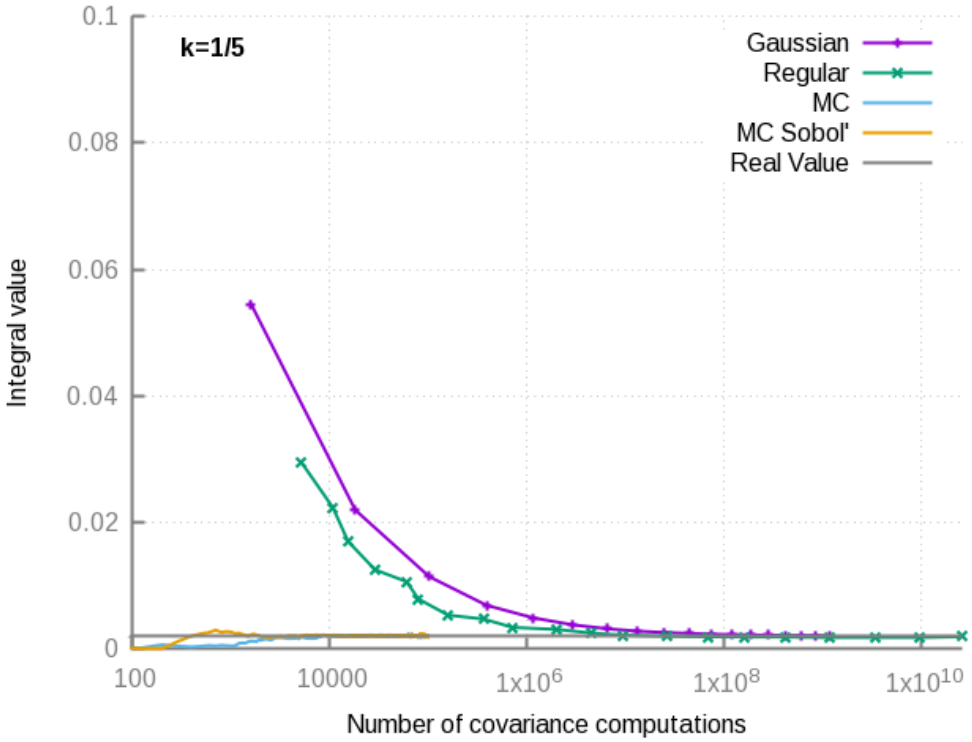


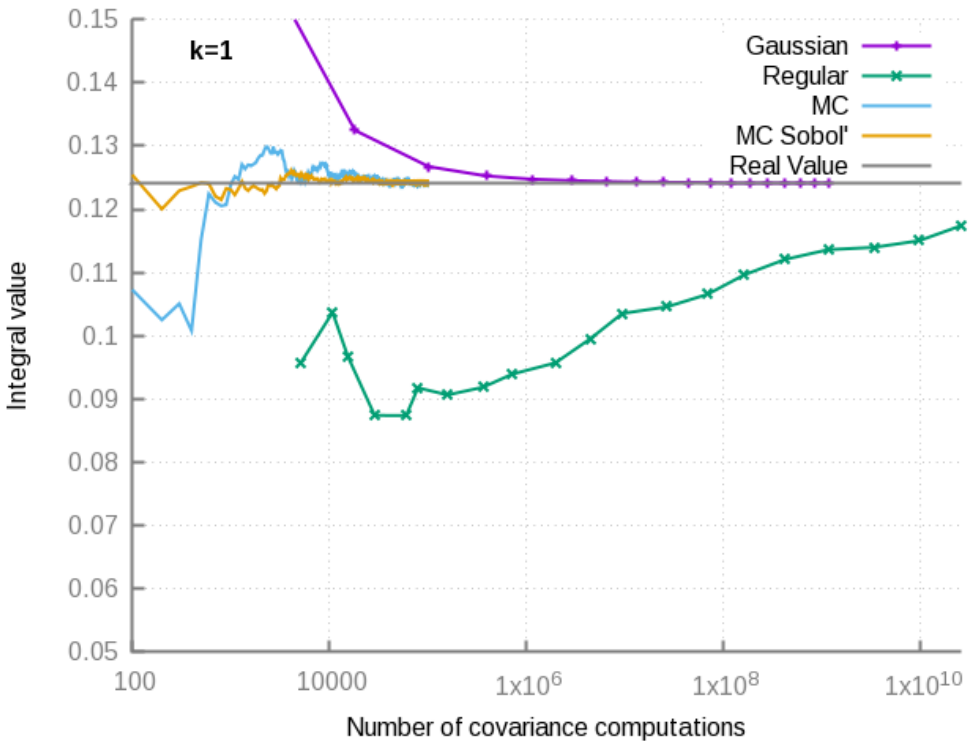
Figure 5-10. Computing $Var\big(Z(v)\big)$ for $\rho_1(h)$. Abscissa axis indicates the number of covariance computations done; the ordinate axis indicates the integral value for a)$k = \frac{1}{5}$ b) $k = 1$ c) $k = 5$.

Scenario 4 – Spherical covariance with azimuth $\frac{\pi}{4}$.

a)



b)



c)

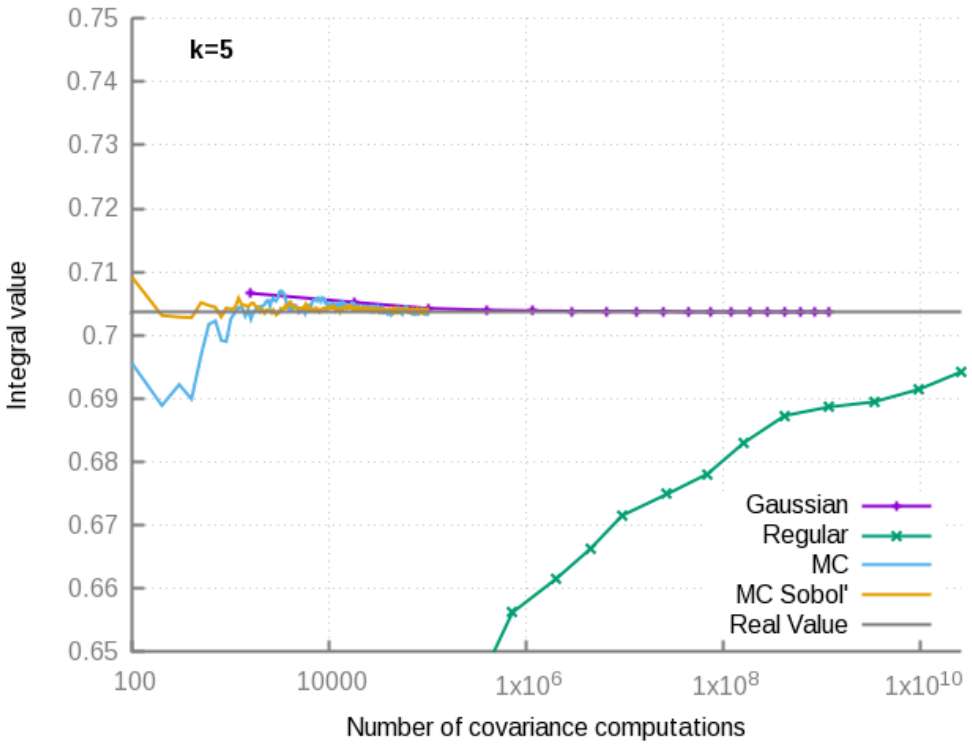


Figure 5-11. Computing $Var(Z(v))$ for $\rho_4(h)$. Abscissa axis indicates the number of covariance computations done; the ordinate axis indicates the integral value for a)$k = \frac{1}{5}$ b) $k = 1$ c) $k = 5$

5.8 Problem of approximation with set of points

Certain algorithms for geostatistical simulations on unstructured grids do not simulate directly on blocks, but simulate on a set of points inside a block and perform upscaling (Boucher & Dimitrakopoulos 2009; Gross & Boucher 2015). This approach raises the problem of placing a “sufficient” number of points in order to discretize the block (the discretization pattern can be irregular).The term “sufficient” can be made precise – since the simulation goal is to reproduce the first and second order moments of the spatial distribution, one can consider the system of discretizing points $\{x_i \in v, i = 1 \dots N\}$ to be “sufficiently good”, if approximating the value $Z(v)$ with $\frac{1}{N}\sum_1^N Z(x_i)$ leads to a correct reproduction of variance. The formal problem statement is the following: find a set of points $\{x_i \in v, i = 1 \dots N\}$ such that

$$Var\left(\frac{1}{N}\sum_{1}^{N} Z(x_i)\right) = \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} cov\left(Z(x_i), Z\left(x_j\right)\right) \approx Var\left(Z(v)\right). \qquad (5.22)$$

We contribute the results (Boucher & Dimitrakopoulos 2009; Gross & Boucher 2015) demonstrating that using Sobol’ quasi-random sequence-based discretization provides a good quality approximation in (5.22).

It is not hard to see that in the general case the problem of finding of such a system of points in the general case is different from the problem of computing the value of $Var\left(Z(v)\right)$. Indeed, verifying condition (5.22) for a set of N points requires significantly more computations than computing the integral value with Monte Carlo in (5.15). Also, the value of $Var\left(Z(v)\right)$ can be accurately found with Gaussian quadrature methods, as demonstrated in the previous section; meanwhile an attempt to use the Gaussian quadrature integration points $\{x_i \in v, i = 1 \dots N\}$ for approximating the value of $Var\left(Z(v)\right)$ with $\frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N} cov\left(Z(x_i), Z\left(x_j\right)\right)$ leads to poor results (see Figure 5-12).

The poor quality of approximating the block variance with Gaussian quadrature integration points through (5.22) is due to the fact that the pattern and density of these points in a block is very irregular and in the original Gaussian quadrature method different weights for these points are used (see (5.12)), but (5.22) attributes each of these points the same weight, which leads to a serious mismatch with the real value of $Var\left(Z(v)\right)$.

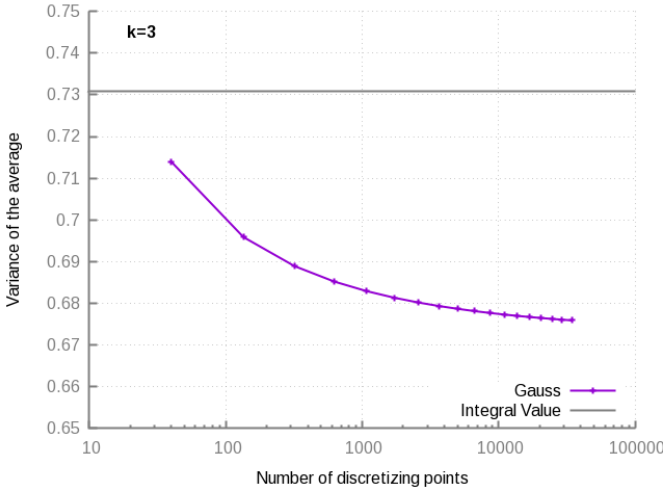
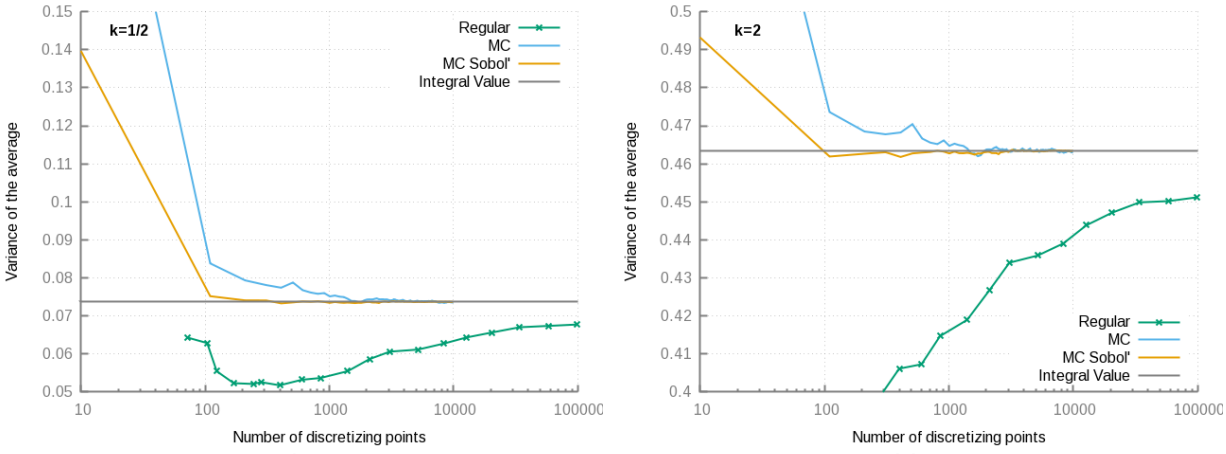


Figure 5-12. Approximating $Var\big(Z(v)\big)$ with Gaussian quadrature integration points through (5.22). Real value is shown with grey line.

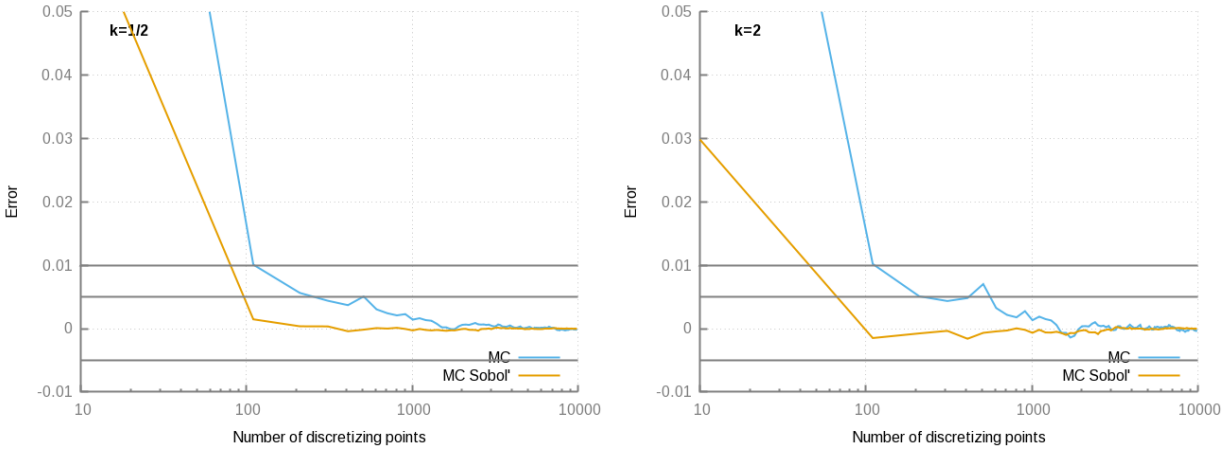
For the four scenarios presented in the previous section, the quality of approximating $Var\ \big(Z(v)\big)$ with (5.22) was verified for placing regularly spaced points, classical Monte Carlo random points and Sobol' quasi-random sequence in 3D points. The exact value of $Var\big(Z(v)\big)$ computed in the previous tests was used for reference. Figure 5-13 demonstrates a typical result of approximating $Var\big(Z(v)\big)$ with $Var\ \left(\frac{1}{N}\sum_1^N Z(x_i)\right)$.



a) b)

Figure 5-13. Approximating the block variance with a set of points for a) spherical covariance $\rho_1(h)$ b) exponential covariance function $\rho_2(h)$.

The corresponding absolute error plots for the Monte Carlo methods are depicted of Figure 5-14.



a) b)

Figure 5-14. Monte Carlo methods absolute error for approximating the block variance with a set of points for a) spherical covariance $\rho_1(h)$ b) exponential covariance function $\rho_2(h)$.

5.9 Conclusion

Test results enable us to conclude that Monte Carlo methods are the solution of choice for computing the value of $Var\big(Z(v)\big)$. As expected, the Sobol' sequence-based Monte Carlo shows faster convergence to the true value of $Var\big(Z(v)\big)$ and smaller error variance than the classical Monte Carlo. The worst quality of approximation is given by the regular sampling method.

It is reasonable to ask what is the desired accuracy for computing the covariance between the blocks of $Var\big(Z(v)\big)$. To answer this question an appropriate measure of accuracy should be determined. Let $\langle Var\big(Z(v)\big)\rangle$ indicate the computed value. The classical measures of accuracy are relative error $\delta_r = \frac{|\langle Var(Z(v))\rangle - Var(Z(v))|}{Var(Z(v))}$ and absolute error $\delta_a = \big|Var\big(Z(v)\big) - \langle Var\big(Z(v)\big)\rangle\big|$. In practical geostatistical applications when computing the covariance value between the points or blocks the relative error δ_r is rarely an appropriate accuracy measure since if the true value of the block variance $Var\big(Z(v)\big) = 0.01$, and the value computed by numerical integration is $\langle Var\big(Z(v)\big)\rangle = 0.02$ then $\delta_r = 1$ which corresponds to a 100% relative error. On the other hand, in the majority of practical applications there is no difference between block variance 0.01 and 0.02. In this case the absolute error $\delta_a = 0.01$ could indicate that the error is small. But in fact in order to state that $\delta_a = 0.01$ is small as a computation error, it is reasonable to compare this value to the range of values of the function $C(x, x')$, and in particular to the sill σ^2 of $C(x, x')$. This consideration gives another accuracy measure: $\delta_c = \frac{\delta_a}{\sigma^2}$ - the error relative to the unconditional point-support variance, which seems to be the most appropriate in the context of computing the covariance between the blocks. In all our tests $\sigma^2 = 1$, so $\delta_c = \delta_a$. From the practical experience the accuracy of covariance computations in

practical applications of geostatistics with 1% accuracy ($\delta_c = 0.01$) is more than sufficient. It is very rare that the other parts of the geostatistical workflow like distribution and covariance fitting give the same level of accuracy.

One of the problems which arise when Monte Carlo methods are used for computing the block to block covariance is that the resulting covariance matrix can be not positive semi-definite due to the introduced error. In particular such matrices can arise when a covariance between two small blocks v_1 and v_2 (relative to covariance range) which are adjacent is computed. In this case the Monte Carlo methods do not guarantee that the Cauchy-Schwarz inequality holds

$$cov\left(\frac{1}{|v_1|^2}\int_{v_1}\int_{v_1}\rho(x,x')dxdx', \frac{1}{|v_2|^2}\int_{v_2}\int_{v_2}\rho(x,x')dxdx'\right) \leq \sqrt{var\left(\frac{1}{|v_1|^2}\int_{v_1}\int_{v_1}\rho(x,x')dxdx'\right)var\left(\frac{1}{|v_2|^2}\int_{v_2}\int_{v_2}\rho(x,x')dxdx'\right)}. \quad (5.23)$$

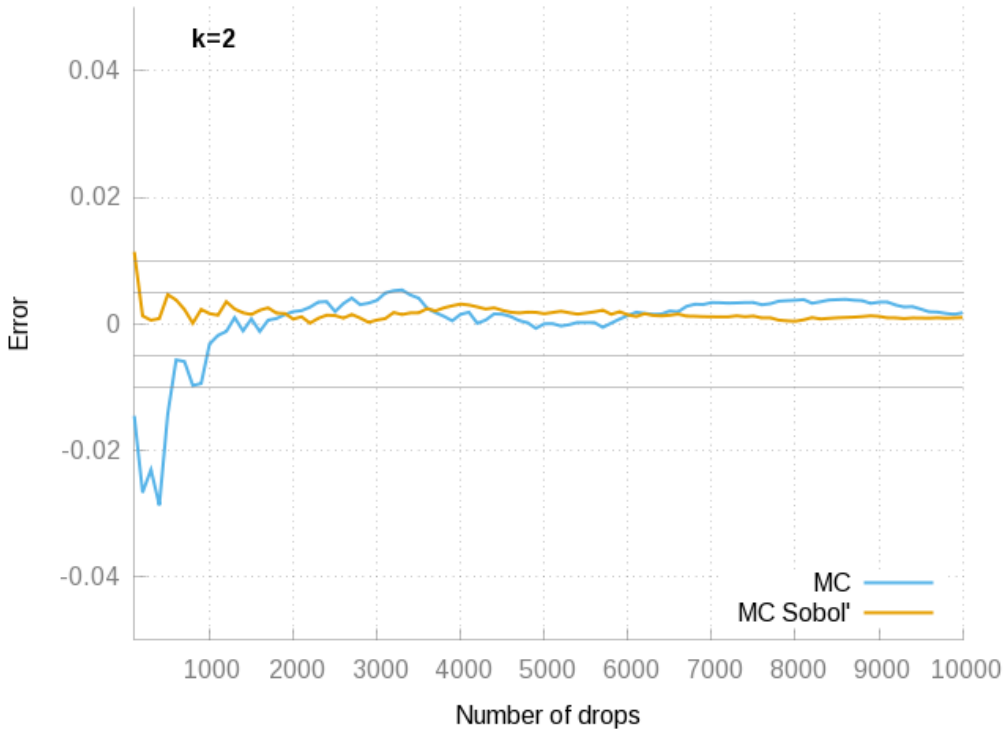
Since the matrix is not positive semi-definite, the corresponding multivariate Gaussian random vector does not exist and SGS would fail producing a realization of this vector due to the matrix inversion problems. In order to overcome this difficulty regularization methods for the covariance matrix should be used. One of the solutions which were applied in this thesis to regularize the kriging matrices in the SGS procedure is the diagonal increments method as described in (Sarra 2014). In the diagonal increments method a small increment $\epsilon > 0$ is selected and the kriging matrix K is substituted with the matrix $K + \epsilon I$, where I stands for the identity matrix. It can be shown that for a small ϵ the solutions of the initial and modified systems of linear equations are close, but the modified matrix is better conditioned than the original. The parameter ϵ is adjusted iteratively until the kriging system could be solved. Resulting simulations demonstrate satisfactory for practical applications statistics reproduction (although, less robust than the Gibbs Propagation algorithm). In order to avoid the problem of the not positive semi-definite matrices, it is advantageous to use the average value of a set of Sobol' quasi-random points placed in a block to approximate the average value of this block. This approach is more computation demanding but avoids the problem of the non-positive semi-definite matrices. Indeed, if for a given set of blocks $\{v_i, i = 1 \dots N_b\}$ the average value of each block in the Gaussian scale $Y(v_i) = \frac{1}{|v_i|}\int_{v_i} Y(x)dx$ is approximated with a linear combination of point values $\{Y(x_{ij}), j = 1 \dots n_j\}$ within the block v_i: $Y(v_i) \approx \frac{1}{n_i}\sum_{j=1}^{n_i} Y(x_{ij})$, the resulting covariance matrix $K = \left(cov\left(Y(v_i), Y(v_j)\right), i = 1 \dots N_b, j = 1 \dots N_b\right)$ is always positive semi-definite, since the set of RV $\left\{\frac{1}{n_i}\sum_{j=1}^{n_i} Y(x_{ij}), i = 1 \dots N_b\right\}$ has a multivariate Gaussian distribution (Hajek 2015). For any set of

discretizing points the elements of the covariance matrix K can be computed through (5.8).

Test results show that all four methods converge to the same value (even though the regular discretization method does not always reach the limit, one can hope that it would converge if the discretization step is further reduced). In real life we are limited in time for the problems of covariance computations and the fastest and the most accurate solution should be adopted. Our tests indicate that the Sobol' Monte Carlo method almost surely reaches the accuracy $\delta_c = 0.01$ for the number of drops $N_6 = 100$, and always for $N_6 = 200$. In fact, in the total of 36 experiments done (4 scenarios, 9 experiments each), only one experiment gives $\delta_c > 0.01$ for $N_6 = 100$, this situation is depicted on Figure 5-15a and, obviously, can be neglected, since the violation is minor. Our tests indicate that for $N_6 = 500$ the accuracy of Sobol' Monte Carlo computation $\delta_c < 0.005$.

In our hardware setting, computing Sobol' Monte Carlo integration for $N_6 = 1000$ takes less than $t = 1$ milliseconds. Although that seems fast, let us consider a practical problem of simulating with SGS on a grid with $N_b = 10^6$ blocks. Suppose that we have a multiprocessor computer which is able to compute the matrix of block to block covariance with the same time t as one block to block covariance (we use this approach in practice) and that we recomputed the covariances at every step of the simulation (this is also realistic – it is not that simple to store the $N_b \times N_b$ covariance matrix even if it is sparse and symmetric). In that case the simulation time only due to the covariance computations is $t * 10^6 \approx 16.5$ minutes which is already a significant amount of time. Further code optimizations should be used to reduce the computation time.

a)



b)

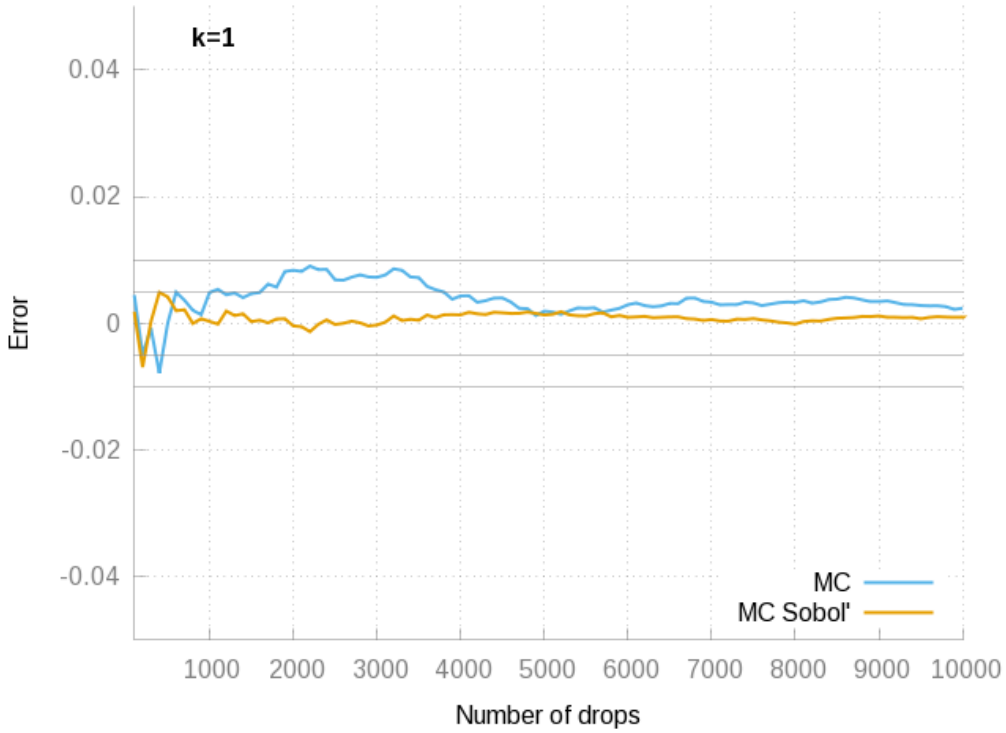


Figure 5-15. Error δ_c (signed) of Monte Carlo methods depending on the number of point drops (in 6D) for a) worst case, scenario 4, $k = 2$ and b) medium case, scenario 1, $k = 1$. Horizontal lines indicate $\pm 1\%$ and $\pm 0.5\%$ boundaries for absolute error.

Chapter 6: Navigation on unstructured grids

Résumé

A cause de ses structures irrégulières les maillages non-structurés requièrent des méthodes adaptées pour déterminer les voisinages des cellules. Dans ce chapitre on décrit l'utilisation de structure de données sur arbre k-d permettant d'effectuer la recherche efficace de voisinages sur les maillages non-structurés.

The term "navigation" in this work is applied to describe the number of searching operations applied to unstructured grids which arise in several different contexts. Although DGM-based theoretical models for simulation do not mention these methods explicitly, assembling a simulation algorithm without them seems unrealistic. The following list mentions some important searching operations required by a simulation algorithm on unstructured grids:

a) determining for each conditioning data point the block of the grid that contains this point (which is required by conditioning procedures)
b) determining the closest data points for every block (required by conditioning through kriging)
c) determining the neighborhood for every block (required by SGS procedure)
d) determining all blocks which are correlated with a given block (required for optimization of computing the block to block covariance matrix, used for Gibbs Propagation algorithm).

In the general case, one cannot determine the neighborhood of a given block on an unstructured grid from the index of this block (as in the case of regular IJK grids), so the algorithms used on regular grids, such as spiral search are not applicable to unstructured grids. It is also evident that direct lookup through all the $(N_b - 1)$ blocks of the grid in order to determine the closest m neighbors of a selected block v_p is extremely inefficient. Indeed, determining the closest m neighbors for every block by direct lookup is a $O\left({N_b}^2\right)$ in time. This result can be significantly improved with the use of k-d trees (Bentley 1975; Friedman et al. 1977), which enable searching for closest neighbors in proven $O(\log N_b)$ time. Application of k-d tree to the problem described above (determining the closest m neighbors for each block of the grid) leads to a $O(N_b \log N_b)$ algorithm, which is a significant improvement compared to the direct lookup through all the blocks.

Search trees were applied for navigation problems on regular and unstructured grids in geostatistical context (Boisvert & Deutsch 2010) for a similar to range of problems as discussed above. For searching problems in which the set of nodes on which the search is performed remains fixed (such as searching of the kriging neighborhoods), application of the optimal k-d tree (Bentley 1975) is the solution of choice. As demonstrated by Boisvert and Deutsch (2010), it outperforms the superblock search approach (Deutsch & Journel 1992). The situation is different

when the neighborhood search is applied to a set of blocks which is changing in time, as in the problem of finding the closest previously simulated blocks in the SGS procedure. Boisvert and Deutsch (2010) propose using k-d tree for the problem of SGS neighborhood search, organizing all N_b blocks in the tree, but marking the previously simulated nodes as “informed”, and performing the search only through the “informed” nodes. The disadvantage of this approach is that on the first iterations of SGS the k-d tree contains mostly “non-informed” nodes, and many of them have to be scanned in order to find the desired neighborhood. We propose an alternative solution to this problem – constructing the k-d tree dynamically from the previously simulated nodes. In the proposed approach, on the first iteration of SGS the k-d search tree is empty, and every time a new grid block is simulated, it is inserted into the k-d search tree (the insertion operation is $O(tree\ hight)$ in complexity). The construction by this way may not be optimal in the sense of Bentley (1975), but it contains only the “informed nodes”.

The proposed method of using the dynamic tree construction for the neighborhood search problem in SGS on an unstructured grid is compared to the method of building the optimal tree and using only the “informed” nodes. In our testing workflow we denote the approach of “informed” nodes as “dead leaves” approach, since we use the tree structure presented in (Friedman et al. 1977), where all the grid blocks correspond to the leaves of the k-d tree. The following testing methodology is proposed: a random grid of $N_b = 10^6$ nodes is generated in 3D, on this grid SGS is performed 2500 times with different random paths, the nearest 25 neighbors are searched. Average tree operation times (searching, or searching and insertion) are computed for each iteration of SGS. The average tree operation time for the first 150,000 iterations of SGS is depicted of Figure 6-1, the results for direct search are given for comparison (and as an example of a poor algorithm design).

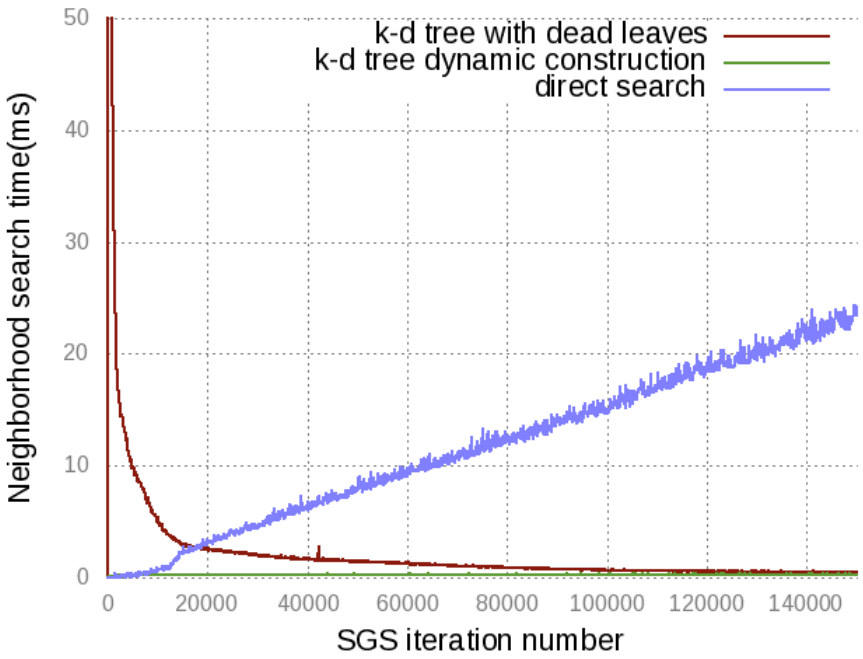


Figure 6-1. Dependence of the average time (ms) of the neighborhood search on a grid of 10^6 blocks from the SGS iteration number.

Our tests show that the dynamic construction approach demonstrates significantly better performance than the “dead leaves” approach for the first iterations of SGS. In our test the search time in the dynamically constructed and the “dead leaves” k-d trees become equivalent around iteration 500,000 of SGS, and further the “dead leaves” solution becomes more efficient. However, the high costs of the first 500,000 iterations of the “dead leaves” approach makes an important contribution to the total time required to perform the 10^6 iterations of SGS, and in our tests the dynamic tree construction requires in average almost twice less time to perform SGS on the full grid of 10^6 blocks than the “dead leaves” approach (see Figure 6-2 for average cumulative time).

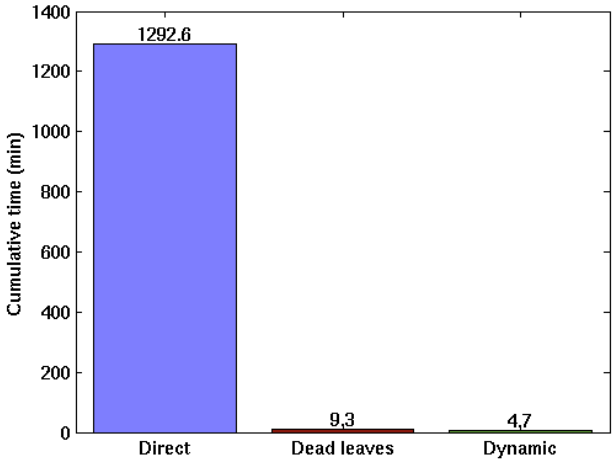


Figure 6-2. Average cumulative time (min) for direct search, “dead leaves” and dynamic k-d tree construction.

In the tests considered in this thesis the number of blocks in the grid rarely exceeds $N_b = 3 \times 10^6$ and we use the dynamic tree construction approach in the algorithm. The behavior of the “dead leaves” and dynamically constructed k-d trees in the SGS enables to propose a more optimal hybrid approach – for the big grids ($N_b > 10^6$) the search tree should be constructed dynamically and optimized at given periods of time. In the hybrid approach the k-d tree will contain only the pertinent nodes at the given iteration of SGS nodes, and will always be close to an optimally balanced k-d tree.

The cumulative time result on Figure 6-2 is also important since it gives a lower bound for the time of SGS on an unstructured grid of 10^6 nodes. Only the neighborhood searching time in this procedure takes 4.7 minutes.

Chapter 7: Case Study – Field X

Résumé

Ce chapitre est consacré à une application des méthodes de simulation avec le modèle Gaussien discret présenté dans cette thèse appliquée au cas d'étude réel – champ X – un gisement de gaz offshore avec un maillage tartan. La méthode de simulation avec DGM est comparée à la méthode de simulation classique, qui ignore l'effet du volume dans la simulation. Les résultats de comparaison des méthodes développées dans cette thèse par rapport aux méthodes classiques sont analysés.

Field X is an offshore gas reservoir. The reservoir model covers a large subsurface region with dimensions $140km \times 200km \times 350m$. Due to the enormous area covered by the model, it uses irregular meshing with grid refinement in the most important areas (see Figure 7-1). The refinement step in the horizontal direction varies from $250m$ to $4km$.The type of meshing used for field X is referred to as tartan meshing. The resulting model is not unstructured, since one can introduce an IJK coordinate system on it and will be able to use the IJK coordinates of the blocks in order to retrieve their neighbors. The dimensions of the grid in the IJK coordinate system are $132 \times 220 \times 31$. The grid remains irregular since it is composed of blocks of varying size.

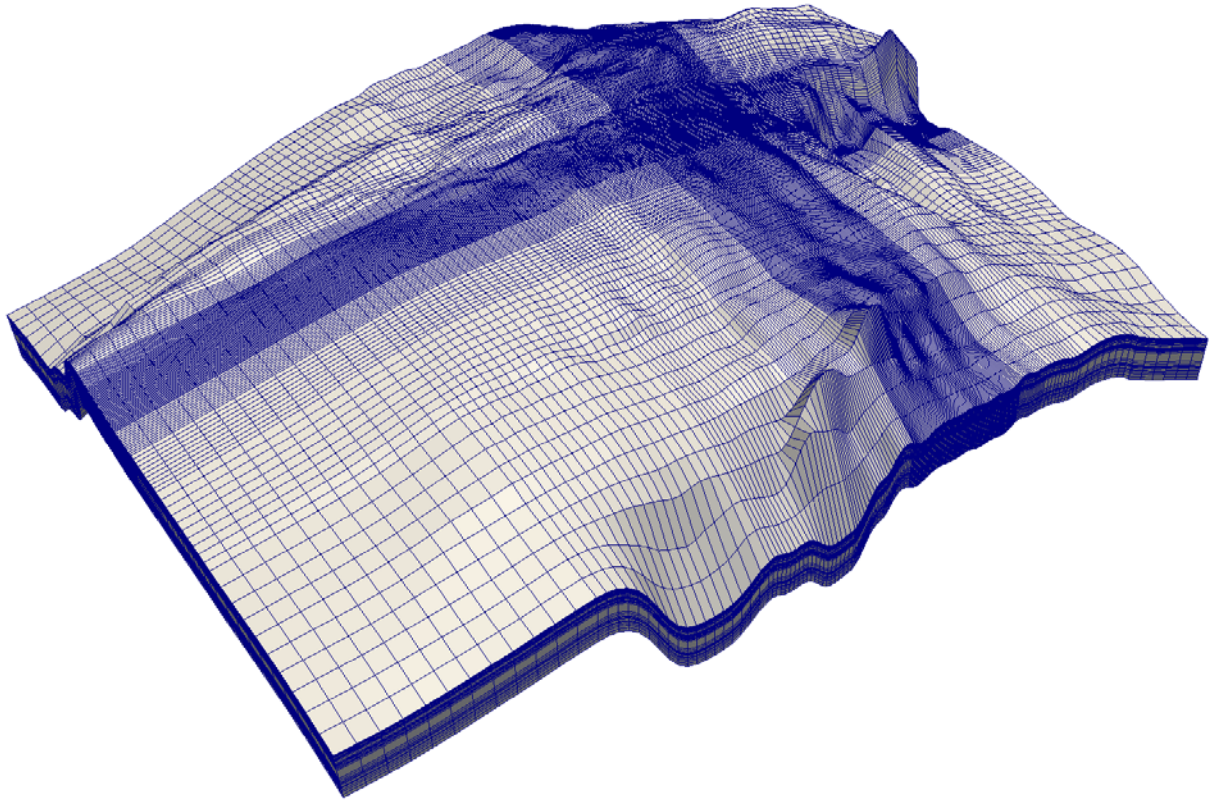


Figure 7-1. Field X reservoir model.

The goal of this case study is demonstrating the difference between the classical approach to geostatistical simulations and the proposed DGM-based approach which considers the change of support effect. For both approaches quasi-point support distribution and covariance function are used as initial inputs, the simulations are unconditional. The modeled petrophysical parameter is porosity with beta distribution on $[0.01,0.28]$ with parameters $\alpha = 2.15$ $\beta = 2.1$ (see Figure 7-2) and a spherical covariance function ρ in the Gaussian scale with ranges $3km \times 3km \times 30m$.

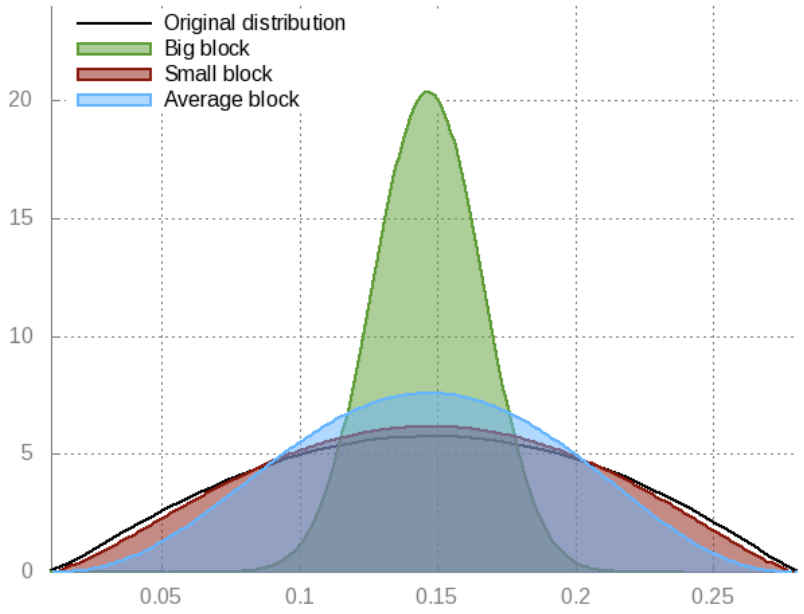


Figure 7-2. Average block porosity distribution.

The classical approach for the tartan grids consists in performing the simulation in the IJK space and transferring the results to the original space. The IJK formalism is useful since it simplifies the algorithmic part of simulation (neighborhood queries), enables applying simulation methods for regular meshes (i.e. spectral simulation) and the most important – ensures the continuity of the simulated parameter along the geological layers of the model. In order to make the classical simulation the point support inputs are upscaled to the "average support" of the model and the upscaled inputs are used. For the field X the block of the average size has dimensions $1km \times 900m \times 12m$. The input beta distribution was upscaled with DGM 2 in order to obtain the distribution of the average block (Figure 7-2). Figure 7-2 illustrates also the average porosity distribution of the smallest and the biggest blocks in the model for given inputs. The drawback of the classical approach is that it ignores the volumetric differences between the blocks which can lead to incorrect filling of the reservoir model.

In order to challenge the classical approach, we apply a DGM 2 simulation algorithm backed with Gibbs propagation algorithm for simulating the multivariate Gaussian random vector. The simulation algorithm cannot be applied directly to the original grid from Figure 7-1 since in that case the continuity of the simulated property along the geological layers would not be respected. In order to respect the continuity along the geological layers, the simulation should be performed in the depositional space which corresponds to the input geometry. The depositional space was derived as proposed in Mallet (2014). The reservoir model in the depositional space for field X is depicted on Figure 7-3. For this test we consider that the input point-support covariance function ρ is the covariance function of porosity in the depositional space.

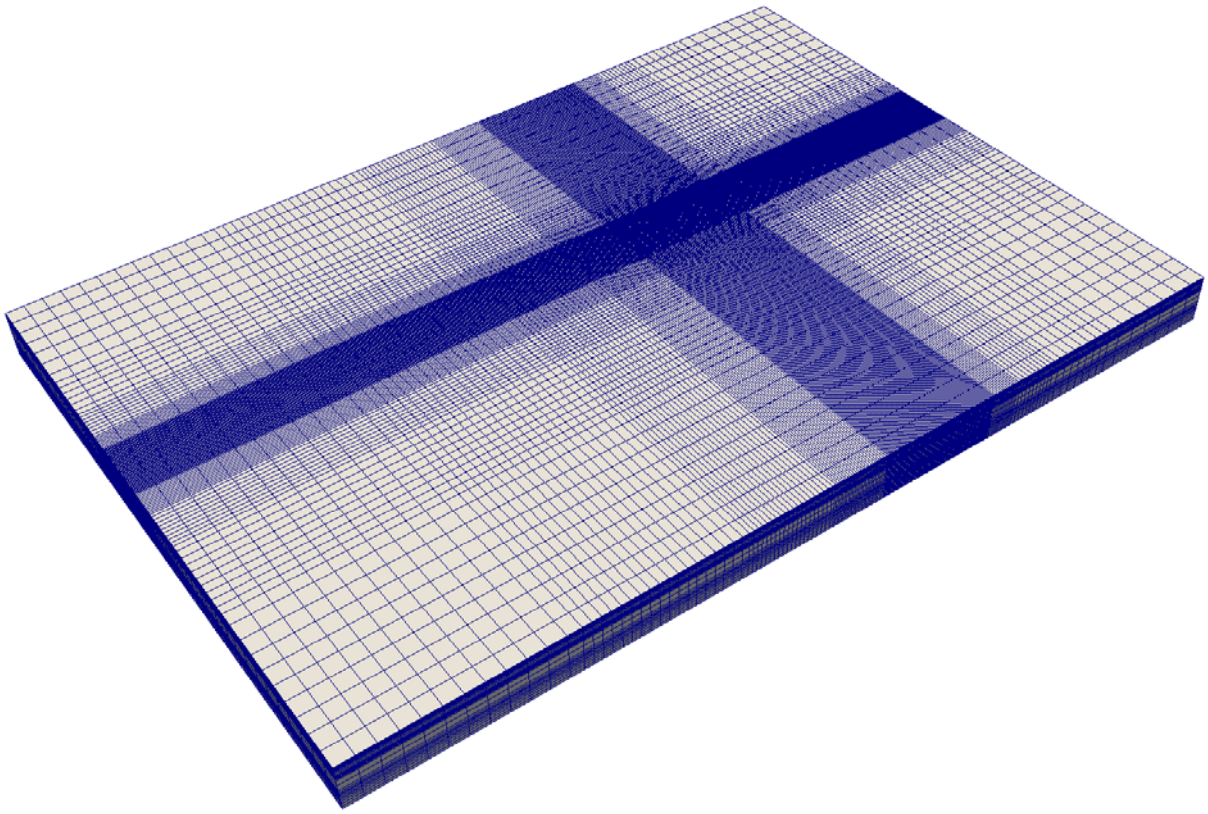


Figure 7-3. Depositional space reservoir model for field X.

The reservoir model in the depositional space preserves the mesh structure of the original model, which enables treating the volume support effect in the simulation. In addition to providing the continuity of the simulated property along the geological layers of the model, an argument for the idea of simulating in depositional space is that it is the space in which the hypothesis of the stationarity of the simulated physical parameter seems to be less invalid. Indeed, some similarity in the spatial structure of the studied physical parameter in two different regions of the reservoir is more likely to be expected if the depositional process took place in the same moment of time in the past.

The simulation results in the IJK space with classical approach (SGS with 50 neighbors) and in the depositional space with DGM 2 are demonstrated on Figure 7-4.

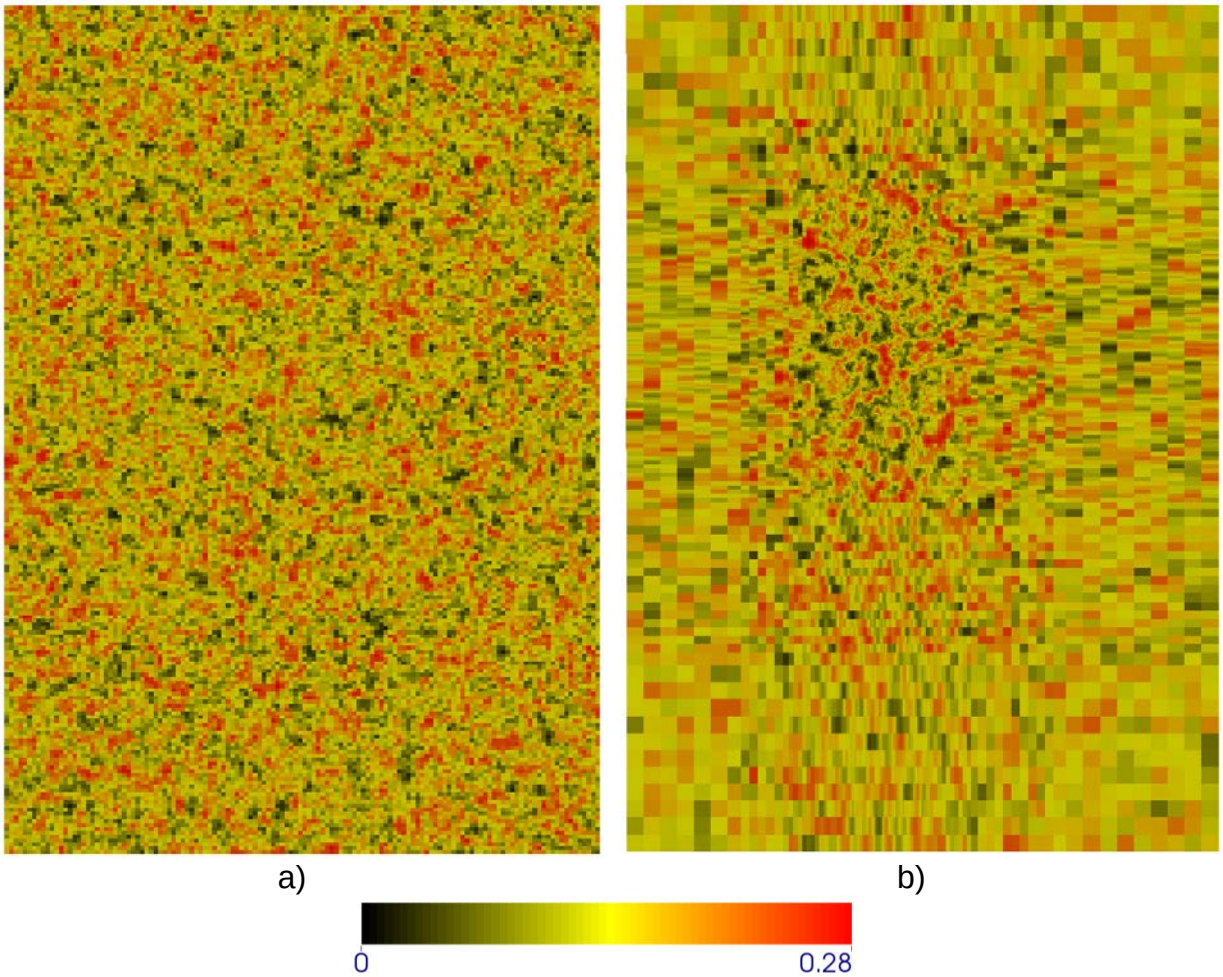
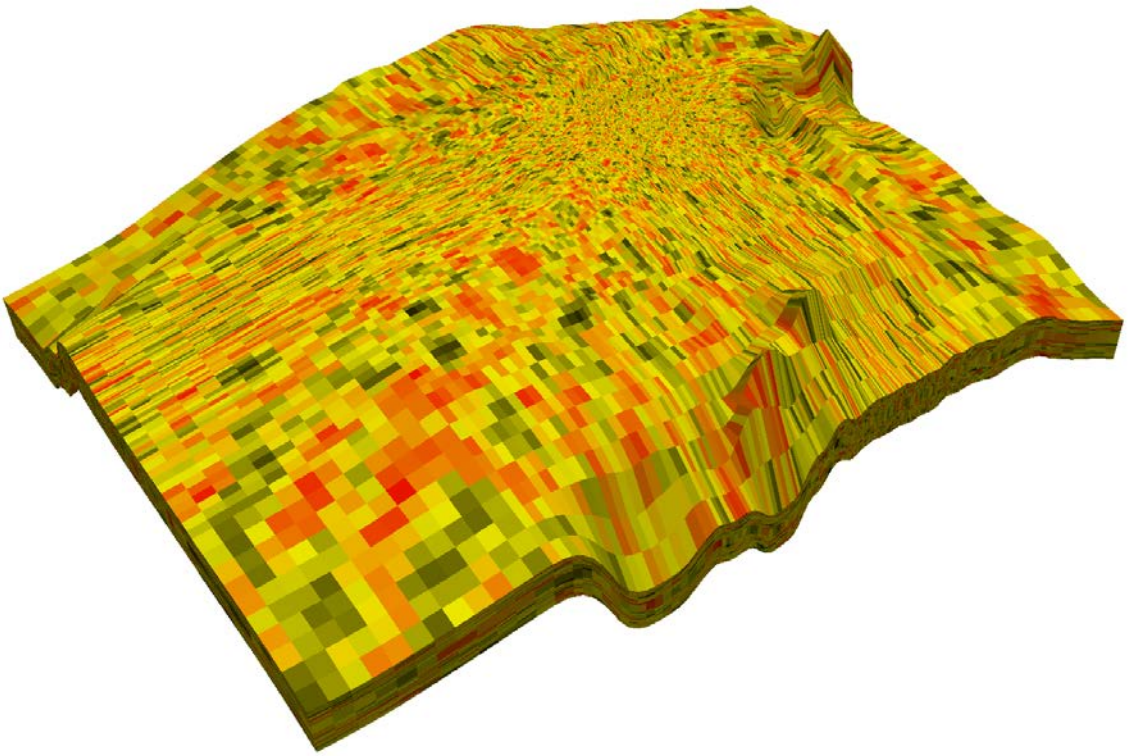


Figure 7-4. Simulation result (top view) a) in IJK space simulated with SGS b) in depositional space simulated with DGM 2.

The mesh on Figure 7-4 is not appropriate since it obstructs the image analysis. As expected, for the simulation in the IJK space the volume support information was lost and the same character of variations is observed everywhere in the model. For the simulation in the depositional space the refinement area demonstrates high variations of simulated porosity with an abundance of small porosity values (black) and large values (red). When approaching the corners of the model with the coarse blocks, the amount of extreme variations of porosity diminishes and less variation is observed. After simulation in the IJK and depositional spaces, the results are transferred to the original grid (Figure 7-5).

a)



b)

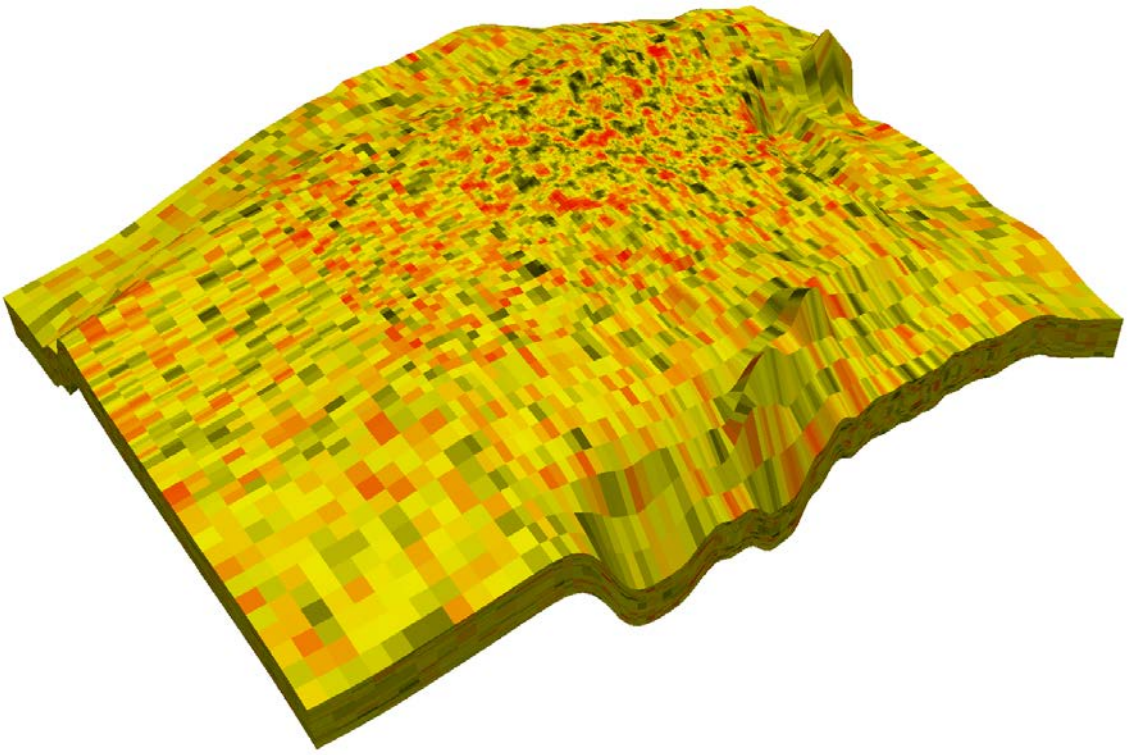


Figure 7-5. Simulation results. a) Classical approach b) DGM 2 approach. Vertical zoom factor 30.

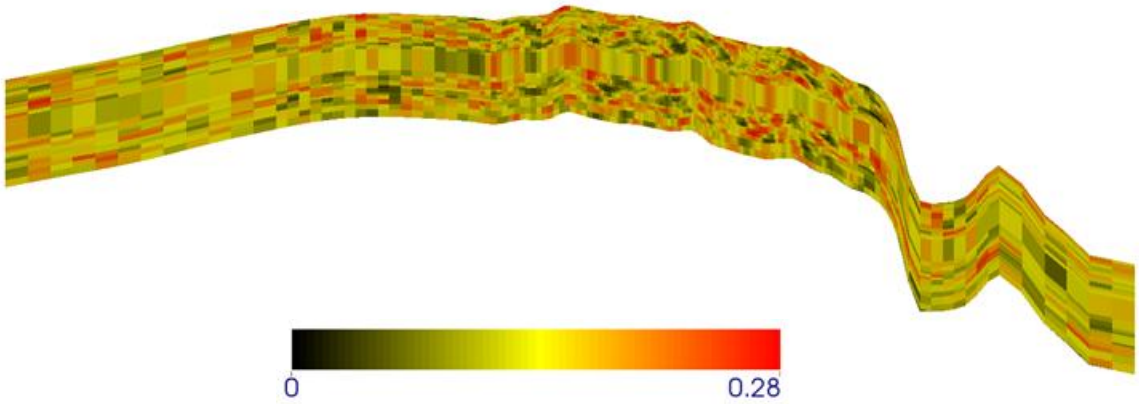


Figure 7-6. Vertical cross-section through the center of the model for the DGM 2 simulation. Vertical zoom factor 50.

It is visible on Figure 7-5a that the result provided by the classical approach is strongly impacted by the mesh structure and the desired continuity of the porosity field is not reproduced. Transferring the results from the IJK to the original space in this case can be considered as “stretching” the cells of the IJK model so that they get the desired shape. The effect of this “stretching” is visible on Figure 7-5a in the refinement zones. Although a stationary covariance function ρ was used for simulation, the resulting image does not demonstrate the patterns typical for stationary covariance models. On the other hand the DGM 2 simulation provides the desired model behavior. Transferring the simulation result from the depositional space to the original grid does not cause any artifacts. The refinement zone of reservoir demonstrates high variations of porosity whereas the coarse block regions look smooth. Figure 7-6 demonstrates a cross-section of the DGM 2 simulated model from Figure 7-5b. It is visible that the simulated porosity demonstrates continuity along the geological layers of the model. The example of case study X demonstrate that ignoring the volume support effect can lead to incorrect result for a geostatistical simulation on a reservoir model.

Chapter 8: Conclusions

Résumé

Les propriétés des méthodes de simulations géostatistiques sur les maillages non-structurés sont discutées. L'intégration de l'effet de volume dans les simulations et les algorithmes correspondants développés dans cette thèse ont permis d'améliorer la qualité de la modélisation. Le point est fait sur les étapes principales de la construction des algorithmes proposés et des améliorations possibles sont suggérées.

Classical geostatistical simulation algorithms reproduce correctly the spatial structure of the simulated natural phenomenon only on grids composed of identical blocks. When the block shape (the support) varies within the simulation domain, classical algorithms such as SGS, even though applicable on irregular lattices, do not reproduce correctly the marginal distributions of the block average values $Z(v)$ and the covariance between them due to ignoring the change of support effect (see case study X).

8.1 Contributions summary

The new approach for geostatistical simulation on unstructured grids presented in this thesis enables including the change of support effect implicitly in the simulation routine. Based on the simulation of multivariate Gaussian vectors, the proposed approach enables constructing a coherent theoretical model for geostatistical simulations in the presence of varying supports. Two generalizations of DGM for simulations on unstructured grids (DGM 1 and DGM 2) were presented. The difference between these models was demonstrated both analytically and through numerical tests in Chapter 3. Various methods for conditioning the simulations were presented and demonstrated. One of the main advantages of the simulation model developed in this thesis is that it is not restricted to application of the sequential simulation principle as the previously published solutions (Oz et al. 2003; Soares 2001) and enables using other sub-routines for simulating the random vectors, such as the Gibbs Propagation algorithm (Lantuéjoul & Desassis 2012). It was demonstrated in Chapter 3 that using the Gibbs Propagation algorithm provides better results in terms of reproducing the model statistics than the SGS algorithm. These results were published in several journal and conference papers, including (Zaytsev et al. 2016) and (Zaytsev et al. 2015).

The subject of simulating facies on unstructured grids was addressed. A formalization of the facies simulation problem on unstructured grids was proposed and a theoretical model for constructing pluri-Gaussian simulations on unstructured grids with DGM was presented. It was demonstrated that the resulting model has a limited scope of application, and alternative solutions, such as the algorithm of Gross and Boucher (2015), were discussed.

The application of methods for representing the convex polygons (Korenblit & Shmerling 2006) to the problem of geostatistical simulation on unstructured grids was demonstrated. Representing the convex blocks of unstructured grids as unions of mutually exclusive systems of inequalities enables constructing algorithms for computing the covariance between the blocks. The advantage of applying the Sobol' quasi-random sequence to the problem of computing the volumetric integrals of the covariance function was demonstrated with an extensive set of tests.

A case study demonstrates the application of the methods proposed in this thesis. Case study X presents a comparison of the classical approach to geostatistical simulations (simulation in the IJK space) with the proposed method of simulating in the depositional space considering the change of support effect. This test demonstrates that ignoring the change of support effect may lead to disastrous impact on the simulation result.

8.2 Perspectives

The methods proposed in this thesis depend on the subroutine that simulates the multivariate Gaussian random vectors. Further research in the area of simulating the multivariate Gaussian vectors can improve the performance of simulations on unstructured grids with DGM. Often in the practical applications the covariance matrix of the simulated multivariate Gaussian random vector is sparse and applying specialized methods adapted for sparse matrices (band-Cholesky factorization, multifrontal supernodal Cholesky factorization) enable improving significantly the algorithm performance (Rue & Held 2005).

Generalizations of other change of support models, such as Discrete Laguerrian Model (DLM, see Emery (2007)) for geostatistical simulations on unstructured grids can be considered. In DLM the point-support variable $Z(x) = \varphi\big(Y(x)\big)$ is a transformation of a variable $Y(x)$ with a gamma univariate distribution. Since the commonly applied geostatistical simulation methods (SGS, spectral simulation, Cholesky decomposition, turning bands) assume that $Y(x)$ is multivariate Gaussian, considering the DLM can extend the available class of simulation methods.

Appendix A: DGM

A.1 Hermite polynomials basis

Let $g(y)$ denote the density of standard Gaussian distribution. The Hermite polynomial $H_n(y)$ of degree $n = 1 \ldots \infty$ is defined by the formula

$$H_n(y)g(y) = \frac{d^n}{dy^n} g(y), \qquad y \in R \tag{A.1}$$

By definition, $H_0(y) = 1$. Normalized Hermite polynomial of degree $n = 0 \ldots + \infty$ are defined as follows

$$\chi_n(y) = \frac{1}{\sqrt{n!}} H_n(y), \qquad y \in R \tag{A.2}$$

with $0! = 1$ by convention. Let us consider the space $L^2(g)$ of piecewise-continuous functions $f: R \to R$, for which

$$\int_{-\infty}^{+\infty} f(y)^2 g(y) dy < +\infty. \tag{A.3}$$

In that space we define a scalar product of two functions as follows

$$(f_1, f_2) = \int_{-\infty}^{+\infty} f_1(y) f_2(y) g(y)\, dy. \tag{A.4}$$

Statement

The set of functions $\{\chi_j(y) = \frac{1}{\sqrt{j!}} H_j(y), j = 0 \ldots \infty\}$ form an orthonormal Hilbert basis in the space $L^2(g)$. We further refer to this basis as “Hermite polynomials basis”. A more detailed introduction in the subject can be found in (Chilès & Delfiner 2012).

The DGM-based geostatistical methods require decomposition of $\varphi(y)$ in the Hermite polynomials basis. In this basis $\varphi(y)$ is decomposed in the following manner:

$$\varphi(y) = \sum_{j=0}^{\infty} \varphi_j \chi_j(y), \tag{A.5}$$

where the coefficients $\{\varphi_i, i = 0 \ldots \infty\}$ are the coordinates of $\varphi(y)$ in the given basis.

A.2 Covariance approximation implied by DGM 2

This section demonstrates that simulations produced with generalized DGM 2 are biased in terms of covariance relative to the theoretically expected result. The theoretical expression of the covariance between $Z(v_p)$ and $Z(v_q)$ for two blocks v_p and v_q can be expressed as follows

$$cov\left(Z(v_p), Z(v_q)\right) = \frac{1}{|v_p||v_q|}\int_{v_p}\int_{v_q} C(x-x')dxdx'$$
$$= \frac{1}{|v_p||v_q|}\sum_{i=1}^{\infty}\varphi_i^2\int_{v_p}\int_{v_q}\rho^i(x-x')dxdx'. \quad \text{(A.6)}$$

For DGM 2 the block to block covariance is equal to

$$cov\left(Z(v_p), Z(v_q)\right) = cov\left(\varphi_{v_p}\left(Y_{v_p}\right), \varphi_{v_q}\left(Y_{v_q}\right)\right)$$
$$= \sum_{i=1}^{\infty}\varphi_i^2 r_p^i r_q^i cov\left(Y_{v_p}, Y_{v_q}\right)^i$$
$$= \sum_{i=1}^{\infty}\varphi_i^2 r_p^i r_q^i \left(\frac{1}{|v_p||v_q|r_p r_q}\int_{v_p}\int_{v_q}\rho(x-x')\,dxdx'\right)^i \quad \text{(A.7)}$$
$$= \sum_{i=1}^{\infty}\varphi_i^2 \left(\frac{1}{|v_p||v_q|}\int_{v_p}\int_{v_q}\rho(x-x')\,dxdx'\right)^i,$$

which does not coincide with the expected theoretical result. The difference Δ between the theoretical and implied by DGM 2 covariance values is

$$\Delta = \sum_{i=2}^{\infty}\varphi_i^2\left(\frac{1}{|v_p||v_q|}\int_{v_p}\int_{v_q}\rho^i(x-x')\;dxdx'\right.$$
$$\left. - \left(\frac{1}{|v_p||v_q|}\int_{v_p}\int_{v_q}\rho(x-x')\,dxdx'\right)^i\right), \quad \text{(A.8)}$$

which finalizes the proof.

A.3 Modeling transformation functions

The method used for modeling transform functions depends on the representation which is used for the input distribution $F(z)$ and for the transform function $\varphi(y)$. Since we are generally simulating properties with a physical meaning, we can consider that these properties do not take infinitely big values. Let us consider that the transform function $\varphi(y)$ is piecewise-linear on a closed and bounded interval $[y_0, y_N] \subset \mathcal{R}$, which is divided into N intervals $\Delta_i = [y_{i-1}, y_i]$, $i = 1 \dots N$ and that it is constant outside this interval:

$$\varphi(y) = \begin{cases} z_0, y < y_0 \\ \sum_{i=1}^{N} I\{x \in \Delta_i\} (a_i + b_i y), y \in [y_0, y_N]. \\ z_N, y > y_N \end{cases} \quad \text{(A.9)}$$

A.3.1 Modeling with Hermite polynomials

In order to find the change of support coefficient r for a block v and to model the block transformation functions $\varphi_v(y)$ as required by DGM, the decomposition (A.5) of $\varphi(y)$ in the basis of Hermite polynomials should be determined. This section gives the necessary equations for the decomposition coefficients.

$$\begin{aligned} \varphi_j &= \int_{-\infty}^{+\infty} \varphi(y)\, \chi_j(y) g(y) dy \\ &= \int_{-\infty}^{y_0} z_0 \chi_j(y) g(y) dy \\ &+ \int_{y_0}^{y_N} \left(\sum_{i=1}^{N} I\{x \in \Delta_i\} * (a_i + b_i y) \right) \chi_j(y) g(y) dy \\ &+ \int_{y_N}^{+\infty} z_N \chi_j(y) g(y) dy \\ &= z_0 \int_{-\infty}^{y_0} \chi_j(y) g(y) dy + \sum_{i=1}^{N} \int_{\Delta_i} (a_i + b_i y)\, \chi_j(y) g(y) dy \\ &+ z_N \int_{y_N}^{+\infty} \chi_j(y) g(y) dy \\ &= z_0 \int_{-\infty}^{y_0} \chi_j(y) g(y) dy + z_N \int_{y_N}^{+\infty} \chi_j(y) g(y) dy \\ &+ \sum_{i=1}^{N} a_i \int_{\Delta_i} \chi_j(y) g(y) dy + \sum_{i=1}^{N} b_i \int_{\Delta_i} y\, \chi_j(y) g(y) dy \end{aligned} \quad \text{(A.10)}$$

The first three integrals in the sum above can be easily computed (Chilès & Delfiner 2012):

$$\int_{\Delta_i} \chi_j(y) g(y) dy = \begin{cases} G(y)|_{\Delta_i} \ \ if\ j = 0 \\ \frac{1}{\sqrt{j!}} H_{j-1}(y) g(y)|_{\Delta_i} \ otherwise. \end{cases} \tag{A.11}$$

The last integral $\int_{\Delta_i} y\, \chi_j(y) g(y) dy$ is a bit more complicated. Let us use the following identity for the Hermite polynomials for $n \geq 1$:

$$H_{n+1}(y) = -yH_n(y) - nH_{n-1}(y). \tag{A.12}$$

From this one can find: $yH_n(y) = -H_{n+1}(y) - nH_{n-1}(y)$, $n \geq 1$. So, for $j \geq 1$

$$\begin{aligned} \int_{\Delta_i} y\, \chi_j(y) g(y) dy &= \frac{1}{\sqrt{j!}} \int_{\Delta_i} y\, H_j(y) g(y) dy \\ &= \frac{1}{\sqrt{j!}} \int_{\Delta_i} (-H_{j+1}(y) - jH_{j-1}(y))\, g(y) dy \\ &= -\frac{1}{\sqrt{j!}} \int_{\Delta_i} H_{j+1}(y)\, g(y) dy - \frac{j}{\sqrt{j!}} \int_{\Delta_i} H_{j-1}(y)\, g(y) dy \\ &= -\frac{1}{\sqrt{j!}} \left(H_j(y) + j * \begin{cases} G(y), & if\ j = 1 \\ H_{j-2}(y) g(y), & if\ j > 1 \end{cases} \right) \Bigg|_{\Delta_i} \end{aligned} \tag{A.13}$$

The case $j = 0$ should be treated separately

$$\int_{\Delta_i} y\, \chi_0(y) g(y) dy = \int_{\Delta_i} y\ g(y) dy = -\int_{\Delta_i} g'(y)\, dy = -g(y)|_{\Delta_i} \tag{A.14}$$

Finally, one can write the result:

$$\begin{aligned} \varphi_0 &= z_0 G(y_0) + z_N \big(1 - G(y_N)\big) + \sum_{i=1}^{N} a_i G(y) - \sum_{i=1}^{N} b_i g(y) \Bigg|_{\Delta_i} \\ &= z_0 G(y_0) + z_N \big(1 - G(y_N)\big) + \sum_{i=1}^{N} \big(a_i G(y) - b_i g(y)\big) \Bigg|_{\Delta_i} \end{aligned} \tag{A.15}$$

$$\varphi_1 = -z_0 g(y_0) - z_N g(y_N) + \sum_{i=1}^{N} a_i H_0(y) g(y) - b_i [H_1(y) g(y) + G(y)] \Big|_{\Delta_i}$$

$$= -z_0 g(y_0) - z_N g(y_N) + \sum_{i=1}^{N} \big(a_i - b_i H_1(y)\big) g(y) - b_i G(y) \quad \text{(A.16)}$$

$$= -z_0 g(y_0) - z_N g(y_N) + \sum_{i=1}^{N} \big(a_i - b_i \chi_1(y)\big) g(y) - b_i G(y)$$

$$
\begin{aligned}
\varphi_j = &-\frac{1}{\sqrt{j!}}\Big(z_0 H_{j-1}(y_0)g(y_0) + z_N H_{j-1}(y_N)g(y_N)\Big) \\
&+\frac{1}{\sqrt{j!}}\sum_{i=1}^{N} a_i H_{j-1}(y)g(y) \\
&- b_i\big[H_j(y)g(y) + j\,H_{j-2}(y)g(y)\big]\Big|_{\Delta_i} \\
&= -\frac{1}{\sqrt{j!}}\Big(z_0 H_{j-1}(y_0)g(y_0) + z_N H_{j-1}(y_N)g(y_N)\Big) \\
&+ \frac{1}{\sqrt{j!}}\sum_{i=1}^{N} g(y)\big(a_i H_{j-1}(y) - b_i\big[H_j(y) + j\,H_{j-2}(y)\big]\big)\Big|_{\Delta_i} \\
&= -\frac{\sqrt{j-1!}}{\sqrt{j!}\sqrt{j-1!}}\Big(z_0 H_{j-1}(y_0)g(y_0) + z_N H_{j-1}(y_N)g(y_N)\Big) \\
&+ \sum_{i=1}^{N} g(y)\left(a_i \frac{1}{\sqrt{j-1!*j}} H_{j-1}(y)\right. \\
&\left. - b_i\left[\frac{1}{\sqrt{j!}}H_j(y) + \frac{j}{\sqrt{j-2!*(j-1)*j}}\,H_{j-2}(y)\right]\right)\Big|_{\Delta_i} \\
&= -\frac{1}{\sqrt{j}}\Big(z_0\chi_{j-1}(y_0)g(y_0) \\
&+ z_N\chi_{j-1}(y_N)g(y_N)\Big) + \sum_{i=1}^{N} g(y)\left(a_i \frac{1}{\sqrt{j}}\chi_{j-1}(y)\right. \\
&\left. - b_i\left[\chi_j(y) + \sqrt{\frac{j}{j-1}}\chi_{j-2}(y)\right]\right)\Big|_{\Delta_i} \quad for\ j \geq 2
\end{aligned}
\tag{A.17}
$$

Let us select the order of the truncation of the basis at degree 60 and demonstrate, how the anamorphosis function $\varphi_v(y)$ evolves depending on the change of support coefficient r. The function $\varphi(y) = e^y$ selected for the test is the Gaussian anamorphosis function for a standard lognormal random variable. The visible deviation from the exponent for $r = 1$ is due to the small number 60 of Hermite polynomials used. In practice at least 120 Hermite polynomials should be used to model accurately this function (the modeling quality can be assessed comparing the known variance from the input distribution

$Var\ Z(x)$ with the $Var\ \varphi\big(Y(x)\big)$ obtained after approximating $\varphi(y)$ with Hermite polynomials).

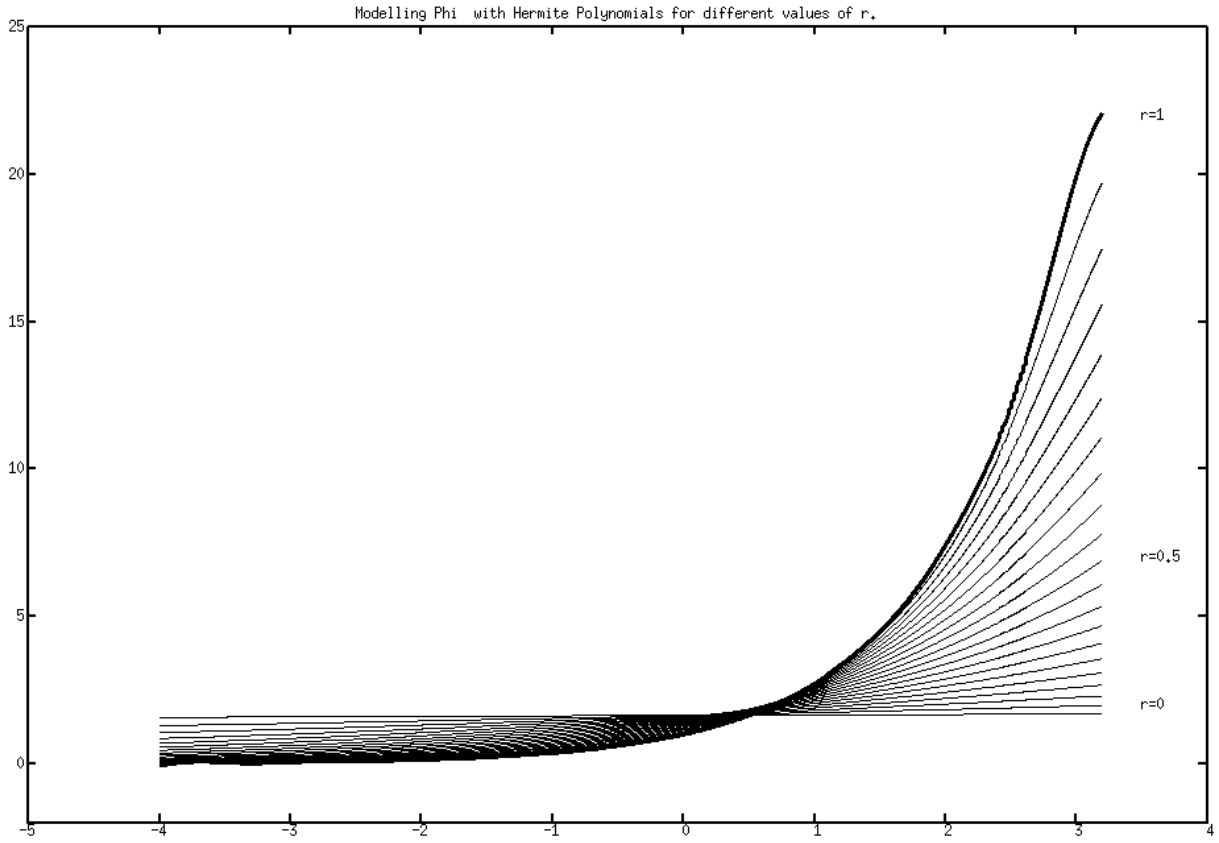


Figure A-1. Evolution of the anamorphosis function modeled with Hermite polynomials depending on the change of support coefficient r. The deviation from e^y is due to the small number 60 of polynomials used.

A.3.2 Exact modeling

In case when $\varphi(y)$ is piecewise-linear, the transformation function $\varphi_v(y)$ for a given r can be computed analytically as a consequence of Cartier's relation.

Let us compute the change of support function $\varphi_v(y)$ for the given block v with a change of support coefficient r:

$$\varphi_v(y) = \int_{-\infty}^{+\infty} \varphi\left(ry + \sqrt{1-r^2}t\right) g(t)dt \qquad \text{(A.18)}$$

Let $\tau = ry + \sqrt{1-r^2}t$. Then from (A.18) and definition of $\varphi(y)$

$$\begin{aligned}\varphi_v(y) &= \frac{1}{\sqrt{1-r^2}}\int_{-\infty}^{+\infty}\varphi(\tau)g\left(\frac{\tau-ry}{\sqrt{1-r^2}}\right)d\tau \\ &= \frac{1}{\sqrt{1-r^2}}\int_{-\infty}^{y_0} z_0 g\left(\frac{\tau-ry}{\sqrt{1-r^2}}\right)d\tau \\ &+ \frac{1}{\sqrt{1-r^2}}\sum_{i=1}^{N}\int_{y_{i-1}}^{y_i}(a_i+b_i\tau)g\left(\frac{\tau-ry}{\sqrt{1-r^2}}\right)d\tau \\ &+ \frac{1}{\sqrt{1-r^2}}\int_{y_N}^{+\infty} z_N g\left(\frac{\tau-ry}{\sqrt{1-r^2}}\right)d\tau\end{aligned} \tag{A.19}$$

Now, make the back transform: $t = \frac{\tau-ry}{\sqrt{1-r^2}}$, and, for sake of simplicity, introduce an auxiliary variable $\tilde{y}_i = \frac{y_i-ry}{\sqrt{1-r^2}}$.

$$\begin{aligned}\varphi_v(y) &= \int_{-\infty}^{\tilde{y}_0} z_0 g(t)dt + \sum_{i=1}^{N}\int_{\tilde{y}_{i-1}}^{\tilde{y}_i}\left(a_i + b_i\left(ry+\sqrt{1-r^2}t\right)\right)g(t)dt \\ &+ \int_{\tilde{y}_N}^{+\infty} z_N g(t)dt \\ &= z_0 G(\tilde{y}_0) + \sum_{i=1}^{N}\left((a_i+b_i ry)G(t)\big|_{\tilde{y}_{i-1}}^{\tilde{y}_i} + b_i\sqrt{1-r^2}\int_{\tilde{y}_{i-1}}^{\tilde{y}_i} t g(t)dt\right) \\ &+ z_N\left(1-G(\tilde{y}_N)\right) \\ &= z_0 G(\tilde{y}_0) + \sum_{i=1}^{N}(a_i+b_i ry)G(t) - b_i\sqrt{1-r^2}g(t)\Big|_{\tilde{y}_{i-1}}^{\tilde{y}_i} \\ &+ z_N\left(1-G(\tilde{y}_N)\right),\end{aligned} \tag{A.20}$$

which gives an explicit formula for the anamorphosis function for a given r. The same result as (A.20) can be found in (Emery 2009).

Let us verify, that the derived $\varphi_v(y) \to \varphi(y)$ when $r \to 1-$. Consider some fixed y. For the $i-th$ term, $s_i = (a_i + b_i ry)G(t) - b_i\sqrt{1-r^2}g(t)\big|_{\tilde{y}_{i-1}}^{\tilde{y}_i}$ we have:

If $y < y_{i-1}$:

$\tilde{y}_i \to +\infty, \tilde{y}_{i-1} \to +\infty \Rightarrow s_i = (a_i + b_i ry)(1-1) - b_i\sqrt{1-r^2}(0-0) = 0.$

If $y > y_i$:

$\tilde{y}_i \to -\infty, \tilde{y}_{i-1} \to -\infty \Rightarrow s_i = (a_i + b_i ry)(0-0) - b_i\sqrt{1-r^2}(0-0) = 0.$

If $y \in [y_{i-1}, y_i]$:

$\tilde{y}_i \to +\infty, \tilde{y}_{i-1} \to -\infty \Rightarrow s_i = (a_i + b_i ry)(1-0) - b_i\sqrt{1-r^2}(0-0) = a_i + b_i ry = \varphi(y).$

If $y < y_0$, then $z_0 G(\tilde{y}_0) \to z_0 G(+\infty) = z_0$, otherwise $z_0 G(\tilde{y}_0) \to 0$

If $y > y_N$, then $z_N\big(1 - G(\tilde{y}_N)\big) \to z_N(1 - G(-\infty)) = z_N$,

otherwise $z_N\big(1 - G(\tilde{y}_N)\big) \to 0$.

Summarizing, we have for $r \to 1-$:

$$\varphi_v(y) \to \begin{cases} z_0, y < y_0 \\ \sum_{i=1}^{N} I\{x \in \Delta_i\}(a_i + b_i y), y \in [y_0, y_N] = \varphi(y) \\ z_N, y > y_N \end{cases}$$

Let us demonstrate how the anamorphosis function evolves depending on the change of support coefficient r when this modeling approach is used.

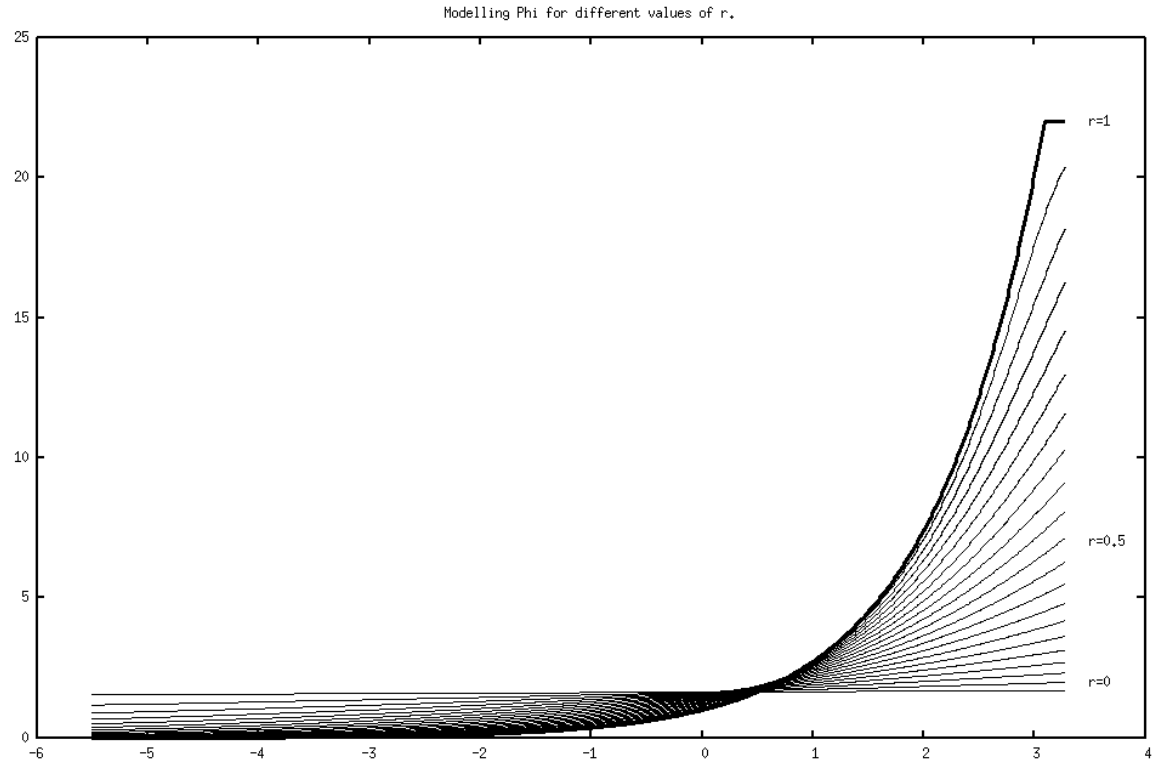


Figure A-2. Evolution of the anamorphosis function modeled with Hermite polynomials depending on change of support coefficient r.

Appendix B: PG-DGM

B.1 Hermite polynomials on the plane

A bivariate Hermite polynomial of degree $n = 0 \ldots + \infty$ and index $m = 0 \ldots n$ is defined as follows

$$H_{n,m}(x,y) = H_{n-m}(x)H_m(y) \qquad \forall (x,y) \in R^2 \tag{B.1}$$

A normalized bivariate Hermite polynomial of degree $n = 0 \ldots + \infty$ and index $m = 0 \ldots n$ is defined by

$$\chi_{n,m}(x,y) = \frac{1}{\sqrt{(n-m)!\,m!}} H_{n,m}(x,y) \qquad \forall (x,y) \in R^2 \tag{B.2}$$

Let us derive some properties of $\{\chi_{n,m}(x,y), n = 0 \ldots + \infty, m = 0 \ldots n\}$:

Property 1

Let $G(x,y) = G(x)G(y)$ denote the joint distribution of two independent standard Gaussian RVs. Let $L^2(G)$ denote the space of functions f for which

$$\int_R \int_R f^2(x,y) g(x) g(y) dx dy < +\infty \tag{B.3}$$

Then the set of bivariate polynomials $\{\chi_{n,m}(x,y),\ n = 0 \ldots + \infty, m = 0 \ldots n\}$ forms a Hilbert basis in the space $L^2(G)$ with the scalar product

$$< f_1, f_2 > = \int_R \int_R f_1(x,y) f_2(x,y) g(x) g(y) dx dy \qquad where\ f_1, f_2 \in L^2(G) \tag{B.4}$$

Indeed,

$$\begin{aligned} < \chi_{n_1,m_1}, \chi_{n_2,m_2} > &= \int_R \int_R \chi_{n_1-m_1}(x) \chi_{m_1}(y) \chi_{n_2-m_2}(x) \chi_{m_2}(y) g(x) g(y) dx dy \\ &= \int_R \chi_{n_1-m_1}(x) \chi_{n_2-m_2}(x) g(x) dx \times \int_R \chi_{m_1}(y) \chi_{m_2}(y) g(y) dy \\ &= \delta_{n_1-m_1,n_2-m_2} \delta_{m_1,m_2} = \begin{cases} 1 \ if\ m_1 = m_2, n_1 = n_2 \\ 0 \qquad\qquad otherwise \end{cases} \end{aligned} \tag{B.5}$$

In probabilistic terms, for independent standard Gaussian $Y_1(x)$ and $Y_2(x)$, Eqn.(B.33) means

$$E\{\chi_{n_1,m_1}(Y_1(x), Y_2(x))\, \chi_{n_2,m_2}(Y_1(x), Y_2(x))\} = \begin{cases} 1 \ if\ m_1 = m_2, n_1 = n_2 \\ 0 \qquad\qquad otherwise \end{cases}. \tag{B.6}$$

Also,

$$E\{\chi_{n_1,m_1}(Y_1(x),Y_2(x))\} = \int_R \int_R \chi_{n_1-m_1}(x)\chi_{m_1}(y)g(x)g(y)dxdy$$
$$= \int_R \chi_{n_1-m_1}(x)g(x)dx \times \int_R \chi_{m_1}(y)g(y)dy \quad \text{(B.7)}$$
$$= \begin{cases} 1 \; if \; m_1 = 0, n_1 = 0 \\ 0 \qquad otherwise \end{cases}$$

For $(Y_1(x), Y_1(x')) \sim \mathrm{B}(\rho_1(x,x'))$ and $(Y_2(x), Y_2(x')) \sim \mathrm{B}(\rho_2(x,x'))$, $Y_1(x) \perp Y_2(x)\ \forall x$:

$$E\{\chi_{n_1,m_1}(Y_1(x),Y_2(x))\ \chi_{n_2,m_2}(Y_1(x'),Y_2(x'))\} =$$
$$\int_{R^4} \chi_{n_1,m_1}(x_1,y_1)\,\chi_{n_2,m_2}(x_2,y_2)g_{\rho_1}(x_1,x_2)g_{\rho_2}(y_1,y_2)dx_1dx_2dy_1dy_2 =$$
$$\int_{R^2} \chi_{n_1-m_1}(x_1)\chi_{n_2-m_2}(x_2)g_{\rho_1}(x_1,x_2)dx_1dx_2 \times$$
$$\int_{R^2} \chi_{m_1}(y_1)\chi_{m_2}(y_2)g_{\rho_2}(y_1,y_2)dy_1dy_2 = \quad \text{(B.8)}$$
$$\delta_{n_1-m_1,n_2-m_2}\rho_1^{n_1-m_1}(x,x')\delta_{m_1,m_2}\rho_2^{m_1}(x,x') =$$
$$\begin{cases} \rho_1^{n_1-m_1}(x,x')\rho_2^{m_1}(x,x') \; if \; m_1 = m_2 > 0, n_1 = n_2 > 0 \\ 0 \qquad\qquad otherwise \end{cases}.$$

B.2 Decomposition into bivariate Hermite polynomials

Let us find the coefficients of decomposition of a function $f(x,y)$ in the basis $\{\chi_{n,m}(x,y), n = 0 \ldots +\infty, m = 0 \ldots n\}$. The representation of $f(x,y)$ in the basis of bivariate Hermite polynomials is the following

$$f(x,y) = \Sigma_{n=0}^{+\infty}\Sigma_{m=0}^{n} f_{n,m}\chi_{n,m}(x,y) \quad \text{(B.9)}$$

Due to the orthonormality property (B.5), the coefficient $f_{n,m}$ can then be found through taking a scalar product of f with $\chi_{n,m}$

$$< f, \chi_{n,m} > = f_{n,m} = \int_{R^2} f(x,y)\chi_{n,m}(x,y)g(x)g(y)dxdy \quad \text{(B.10)}$$

When f is a bivariate indicator function that can be decomposed into a product of univariate indicators

$$f(x,y) = I\{x \in \Delta^1, y \in \Delta^2\} = I\{x \in \Delta^1\}\, I\{y \in \Delta^2\}, \quad \text{(B.11)}$$

and the decompositions of the above-mentioned univariate indicators in the univariate Hermite polynomials basis are known

$$I\{x \in \Delta^t\} = \Sigma_{n=0}^{+\infty} c(t,n)\chi_n(x), for \; t = 1,2, \quad \text{(B.12)}$$

the decomposition coefficient $f_{n,m}$ in the basis of bivariate polynomials can be easily derived through the decomposition coefficients $\{c(1,n), n = 0 \ldots + \infty\}$ and $\{c(2,n), n = 0 \ldots + \infty\}$ in the univariate polynomials basis

$$f_{n,m} = \int_R I\{x \in \Delta^1\}\chi_{n-m}(x)g(x)dx \times \int_R I\{y \in \Delta^2\}\chi_m(y)g(y)dy = c(1, n-m) \times c(2, m). \quad \text{(B.13)}$$

So, in order to determine the decomposition of the indicator $I\{Y_1(x) \in \Delta t_i^1, Y_2(x) \in \Delta t_i^2\}$ with independent $Y_1(x)$ and $Y_2(x)$ it is sufficient to know the decomposition of indicators $I\{Y_1(x) \in \Delta t_i^1\}$ and $I\{Y_2(x) \in \Delta t_i^2\}$ into univariate Hermite polynomials. This decomposition is given by the following formula (Chilès & Delfiner 2012)

$$I\{y \leq Y(x) \leq y'\} = \big(G(y') - G(y)\big) + \sum_{i=1}^{+\infty} \frac{\big(\chi_{i-1}(y')g(y') - \chi_{i-1}(y)g(y)\big)}{\sqrt{i}} \chi_i\big(Y(x)\big) \quad \text{(B.14)}$$

B.3 Transformation function for facies proportion

Cartier's relation (4.10) leads to an explicit expression for $\psi_{v,i}\,(y_1, y_2)$ through $\varphi_i(y_1, y_2)$. Here, we recall the following property of normalized univariate Hermite polynomials from (Chilès & Delfiner 2012): if $(Y, Y') \sim \mathrm{B}(\rho)$ and $Y'' \sim G(\rho y, 1-\rho^2)$ then

$$E[\chi_n(Y')|Y = y] = E[\chi_n(Y'')] = \rho^n \chi_n(y) \tag{B.15}$$

Using (B.15), from Cartier's relation (4.10) one derives

$$\begin{aligned} &E\left[I\{Y_1(\underline{x}) \in \Delta t_i^1, Y_2(\underline{x}) \in \Delta t_i^2\} \middle| (Y_v^1, Y_v^2) = (y_1, y_2)\right] \\ &= E\left[\Sigma_{n=0}^{+\infty}\Sigma_{m=0}^{n}\varphi(i,n,m)\chi_{n-m}\left(Y_1(\underline{x})\right)\chi_m\left(Y_2(\underline{x})\right) \mid (Y_v^1, Y_v^2) = (y_1, y_2)\right] \\ &= \Sigma_{n=0}^{+\infty}\Sigma_{m=0}^{n}\varphi(i,n,m)E[\chi_{n-m}(\xi_1)\chi_m(\xi_2)] \\ &= \Sigma_{n=0}^{+\infty}\Sigma_{m=0}^{n}\varphi(i,n,m)E[\chi_{n-m}(\xi_1)]\,E[\chi_m(\xi_2)] \\ &= \Sigma_{n=0}^{+\infty}\Sigma_{m=0}^{n}\varphi(i,n,m)r_1^{n-m}r_2^{m}\chi_{n-m}(y_1)\chi_m(y_2), \end{aligned} \tag{B.16}$$

where $\xi_i \sim G(r_i y_i, 1-r_i^2), \xi_1 \perp \xi_2$. Comparing (B.16) with (4.9) gives the desired decomposition coefficients in (4.11).

B.4 Modeling the indicator transformation functions

When a bivariate indicator functions $\varphi_i(y_1, y_2) = I\{y_1 \in \Delta t_i^1, y_2 \in \Delta t_i^2\} = I\{y_1 \in \Delta t_i^1\}\, I\{y_2 \in \Delta t_i^2\}$ can be factorized with respect to y_1 and y_2, the Cartier's relation can be simplified

$$\begin{aligned} \psi_{v,i}\,(y_1, y_2) = &\; E\left[I\{Y_1(\underline{x}) \in \Delta t_i^1, Y_2(\underline{x}) \in \Delta t_i^2\} \middle| (Y_v^1, Y_v^2) = (y_1, y_2)\right] \\ &= \int_R\int_R \varphi_i\left(r_1 y_1 + \sqrt{1-r_1^2}\tau_1, r_2 y_2\right. \\ &\left. + \sqrt{1-r_2^2}\tau_2\right) g(\tau_1)g(\tau_2)d\tau_1 d\tau_2 \\ &= \int_R I\left\{r_1 y_1 + \sqrt{1-r_1^2}\tau_1 \in \Delta t_i^1\right\} g(\tau_1)\, d\tau_1 \\ &\times \int_R I\left\{r_2 y_2 + \sqrt{1-r_2^2}\tau_2 \in \Delta t_i^2\right\} g(\tau_2)\, d\tau_2 \\ &= I^1(y_1, r_1) \times I^2(y_2, r_2) \end{aligned} \tag{B.17}$$

Since univariate indicator functions are piecewise-linear, analytical expressions for $I^1(y_1, r_1)$ and $I^2(y_2, r_2)$ can be found with Appendix A. Let $\Delta t_i^1 = [\delta_1^l, \delta_1^r], \Delta t_i^2 = [\delta_2^l, \delta_2^r]$ and $\tilde{x}(r, y) = \frac{x - ry}{\sqrt{1-r^2}}$. Then for $q = 1,2$

$$I^q(y_q, r_q) = \left(G\left(\widetilde{\delta_q^r}(r_q, y_q)\right) - G\left(\widetilde{\delta_q^l}(r_q, y_q)\right)\right) \quad \text{(B.18)}$$

This equation demonstrate that for modeling the transformation functions of bivariate indicators required by PG-DGM it is possible to avoid using decompositions in Hermite polynomial basis.

B.5 Proof of proposition 1

Let us prove that the facies proportions modeled with PG-DGM always sum up to 1 for every block. It is sufficient to show that $\Sigma_{i=1}^{k}\psi_{v,i}(y_1, y_2) = 1$ for any y_1 and y_2. Indeed, we know that $\Sigma_{i=1}^{k} I_i(x) = 1$ which, expressed in bivariate Hermite polynomials, means

$$\Sigma_{n=0}^{+\infty}\Sigma_{m=0}^{n}\Sigma_{i=1}^{k}[\varphi(i,n,m)]\chi_{n,m}\big(Y_1(x), Y_2(x)\big) = 1. \tag{B.19}$$

Let us consider this expression at $Y_1(x) = y_1, Y_2(x) = y_2$ and take a scalar product of both parts with $\chi_{\tilde{n},\tilde{m}}(y_1, y_2) \neq 1$. Due to orthogonality, we have

$$\Sigma_{i=1}^{k}\varphi(i,\tilde{n},\tilde{m}) = 0 \tag{B.20}$$

Now, if we consider $\Sigma_{i=1}^{k}\psi_{v,i}(y_1, y_2)$

$$\begin{aligned}\Sigma_{i=1}^{k}\psi_{v,i}(y_1, y_2) = \ & \Sigma_{n=0}^{+\infty}\Sigma_{m=0}^{n}\Sigma_{i=1}^{k}[\varphi(i,n,m)]r_1^{n-m}r_2^{m}\chi_{n,m}(y_1, y_2) \\ = \ & 1 + \Sigma_{n=1}^{+\infty}\Sigma_{m=0}^{n} 0 \times r_1^{n-m}r_2^{m}\chi_{n,m}(y_1, y_2) = 1,\end{aligned} \tag{B.21}$$

which proves the statement.

B.6 Derivation of the change of support coefficients

Let us consider the theoretical variance of $P_i(v)$ - proportion of facies i over the block. From the problem statement of facies simulations on an unstructured grid, the value of $Var\big(P_i(v)\big)$ is defined through the integral of the covariance function $C_{ii}(x,x')$ of $I_i(x)$.

$$\begin{aligned} Var\big(P_i(v)\big) &= \frac{1}{|v|^2}\int_v\int_v cov\,(I_i(x), I_i(x'))dxdx' \\ &= \frac{1}{|v|^2}\int_v\int_v C_{ii}(x,x')dxdx' =^{def} \overline{C}_{ii}(v) \\ &= \frac{1}{|v|^2}\int_v\int_v \Sigma_{n=1}^{+\infty}\Sigma_{m=0}^{n}\varphi^2(i,n,m)\rho_1^{n-m}\,(x,x')\rho_2^{m}\,(x,x')\,dxdx' \end{aligned} \tag{B.22}$$

On the other hand due to the representation (4.11)

$$\begin{aligned} P_i(v) = \psi_{v,i}(Y_v^1, Y_v^2) \Rightarrow Var\big(P_i(v)\big) &= \Sigma_{n=2}^{+\infty}\Sigma_{m=1}^{n-1}\psi^2(v,i,n,m) \\ &= \Sigma_{n=1}^{+\infty}\Sigma_{m=0}^{n}\varphi^2(i,n,m)r_1^{2(n-m)}r_2^{2m} \end{aligned} \tag{B.23}$$

If we want to reproduce the variance of every proportion within the block, for any $i = 1 \dots K$ the following identity should take place

$$Var\big(P_i(v)\big) = \frac{1}{|v|^2}\int_v\int_v C_{ii}(x,x')dxdx' = \Sigma_{n=1}^{+\infty}\Sigma_{m=0}^{n}\varphi^2(i,n,m)r_1^{2(n-m)}r_2^{2m} \tag{B.24}$$

Using (B.22), (B.23) and (B.24), the following system of K equations for two variables r_1 and r_2 is obtained

$$\begin{aligned} &\Sigma_{n=1}^{+\infty}\Sigma_{m=0}^{n}\varphi^2(i,n,m)\left\{\frac{1}{|v|^2}\int_v\int_v \rho_1^{n-m}\,(x,x')\rho_2^{m}\,(x,x')\,dxdx' - r_1^{2(n-m)}r_2^{2m}\right\} \\ &\qquad = 0 \qquad \forall i = 1 \dots K \end{aligned} \tag{B.25}$$

Obviously, in the general case the solution to system (B.25) does not exist. The best thing which can be done in this case is trying to solve the following optimization problem in order to minimize the error of the covariance reproduction

$$\sum_{k=1}^{K}\Big[\,\bar{C}_{ii}(v,v) - \Sigma_{n=1}^{+\infty}\Sigma_{m=0}^{n}\varphi^2(i,n,m)r_1^{2(n-m)}r_2^{2m}\Big]^2 \to min. \tag{B.26}$$

Optimization problem (B.26) can be solved with any classical method and we do not detail this approach further here. Instead of really solving the optimization problem, an approximation for solution of (B.26) can be found under certain assumptions. Notice that the system of K equations (B.25) has a solution if the following infinite system of equations has a solution relative to (r_1, r_2). For $\forall n = 1 \dots +\infty, m = 0 \dots n$

$$\frac{1}{|v|^2}\int_v\int_v \rho_1^{n-m}(x,x')\rho_2^m(x,x')\,dxdx' - r_1^{2(n-m)}r_2^{2m} = 0\,. \tag{B.27}$$

Obviously, it is not possible to find the exact solution to this infinite system, but we can consider the most significant part of this infinite system of equations is "concentrated" in equations with indices $(n,m) = (1,0),(1,1)$. In this case we have two equations ($\overline{hat}$ on top of the function denotes double volumetric integral over the given volumes normalized by the volume squared)

$$\begin{cases} r_1^2 = \overline{\rho_1}(v,v) \\ r_2^2 = \overline{\rho_2}(v,v)' \end{cases} \tag{B.28}$$

which coincides with (4.14).

Choosing (r_1, r_2) as a solution of (B.25) we approximately reproduce the variance of all proportions inside a block.

B.7 Derivation of the block to block covariance

Consider two blocks v_p and v_q. Let us determine the covariance between $Y_{v_p}^1$ and $Y_{v_q}^1$ and between $Y_{v_p}^2$ and $Y_{v_q}^2$. From the problem statement of facies simulation on unstructured grids

$$\begin{aligned} cov\left(P_i(v_p), P_j(v_q)\right) &= \overline{C_{ij}}(v_p, v_q) = \frac{1}{|v_p||v_q|}\int_{v_p}\int_{v_q} cov\left(I_i(x), I_j(x')\right)dxdx' \\ &= \frac{1}{|v_p||v_q|}\int_{v_p}\int_{v_q} \Sigma_{n=1}^{+\infty}\Sigma_{m=0}^{n}\varphi(i,n,m)\varphi(j,n,m)\rho_1^{n-m}(x,x')\rho_2^m(x,x')\,dxdx' \end{aligned} \tag{B.29}$$

On the other hand, from the decomposition into Hermite polynomials

$$\begin{aligned} &cov\left(P_i(v_p), P_j(v_q)\right) = cov\left(\psi_{v_p,i}\left(Y_{v_p}^1, Y_{v_p}^2\right), \psi_{v_q,j}\left(Y_{v_q}^1, Y_{v_q}^2\right)\right) \\ &= \Sigma_{n=1}^{+\infty}\Sigma_{m=0}^{n}\psi(v_p,i,n,m)\psi(v_q,j,n,m)cov^{n-m}(Y_{v_p}^1, Y_{v_q}^1)cov^m(Y_{v_p}^2, Y_{v_q}^2) \end{aligned} \tag{B.30}$$

In this case we have K^2 equations (B.25) for $i = 1 \ldots K, j = 1 \ldots K$. Since (B.29) should be equal to (B.25), in order to determine $cov(Y_{v_p}^1, Y_{v_q}^1)$ and $cov(Y_{v_p}^2, Y_{v_q}^2)$ an optimization problem similar to (B.26) should be solved. Here we only give an approximate solution to this problem in the same style as it was done in previous section:

$$\begin{aligned} &\Sigma_{n=1}^{+\infty}\Sigma_{m=0}^{n}\varphi(i,n,m)\varphi(j,n,m)\left\{\overline{\rho_1^{n-m}\rho_2^m}(v_p,v_q)\right. \\ &\left. - r_{1,p}^{n-m}r_{1,q}^{n-m}\,r_{2,p}^m r_{2,q}^m cov^{n-m}\left(Y_{v_p}^1, Y_{v_q}^1\right)cov^m\left(Y_{v_p}^2, Y_{v_q}^2\right)\right\} = 0 \end{aligned} \tag{B.31}$$

The system (B.31) can be satisfied in case if for $\forall n = 1..+\infty, m = 0 \ldots n$ the following identity takes place

$$\overline{\rho_1^{n-m}\rho_2^m}(v_p, v_q) - r_{1,p}^{n-m} r_{1,q}^{n-m} r_{2,p}^m r_{2,q}^m cov^{n-m}\left(Y_{v_p}^1, Y_{v_q}^1\right) cov^m\left(Y_{v_p}^2, Y_{v_q}^2\right) = 0 \qquad \text{(B.32)}$$

Consider the part of this infinite system of equations for $(n, m) = (1,0), (1,1)$. From these two equations, one derives

$$\begin{cases} r_{1,p} r_{1,q} cov\left(Y_{v_p}^1, Y_{v_q}^1\right) = \overline{\rho_1}(v_p, v_q) \\ r_{2,p} r_{2,q} cov\left(Y_{v_p}^2, Y_{v_q}^2\right) = \overline{\rho_2}(v_p, v_q) \end{cases}, \qquad \text{(B.33)}$$

which is equivalent to (4.15).

B.8 Mini-model tests for PG-DGM

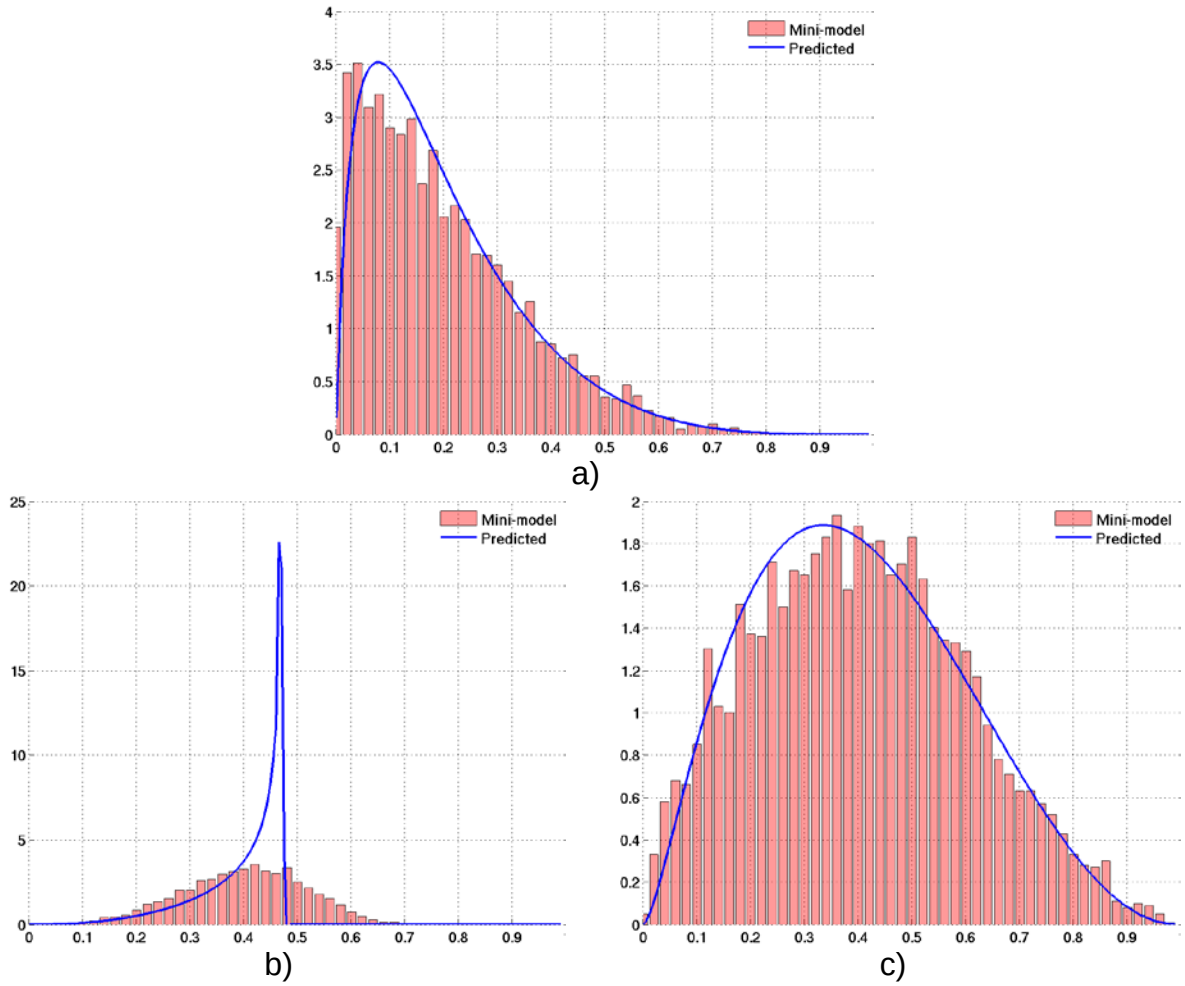


Figure B-1. Flag 1 Medium range. a) Distribution of $P_0(v)$ b) Distribution of $P_1(v)$ c) Distribution of $P_2(v)$

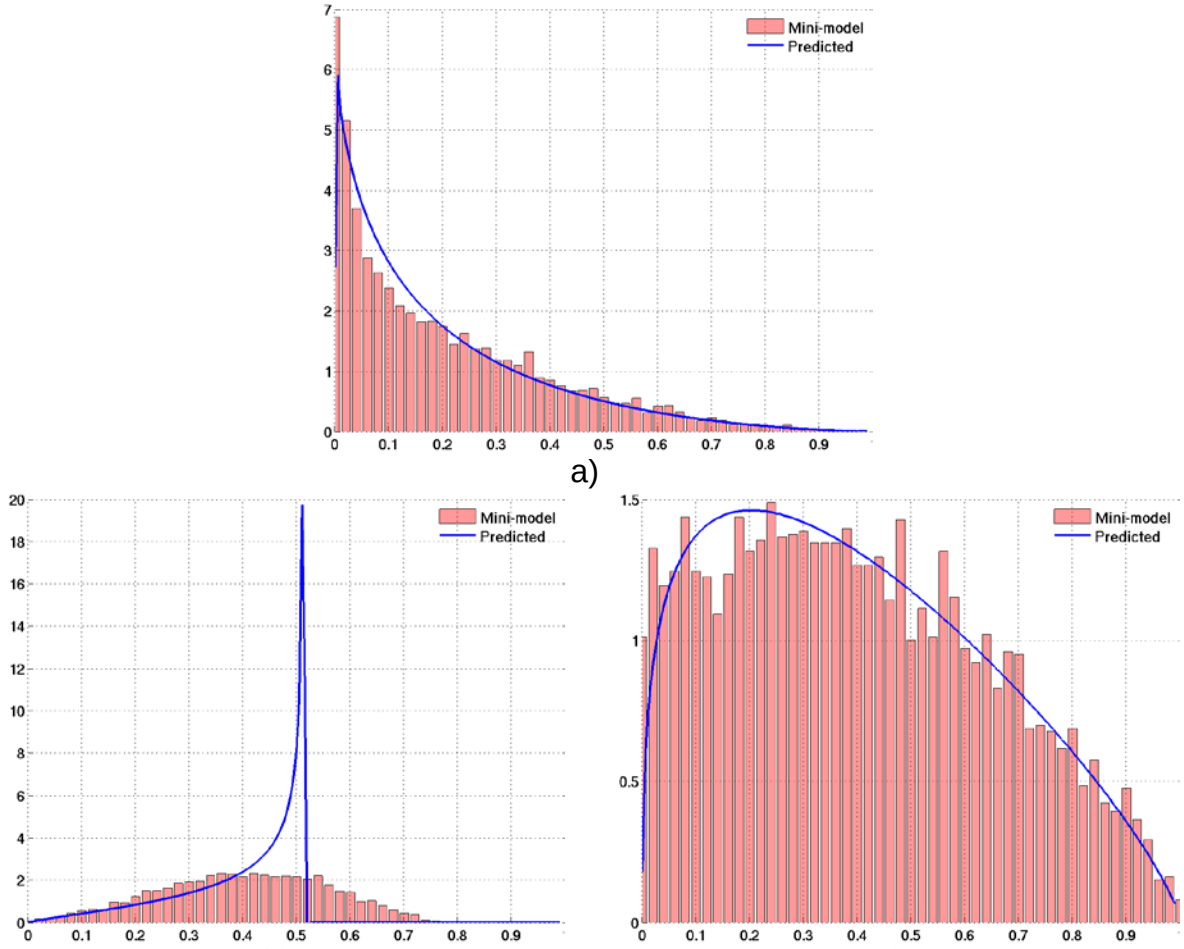


Figure B-2. Flag 1 Long range. a) Distribution of $P_0(v)$ b) Distribution of $P_1(v)$ c) Distribution of $P_2(v)$

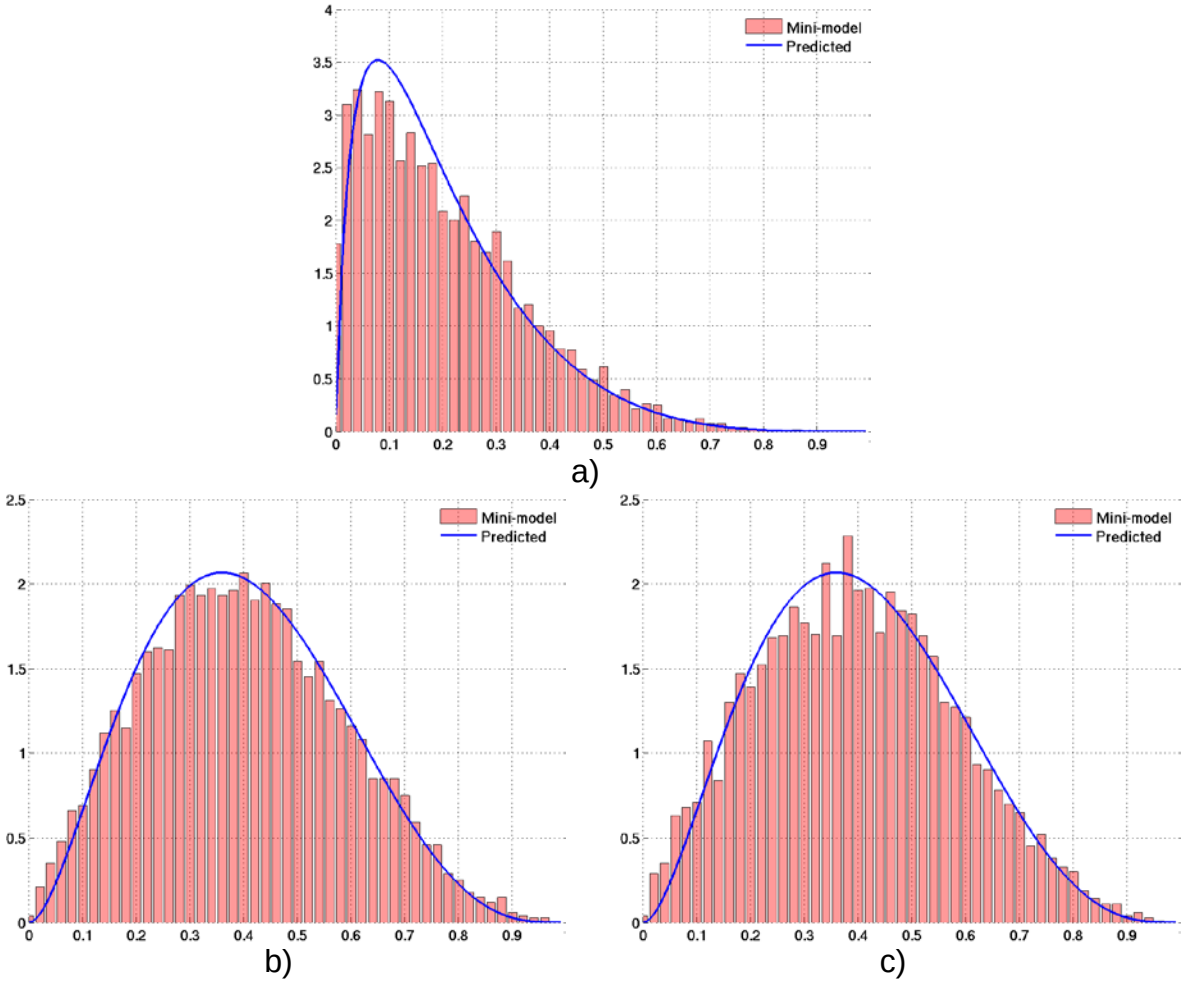
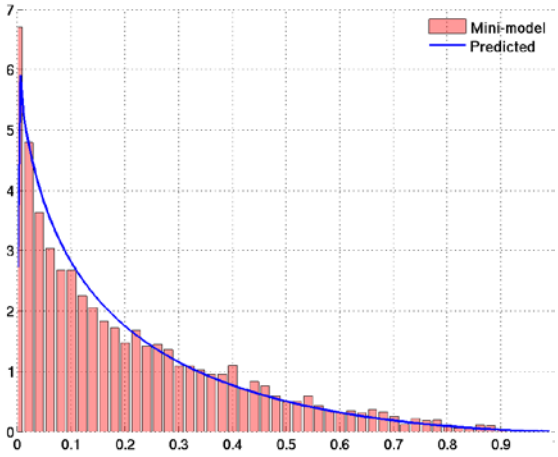
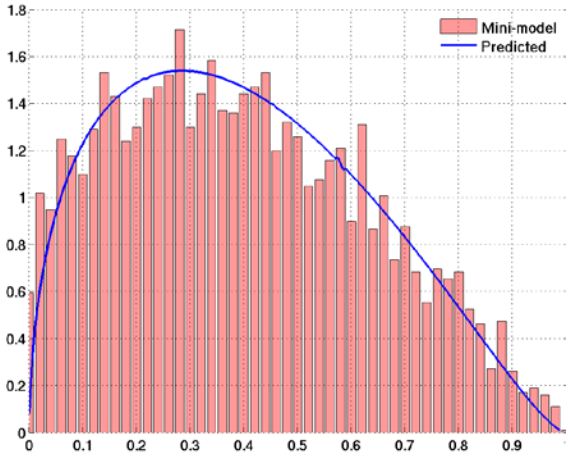


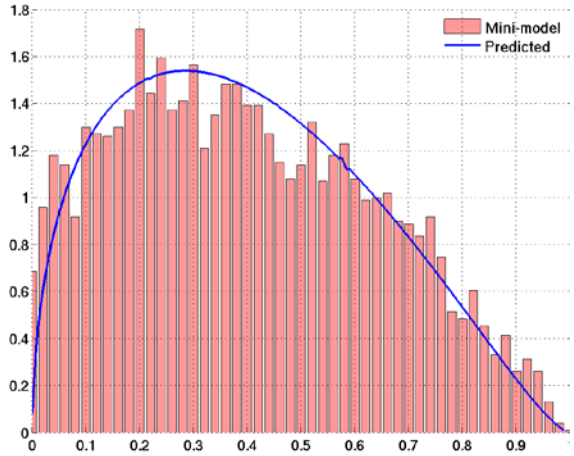
Figure B-3. Flag 2 Medium range. a) Distribution of $P_0(v)$ b) Distribution of $P_1(v)$ c) Distribution of $P_2(v)$



a)



b)



c)

Figure B-4. Flag 2 Long range. a) Distribution of $P_0(v)$ b) Distribution of $P_1(v)$ c) Distribution of $P_2(v)$

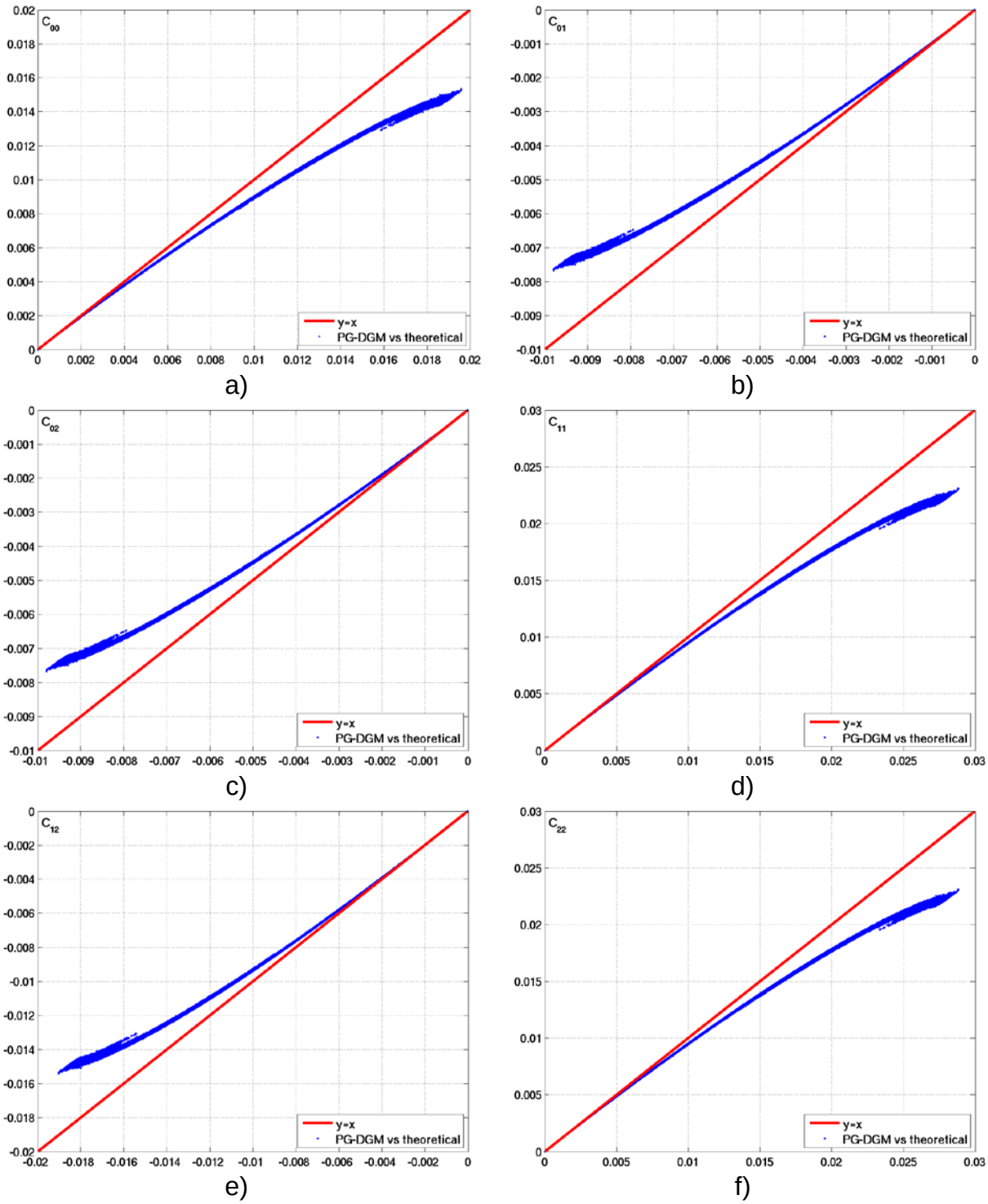
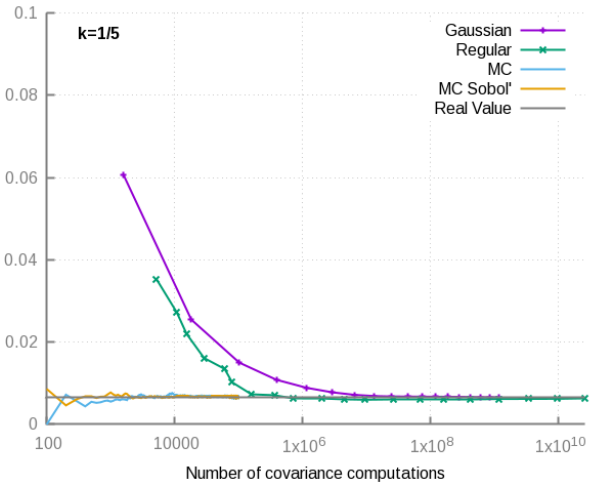
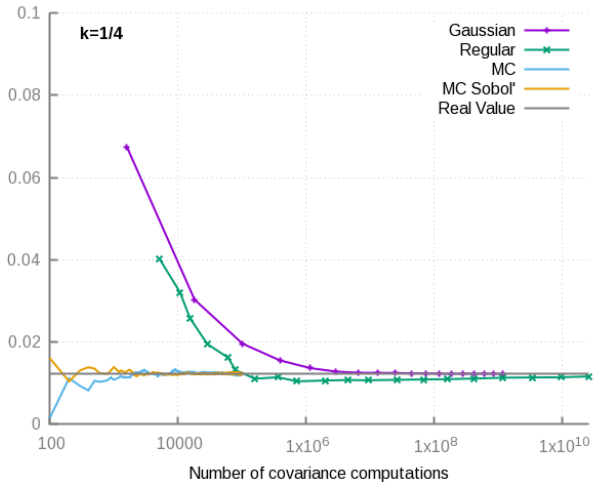


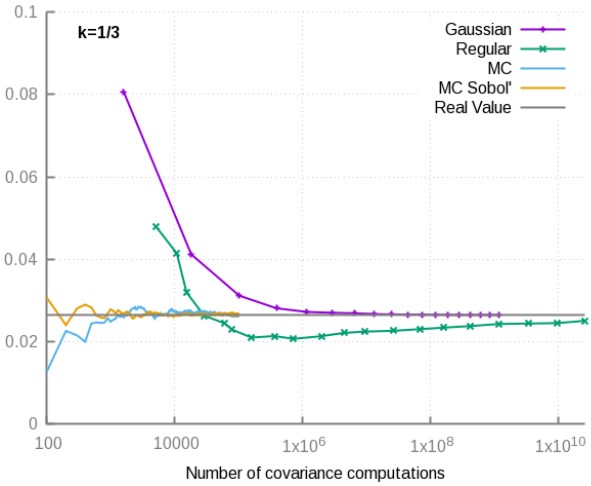
Figure B-5. Approximation of block to block covariance implied by PG-DGM. a)C_{00} b) C_{01} c) C_{02} d) C_{11} e) C_{12} f) C_{22}.

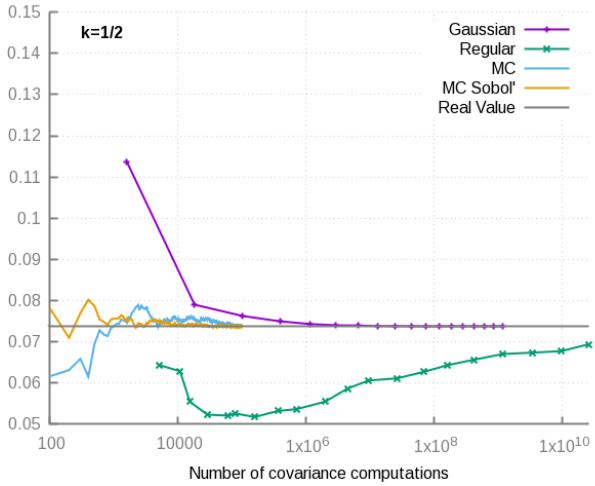
Appendix C: Variance computation sensitivity tests

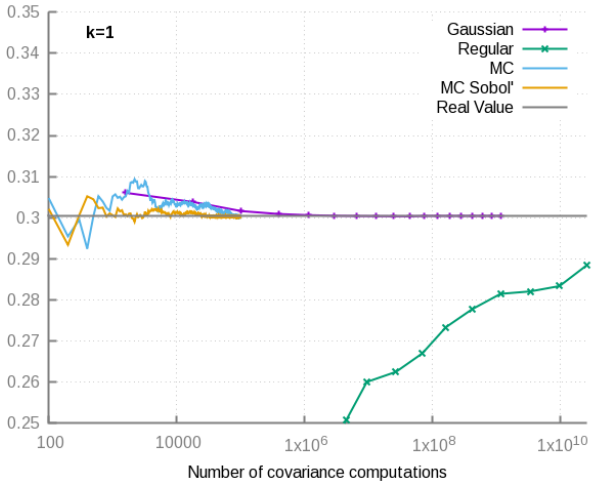
C.1 Spherical covariance











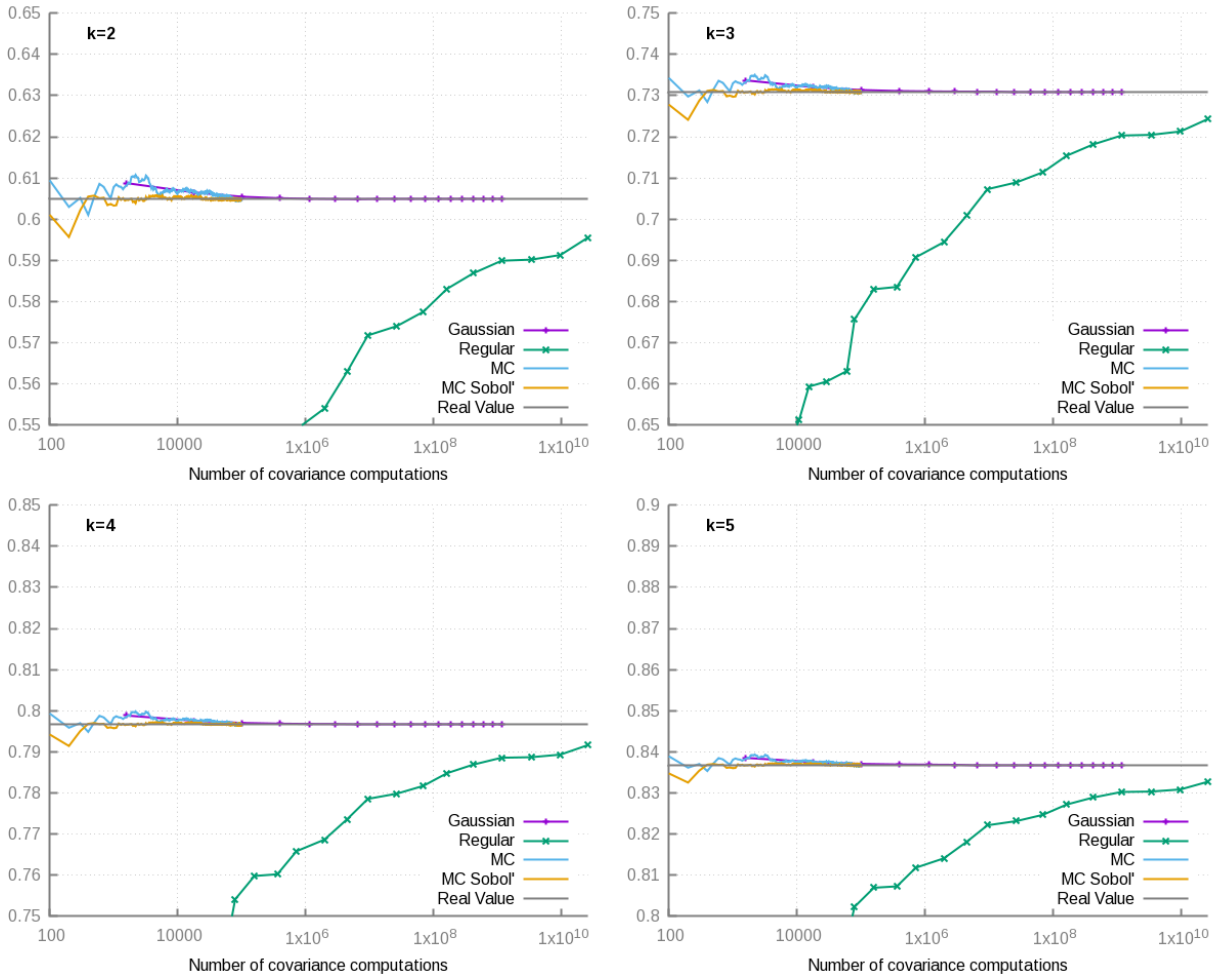
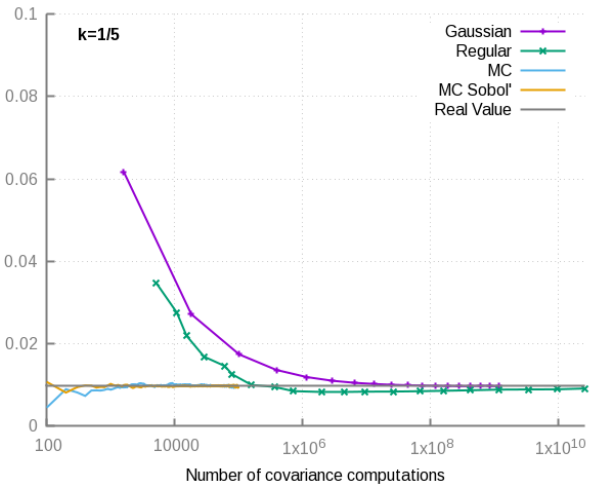
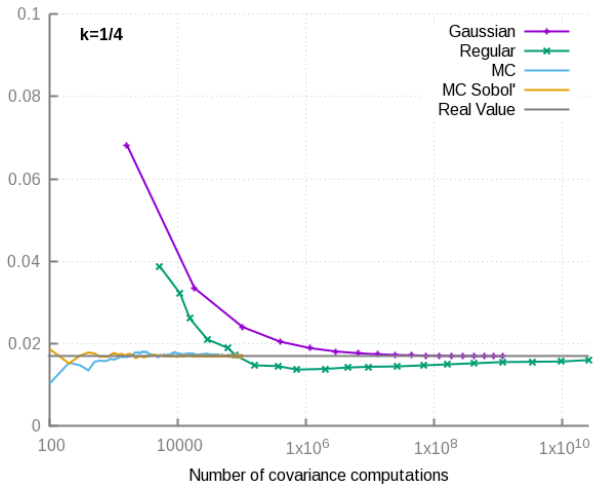
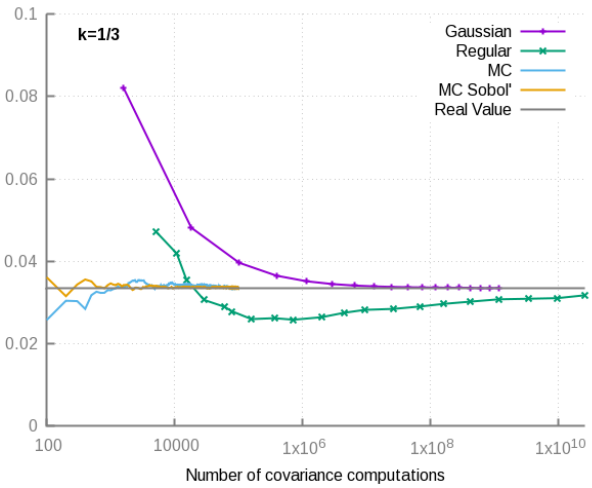


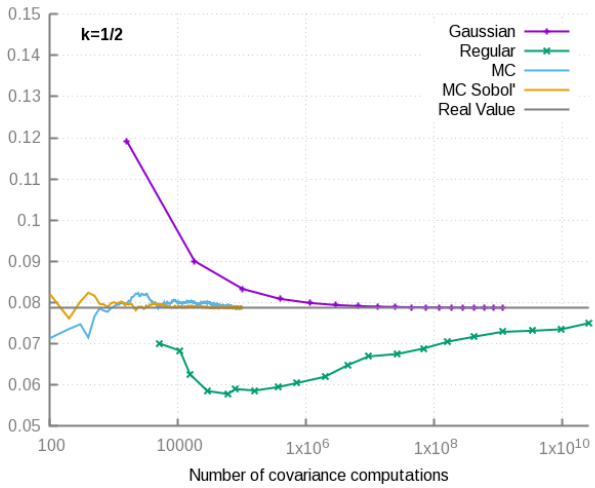
Figure C-1. Comparing the accuracy of different numerical methods for computing the theoretical variance of block average value $Var(Z(v))$ for various ranges of input covariance.

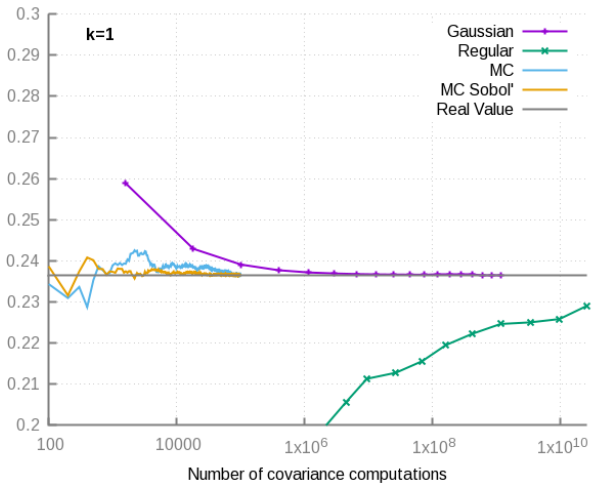
C.2 Exponential covariance











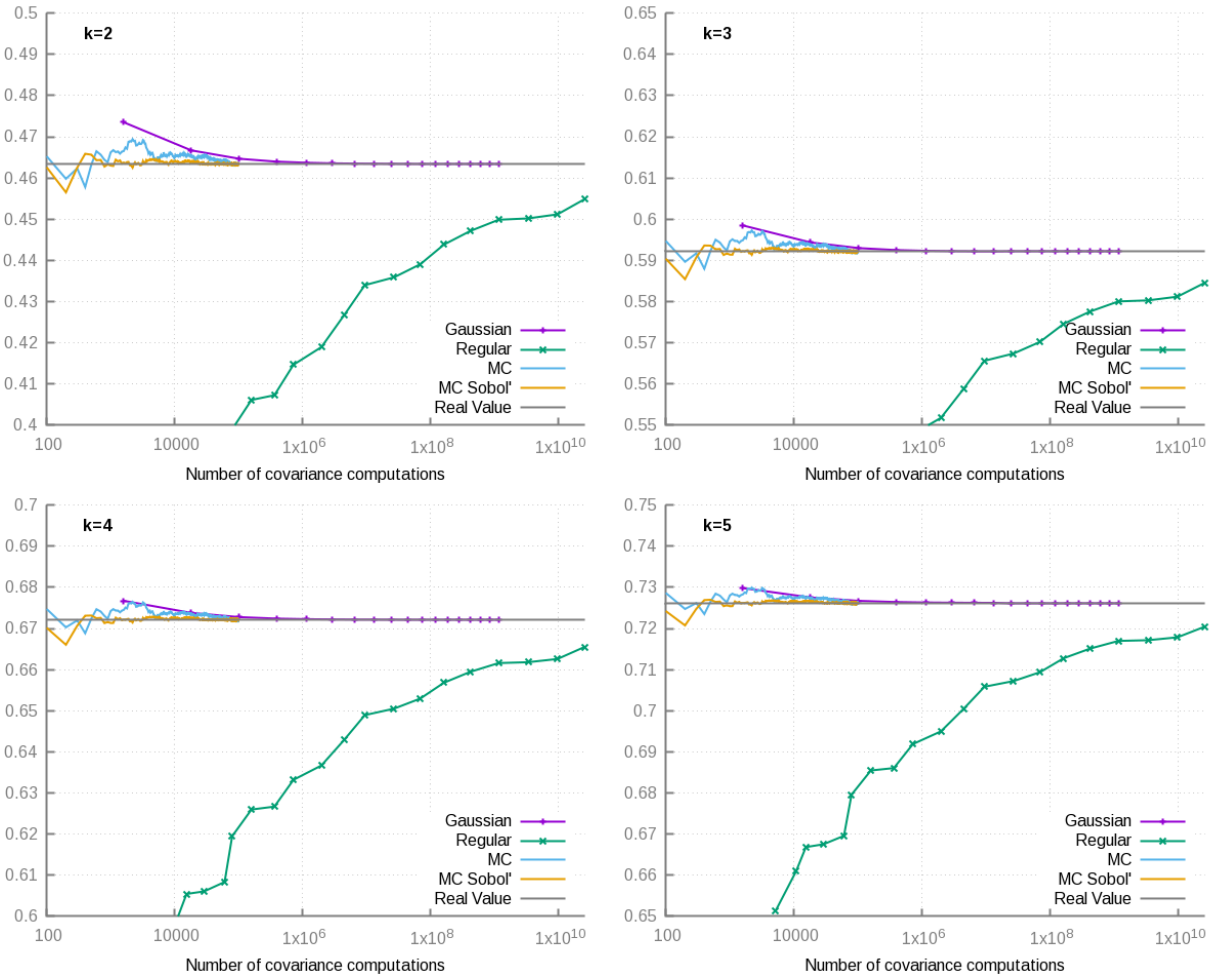
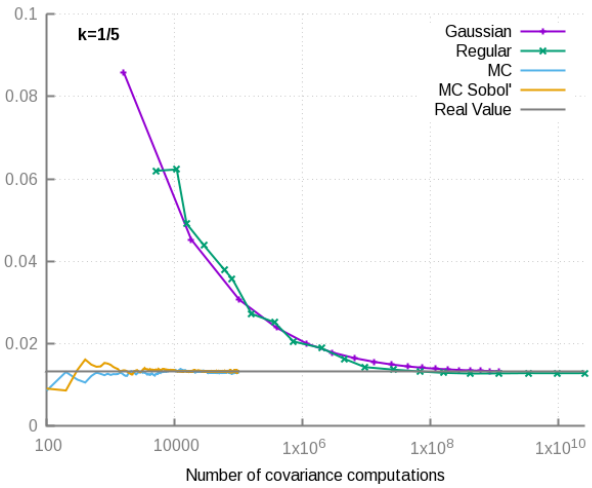
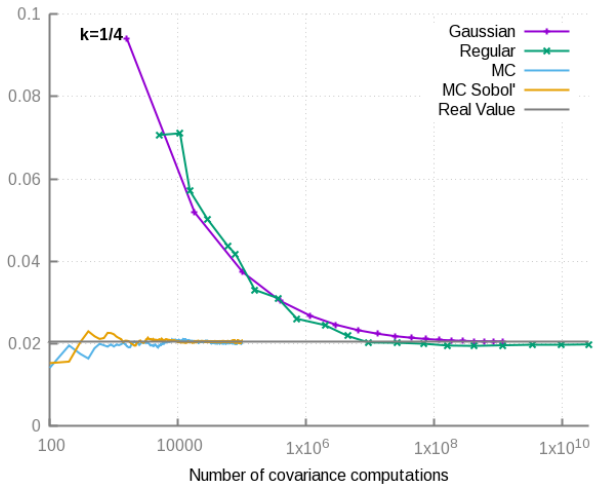
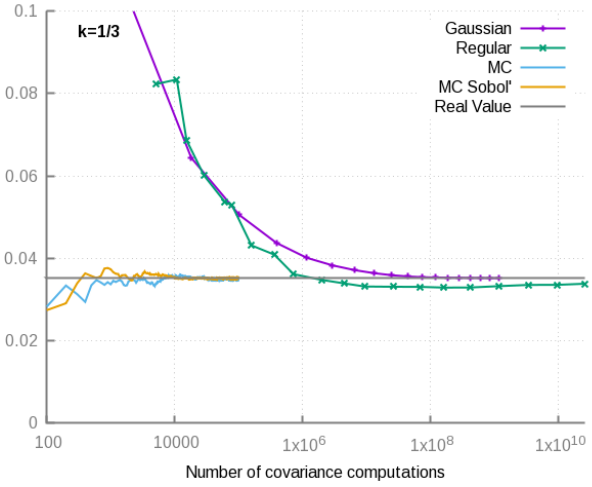


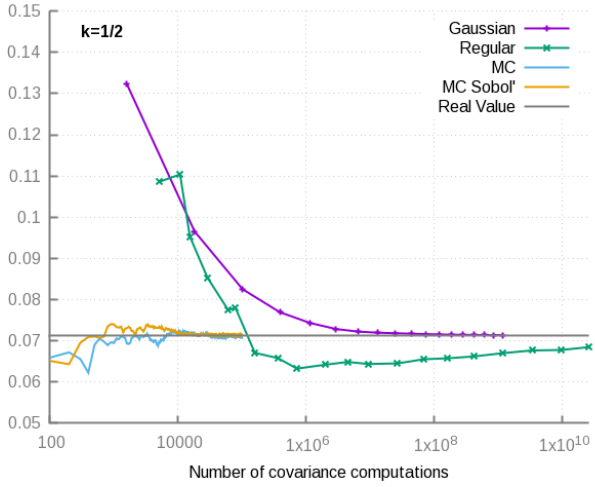
Figure C-2. Comparing the accuracy of different numerical methods for computing the theoretical variance of block average value $Var(Z(v))$ for various ranges of input covariance.

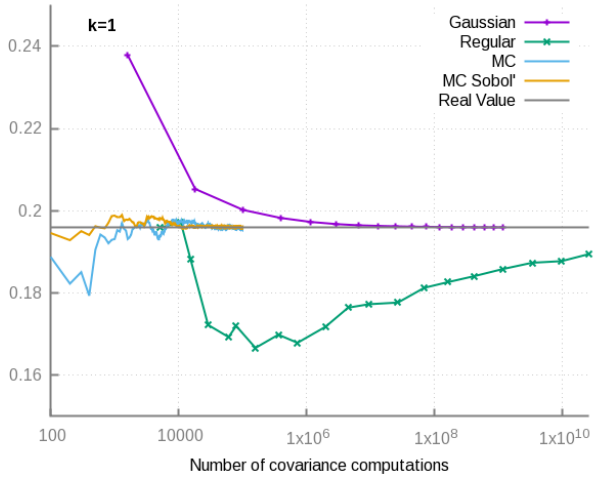
C.3 Double structure covariance











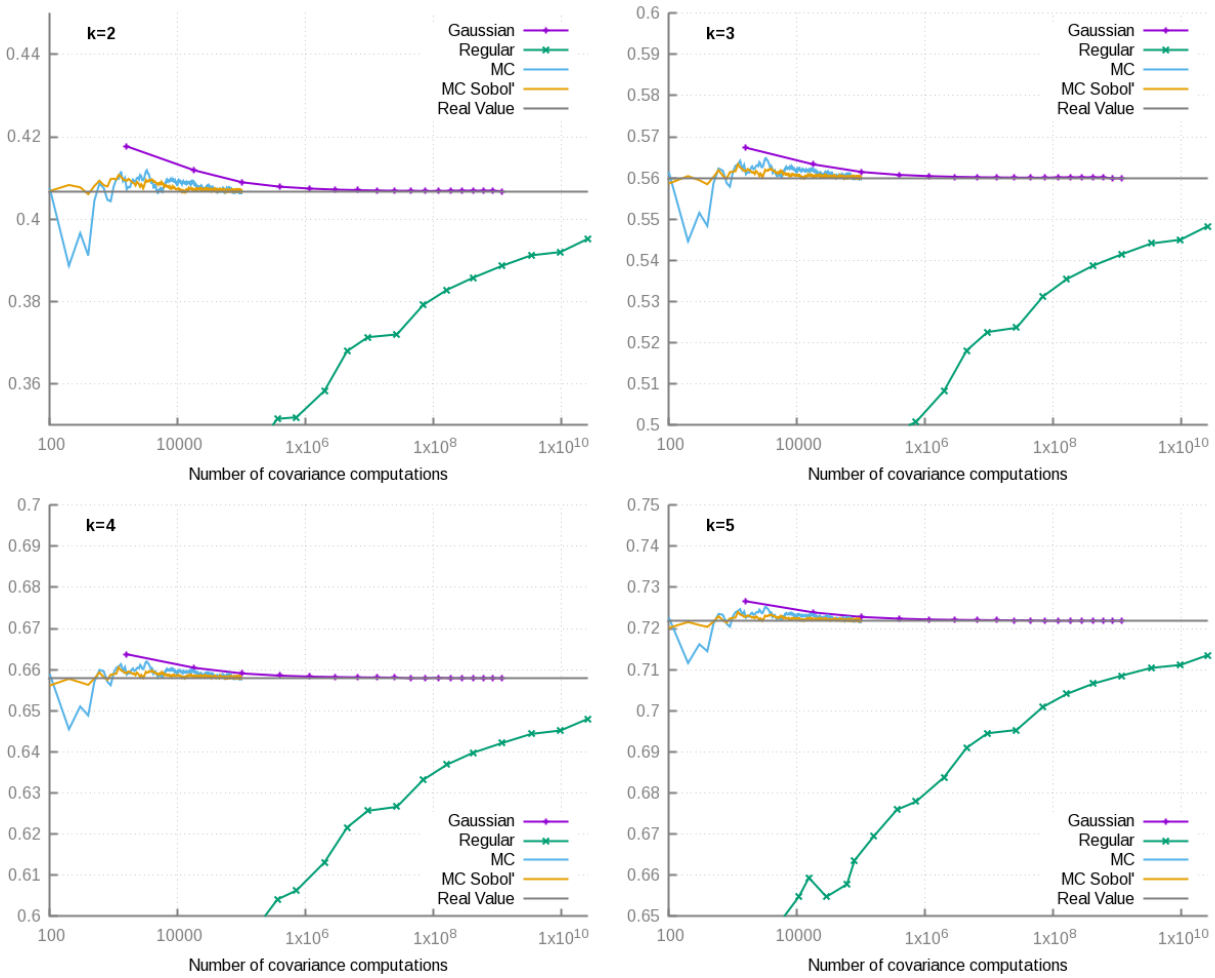
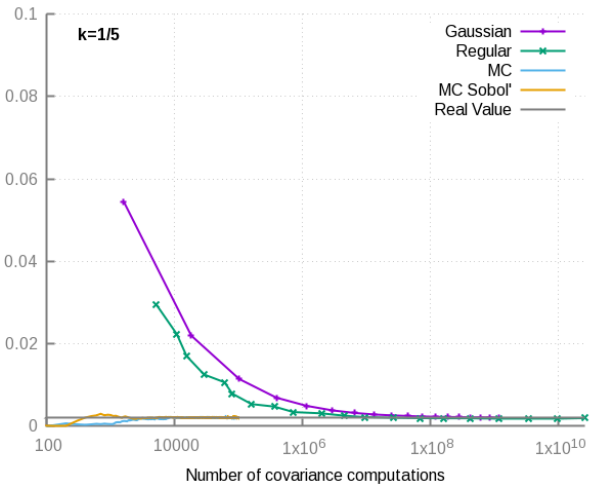
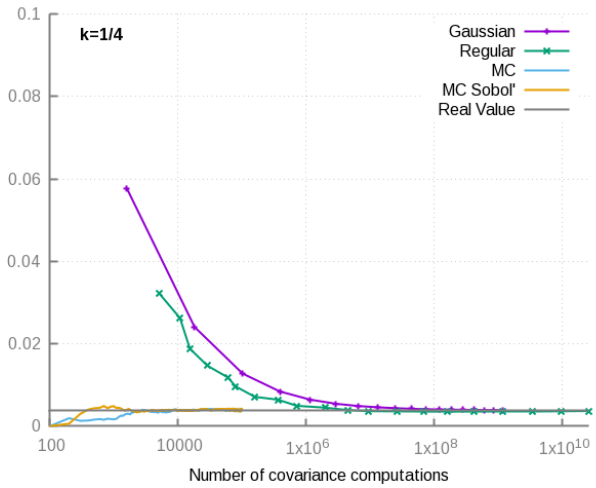
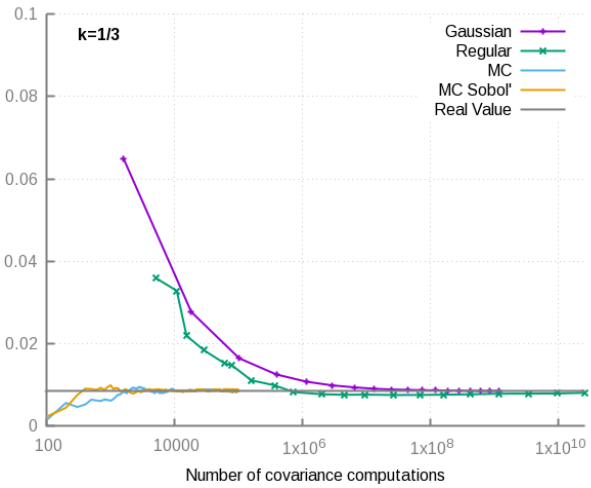


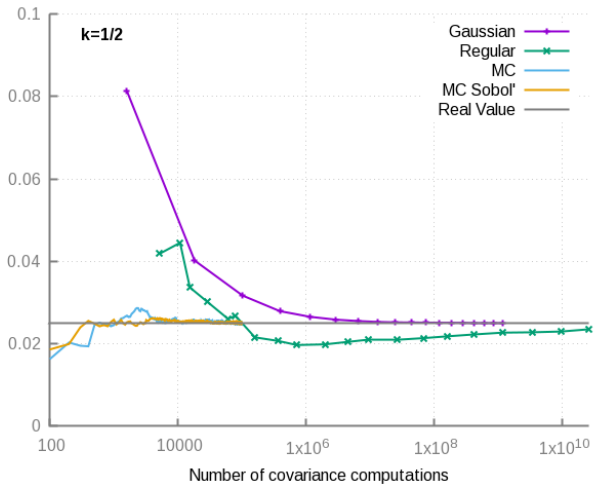
Figure C-3. Comparing the accuracy of different numerical methods for computing the theoretical variance of block average value $Var(Z(v))$ for various ranges of input covariance.

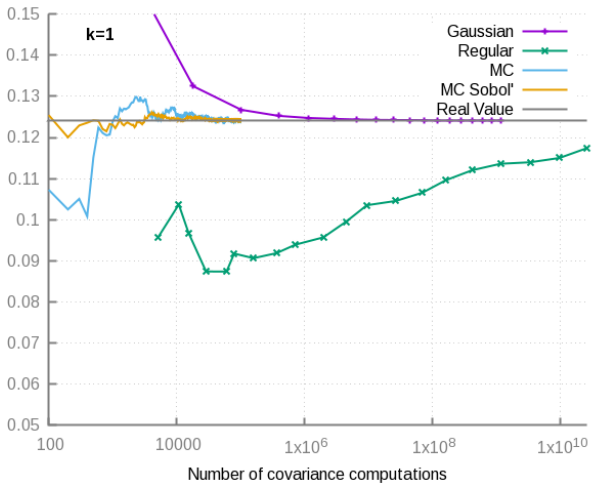
C.4 Spherical covariance with azimuth











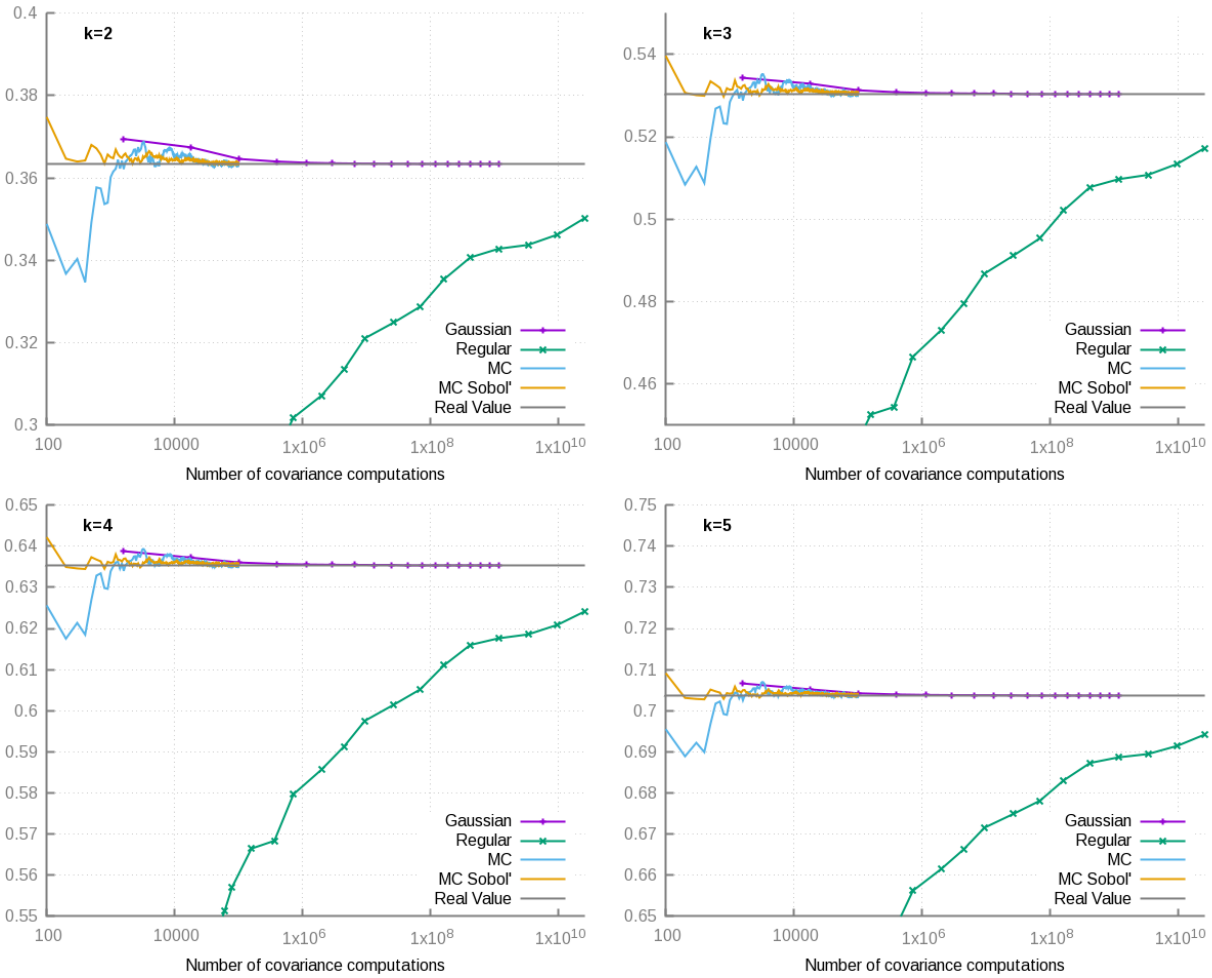


Figure C-4. Comparing the accuracy of different numerical methods for computing the theoretical variance of block average value $Var(Z(v))$ for various ranges of input covariance.

References

Alabert F. (1989). Non-Gaussian data expansion in the earth sciences. Terra Nova, 1, 123-134.

Antonov IA, Saleev V. (1979). An economic method of computing LP τ-sequences. USSR Computational Mathematics and Mathematical Physics, 19, 252-256.

Babak O, Deutsch CV. (2008). Two new approaches to avoid large kriging weights to end samples in strings of data. In: Ortiz JM, Emery X (eds), Geostats2008, Proceedings of the 8th International Geostatistics Congress. FCFM—Gecamin, Santiago, Chile, pp. 369-378.

Bentley JL. (1975). Multidimensional binary search trees used for associative searching. Communications of the ACM, 18, 509-517.

Boisvert JB, Deutsch CV. (2010). Geostatistics with locally varying anisotropy. University of Alberta.

Boucher A, Dimitrakopoulos R. (2009). Block simulation of multiple correlated variables. Mathematical Geosciences, 41, 215-237.

Brown G, Ferreira J, Lantuéjoul C. (2008). Conditional simulation of a Cox process using multiple sample supports. In: Ortiz JM, Emery X (eds), 8th International Geostatistics Congress. Gecamin, Santiago, Chile, pp. 459-468.

Chawla M, Jain M. (1968). Asymptotic error estimates for the Gauss quadrature formula. Mathematics of computation, 22, 91-97.

Chilès J-P. (2014). Validity range of the discrete Gaussian change-of-support model and its variant. Journal of the Southern African Institute of Mining and Metallurgy, 114, 231-235.

Chilès J-P, Delfiner P. (2012). Geostatistics: Modeling Spatial Uncertainty, 2nd Edition. Wiley, Hoboken.

Chilès J-P, Lantuéjoul C. (2005). Prediction by conditional simulation: models and algorithms. In: Space, Structure and Randomness. Springer, pp. 39-68.

Chugunova TL, Hu LY. (2008). Multiple-point simulations constrained by continuous auxilliary data. Mathematical Geosciences, 40, 133-146.

Cressie N, Wikle CK. (2011). Statistics for Spatio-Temporal Data. Wiley, Hoboken.

Davis MW. (1987). Production of conditional simulations via the LU triangular decomposition of the covariance matrix. Mathematical Geology, 19, 91-98.

de Fouquet C. (1994). Reminders on the conditioning kriging. In: Geostatistical Simulations. Springer Netherlands, pp. 131-145.

De Lucia M, De Fouquet C, Lagneau V, Bruno R. (2009). Equivalent block transmissivity in an irregular 2D polygonal grid for one-phase flow: A sensitivity analysis. Comptes Rendus Geoscience, 341, 327-338.

Desbarats A, Dimitrakopoulos R. (2000). Geostatistical simulation of regionalized pore-size distributions using min/max autocorrelation factors. Mathematical Geology, 32, 919-942.

Deutsch CV, Journel AG. (1992). Geostatistical software library and user's guide. New York, 119, 147.

Elliott D, Johnston PR, Johnston BM. (2011). Estimates of the error in Gauss–Legendre quadrature for double integrals. Journal of Computational and Applied Mathematics, 236, 1552-1561.

Emery X. (2004a). Properties and limitations of sequential indicator simulation. Stochastic Environmental Research and Risk Assessment, 18, 414-424.

Emery X. (2004b). Testing the correctness of the sequential algorithm for simulating Gaussian random fields. Stochastic Environmental Research and Risk Assessment, 18, 401-413.

Emery X. (2007). On some consistency conditions for geostatistical change-of-support models. Mathematical Geology, 39, 205-223.

Emery X. (2009). Change-of-support models and computer programs for direct block-support simulation. Computers & Geosciences, 35, 2047-2056.

Emery X, Gálvez I. (2012). A plurigaussian model for simulating regionalized compositions. In: Geostatistics Oslo 2012. Springer, pp. 39-50.

Emery X, Ortiz JM. (2011). Two approaches to direct block-support conditional co-simulation. Computers & Geosciences, 37, 1015-1025.

Emery X, Peláez M. (2011). Assessing the accuracy of sequential Gaussian simulation and cosimulation. Computational Geosciences, 15, 673-689.

Farmer C. (2002). Upscaling: a review. International Journal for Numerical Methods in Fluids, 40, 63-78.

Friedman JH, Bentley JL, Finkel RA. (1977). An algorithm for finding best matches in logarithmic expected time. ACM Transactions on Mathematical Software (TOMS), 3, 209-226.

Galli A, Beucher H, Le Loc'h G, Doligez B. (1994). The pros and cons of the truncated Gaussian method. In: Geostatistical simulations. Springer, pp. 217-233.

Geman S, Geman D. (1984). Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images. IEEE Transactions on pattern analysis and machine intelligence, 721-741.

Golub GH, Welsch JH. (1969). Calculation of Gauss quadrature rules. Mathematics of computation, 23, 221-230.

Goovaerts P. (1997). Geostatistics for natural resources evaluation. Oxford University Press, Oxford.

Goovaerts P, Webster R, Dubois J-P. (1997). Assessing the risk of soil contamination in the Swiss Jura using indicator geostatistics. Environmental and ecological Statistics, 4, 49-64.

Gribov A, Krivoruchko K. (2004). Geostatistical mapping with continuous moving neighborhood. Mathematical Geology, 36, 267-281.

Gross H, Boucher A. (2015). Geostatistics on Unstructured Grid-Coordinate Systems, Connections and Volumes. In: Petroleum Geostatistics 2015, pp. 173-177.

Grünbaum B, Shephard GC. (1969). Convex polytopes. Bulletin of the London Mathematical Society, 1, 257-300.

Guardiano FB, Srivastava RM. (1993). Multivariate geostatistics: beyond bivariate moments. In: Geostatistics Troia'92. Springer, pp. 133-144.

Hajek B. (2015). Random Processes for Engineers. Cambridge University Press.

Hu L, Chugunova T. (2008). Multiple-point geostatistics for modeling subsurface heterogeneity: A comprehensive review. Water Resources Research, 44.

Journel AG. (1993). Modeling Uncertainty: Some Conceptual Thoughts. In: Geostatistics for the Next Century. Kluwer Academic Pub., Dordrecht, pp. 30-43.

Journel AG, Huijbregts CJ. (1978). Mining Geostatistics. Blackburn Press, Caldwell, New Jersey.

Kahaner D, Moler C, Nash S. (1989). Numerical methods and software. Englewood Cliffs: Prentice Hall, 1989, 1.

Korenblit M, Shmerling E. (2006). Algorithm and software for integration over a convex polyhedron. In: Mathematical Software-ICMS 2006. Springer, pp. 273-283.

Lantuejoul 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: The 9th International Geostatistics Congress, Oslo, Norway.

Leuangthong O, Khan KD, Deutsch CV. (2011). Solved problems in geostatistics. John Wiley & Sons.

Mallet J-L. (2014). Elements of mathematical sedimentary geology: The GeoChron model. EAGE publications.

Manchuk JG, Leuangthong O, Deutsch CV. (2005). Direct Geostatistical Simulation on Unstructured Grids. In: Geostatistics Banff 2004. Springer, Netherlands, pp. 85-94.

Mariethoz G, Caers J. (2014). Multiple-point geostatistics: stochastic modeling with training images. John Wiley & Sons.

Matheron G. (1967). Éléments pour une théorie des milieux poreux. Masson et Cie.

Matheron G. (1976a). Forecasting block grade distributions: the transfer functions. In: Advanced geostatistics in the mining industry. Springer, pp. 237-251.

Matheron G. (1976b). A simple substitute for conditional expectation: The disjunctive kriging. . In: Advanced Geostatistics in Mining Industry. Springer, pp. 221-236.

Matheron G. (1984). L'émergence de la loi de Darcy. Ann. Mines, 191, 11-16.

Matheron G. (1985). Change of Support for Diffusion-Type Random Functions. Journal of the International Association for Mathematical Geology, 17, 137-165.

Matheron G. (1989). Estimating and Choosing. Springer, Berlin.

Matheron G. (1993). Quelques inégalités pour la perméabilité effective d'un milieu poreux hétérogène. Cahier de géostatistique, 3, (Paris school of Mines publication), 1-20.

Matheron G, Beucher H, De Fouquet C, Galli A, Guerillot D, Ravenne C. (1987). Conditional simulation of the geometry of fluvio-deltaic reservoirs. In: SPE Annual Technical Conference and Exhibition. Society of Petroleum Engineers.

Matheron G, Beucher H, de Fouquet C, Galli A, Ravenne C. (1988). Simulation conditionnelle à trois faciès dans une lalaise de la formation du Brent. Sciences de la Terre, Série Informatique Géologique, 213-249.

Noetinger B, Haas A. (1996). Permeability averaging for well tests in 3D stochastic reservoir models. In: SPE Annual Technical Conference and Exhibition. Society of Petroleum Engineers.

Oz B, Deutsch CV, Tran T, Xie Y. (2003). DSSIM-HR: A FORTRAN 90 program for direct sequential simulation with histogram reproduction. Computers & Geosciences, 39-51.

Pardo-Iguzquiza E, Chica-Olmo M. (1993). The Fourier integral method: an efficient spectral method for simulation of random fields. Mathematical Geology, 25, 177-217.

Press WH, Teukolsky SA, Vetterling WT, Flannery BP. (1996). Numerical recipes in C. Cambridge university press Cambridge.

Rivoirard J. (1994). Introduction to Disjunctive Kriging and Non-Linear Geostatistics. Oxford University Press, Oxford.

Rivoirard J, Romary T. (2011). Continuity for kriging with moving neighborhood. Mathematical Geosciences, 43, 469-481.

Robertson RK, Mueller UA, Bloom LM. (2006). Direct sequential simulation with histogram reproduction: A comparison of algorithms. Computers & Geosciences, 32, 382-395.

Rue H, Held L. (2005). Gaussian Markov random fields: theory and applications. CRC Press.

Sarra SA. (2014). Regularized symmetric positive definite matrix factorizations for linear systems arising from RBF interpolation and differentiation. Engineering Analysis with Boundary Elements, 44, 76-86.

Shiryaev A. (1996). Probability, 2nd edition. Springer, New York.

Soares A. (2001). Direct sequential simulation and cosimulation. Mathematical Geology, 33, 911-926.

Sobol' IyM. (1967). On the distribution of points in a cube and the approximate evaluation of integrals. Zhurnal Vychislitel'noi Matematiki i Matematicheskoi Fiziki, 7, 784-802.

Srivastava MS, Carter EM. (1983). An introduction to applied multivariate statistics. North-Holland.

Stein ML. (2002). The screening effect in kriging. Annals of statistics, 298-323.

Stoer J, Bulirsch R. (2013). Introduction to numerical analysis. Springer Science & Business Media.

Strebelle S. (2002). Conditional simulation of complex geological structures using multiple-point statistics. Mathematical Geology, 34, 1-21.

Wackernagel H, Petitgas P, Touffait Y. (1989). Overview of methods for coregionalization analysis. In: Geostatistics. Springer, pp. 409-420.

Wu XH, Efendiev Y, Hou TY. (2002). Analysis of upscaling absolute permeability. Discrete and Continuous Dynamical Systems-Series B, 2.

Zaytsev V, Biver P, Wackernagel H, Allard D. (2015). Geostatistical Simulations on Irregular Reservoir Models Using Methods of Nonlinear Geostatistics. In: Petroleum Geostatistics 2015.

Zaytsev V, Biver P, Wackernagel H, Allard D. (2016). Change-of-Support Models on Irregular Grids for Geostatistical Simulation. Mathematical Geosciences, 48, 353-369. doi: 10.1007/s11004-015-9614-x.

Résumé

La simulation des phénomènes physiques exige souvent l'utilisation d'une discrétisation du milieu sous forme de maillage. Un exemple de ce type de situation est la simulation d'écoulement de fluides et la simulation du stress géomécanique pour les gisements pétroliers. Dans ces cas, le milieu étudié n'est pas homogène et l'hypothèse sur l'homogénéité de ce milieu peut mener à des résultats incorrects. C'est pourquoi la simulation des hétérogénéités est très importante pour ce genre de problèmes.

Cette thèse est consacrée à la simulation géostatistique des hétérogénéités sur les maillages non-structurés par les méthodes géostatistiques non-linéaires. Le but de cette thèse est la création d'algorithmes de simulation des hétérogénéités directement sur les maillages non-structurés, sans utiliser les maillages fins réguliers intermédiaires et de l'upscaling. On présente deux modèles théoriques pour les simulations des variables continues sur les maillages non-structurés qui sont les deux versions différentes du modèle Gaussien discret (DGM) - DGM 1 et DGM 2. Le modèle théorique utilisé dans cette thèse permet de convertir le problème de simulation sur un maillage non-structuré en un problème de simulation d'un vecteur Gaussien multivarié et l'application de fonctions de transformation adaptées pour chaque élément du vecteur. La simulation de faciès est aussi envisagée en utilisant une généralisation des modèles pluri-Gaussiens et Gaussien tronqués pour les maillages non-structurés.

L'application des méthodes développées est illustrée sur un gisement pétrolier - le cas d'étude X (gisement du gaz offshore).

Abstract

Simulations of physical phenomenon often require discretizing the medium with a mesh. An example of this type of simulation is the simulation of fluid flow through a porous medium and the evaluation of the geomechanical stress in the petroleum reservoir. The studied medium is often not homogeneous and applying a homogeneity hypothesis can lead to incorrect simulation results. That makes simulation of heterogeneities important for this kind of problems.

This thesis is devoted to geostatistical simulations of heterogeneities on unstructured grids using methods of non-linear geostatistics. The objective of this work is the development of algorithms for simulating heterogeneities directly on unstructured grids without using intermediate fine scale regular grids and upscaling. We present two theoretical models for geostatistical simulations of continuous parameters on unstructured grids which are different generalizations of the Discrete Gaussian model (DGM) – DGM 1 and DGM 2. The proposed theoretical models enable converting the problem of geostatistical simulation on an unstructured grid into the well-studied problem of simulating multivariate Gaussian random vectors followed by application of block-dependent transformation functions. The problem of simulating facies is also addressed in this work, for which generalizations of pluri-Gaussian and truncated Gaussian simulation models for unstructured grids are proposed.

An application of the proposed methods is demonstrated on a case study X, which is an offshore gas reservoir with a tartan-meshed grid.

Mots Clés

Maillage non-structuré, simulations des hétérogénéités, géostatistiques non-linaire, modèle Gaussien discret

Keywords

Unstructured grids, heterogeneities simulation, non-linear geostatistics, discrete Gaussian model