Application of Multiple-Point Simulation of Mineral Deposits Based on Discrete Wavelet Transform

Murilo Nogueira Teixeira

Degree of Master of Engineering

Department of Mining and Materials Engineering

McGill University

Montreal, Quebec, Canada

Dec. 15, 2014

A thesis submitted to McGill University as a partial fulfilment of the requirements of the degree of Master of Engineering

Dedication

This document is fully dedicated to our almighty God, my parents, my sister, my brother and Monessa. You are reason why I live. I love you!

Acknowledgements

I would like to thank my supervisor Dr. Roussos Dimitrakopoulos for his valuable guidance and support during all this time.

My greatest thanks go to my parents, my brother, my sister, my beloved Monessa and to my other brother Mario Silva. This achievement is as much yours as it is mine, I would not make it without you. I also appreciate all the support and time of my family in Canada, Vitor, Ana and Karina who, together with Mario, was always by me side to help me during difficult times and to celebrate the good ones.

Thanks also to my colleagues in Cosmo for all your help and patience during all this time. And I would also like to thank all people I met here in Canada and all my friends and family in Brazil who received me so well every time I went back there. And Prof. Claudio Lucio, I appreciate that you believed in me since the beginning. Thank you very much!

The work in this thesis was funded by the Canada Research Chair (Tier I) in "Sustainable Mineral Resource Development and Optimization under Uncertainty", NSERC Collaborative Research and Development Grant CRDPJ 411270 -10, "Developing new global stochastic optimization and high-order stochastic models for optimizing mining complexes with uncertainty" with industry collaborators: AngloGold Ashanti, Barrick Gold, BHP Billiton, De Beers, Newmont Mining and Vale, and NSERC Discovery Grant 239019, in particular to BHP Billiton and Brian Baird for providing the data of Olympic Dam deposit and conceding the permission to use it in this thesis.

Contribution of Authors

This section states the contribution of the co-author of the paper that comprises the present work. The author is the primary author of all the work presented herein, which was done with the supervision, advice and orientation of his advisor Prof. Roussos Dimitrakopoulos, who is also the co-author of all the papers to be published.

Abstract

Traditionally, geostatistical simulations of mineral deposits are done by using methods based on two-point spatial statistics, such as variograms. However, second-order spatial statistics are not sufficient to capture critical and common features from mineral deposits, such as connectivity of extreme values and curvilinear patterns, which in turn drive a mine production sequence. As a way to overcome these important limitations, multiple-point simulation (MPS) methods have been developed.

In this thesis, a MPS method based on wavelet analysis and on pattern recognition is described in detail. The idea in wavelet analysis applied to image compression is to decompose an image in two types of information: 1) average type information of the nearby pixels, called approximate sub-band of the image; and 2) how these pixels depart from local average. Usually, the sub-band image is a sufficient representation of the whole image, and can be used for several applications instead of the whole image. The simulation method used herein works as follows: first, it scans a training image with a template to generate a pattern database; then this database has its dimension reduced by applying discrete wavelet transform, so the approximate sub-band image of the patterns are obtained; after that, the patterns are divided into classes using k-means clustering algorithm, considering the approximate sub-band image; finally, the grid is simulated by comparing the conditioning data event in each node with the classes prototypes and choosing a pattern from that class. The practical intricacies and the contributions of this approach are tested and analyzed through an application at Olympic Dam copper deposit, which is located in South Australia and is the fourth largest producer of this commodity in the world. Both material types and copper grades were simulated in this case study. The categorical training image is generated through geological interpretation, and the continuous training image is generated by using low-rank tensor completion.

Olympic Dam's simulation results show that the method used herein can be applied successfully to relatively complex and large deposits. Additionally, the results suggest that care must be taken when generating the training image, since it plays a very important role in the simulation process. The resulting simulated realizations are analyzed and validated in terms of histograms,

variograms and high-order statistics, the latter being performed by using high-order spatial cumulants.

Résumé

Traditionnellement, les simulations géostatistiques des gisements minéraux sont effectuées en utilisant des méthodes fondées sur les statistiques spatiales utilisant deux points, comme les variogrammes par exemple. Toutefois, les statistiques spatiales du second ordre ne sont pas suffisantes pour capturer les caractéristiques essentielles et communes des gisements minéraux telles que la connectivité des valeurs extrêmes et les motifs curvilignes, caractéristiques qui ont un impact sur la planification des opérations et de la production de la mine. Pour pallier à ces importantes limitations, les méthodes de simulation à points multiples ont été développées.

Ce mémoire décrit d'une façon détaillée une méthode de simulation à points multiples basée sur l'analyse par ondelettes et la reconnaissance des formes. L'idée à la base de l'analyse par ondelettes lorsqu'appliquée à la compression de l'image est de décomposer l'image selon : 1) la quantité d'information moyenne des pixels voisins, dite sous-bande approximative de l'image (approximate sub-band of the image); et 2) l'écart entre les pixels et la moyenne locale. Souvent, dans plusieurs applications, la sous-bande de l'image constitue une représentation suffisante de l'image et peut substituer l'image entière. La méthode de simulation utilisée dans ce mémoire se résume comme suit: d'abord, l'image de formation (training image) est scannée pour générer la configuration de la base de données. Ensuite, la dimension de cette base de données est réduite en appliquant une transformée en ondelettes discrète ce qui permet d'obtenir la sous-bande approximative de l'image des modèles. Par la suite, les modèles sont partitionnés en classes en utilisant l'algorithme des k-moyennes tout en tenant compte de la sous-bande approximative de l'image. Finalement, la grille est simulée en comparant les données conditionnelles en chaque nœud aux prototypes de la classe et en choisissant un modèle de cette classe. Les contributions ainsi que les complexités pratiques inhérentes à cette approche ont été testées et analysées à travers une application à la mine de cuivre Olympic Dam, située au sud de l'Australie et considérée comme étant le quatrième plus grand producteur de cuivre dans le monde. Aussi bien les types de matériaux que les teneurs en cuivre ont été simulés dans cette étude de cas. L'image de formation catégorique (categorical training image) a été générée via l'interprétation géologique, alors que l'image de formation continue (continuous training image) a été générée à l'aide des techniques de « low-rank tensor completion ».

Les résultats de simulation de la mine Olympic Dam montrent que la méthode utilisée dans ce mémoire peut être appliquée avec succès à des gisements relativement complexes et de grande taille. De plus, les résultats suggèrent qu'il faut générer avec soin l'image de formation (training image) vu son importance dans le processus de simulation. Les réalisations simulées ont été analysées et validées en termes d'histogrammes, variogrammes, et statistiques d'ordre élevé, ces dernières étant effectuées en utilisant les cumulants spatiales d'ordre élevé (high-order spatial cumulants).

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Mining projects typically require intensive capital investments based on a series of decisions which have to be made without having full knowledge of the deposit being considered. In the past, this inherent uncertainty was ignored in the orebody modeling and mine planning stages, causing most of the forecasts to be unrealistic. Very often, the profit and production targets were not met, as the plan suggested. This happens because the planning and scheduling are performed based on one estimated model of the deposit, which is a smooth representation of the reallity, since it is constituted by the mean of all the possible grades in the different locations of the deposit taken separately.

These limitations suggested that only one estimated model of the deposit does not provide means for decisions to be made in a robust way. In order to overcome these problems, this uncertainty can now be modeled through stochastic geostatistical simulations. Rather than estimates of averages grades in the deposit, simulations are generated by directly drawing alternative realizations from the multivariate cumulative distribution function of the random field (RF) that characterizes the deposit, so constituting equally probable scenarios of this RF. Therefore, geostatistical simulation creates a basis for decision makers to take uncertainty into account. As consequence, the resulting plans are more robust, the risk in the project is better managed and plans are more realistic and more likely to be achieved. Stochastic simulations have been used since a long time now (Journel, 1974) and stochastic mine planning methods have been under development for the last decade (Godoy, 2003; Ramazan and Dimitrakopoulos, 2007; Dimitrakopoulos, 2011). Because orebody modeling is the basis for all the planning to be done, it is very important that they provide a reliable representation of the deposit being studied. Traditionally, geostatistical simulations are done through approaches based on two-point statistics only. However, these methods are not able to characterize important featured of mineral deposits, such as complex patterns and spatial connectivity of extreme values. In order to overcome these limitations, multiple-point

simulation (MPS) techniques were developed and an application of one MPS method, which is based on discrete wavelet transform, is shown in this thesis.

Despite being a great advance in the orebody modeling and mine planning fields, these methods are just starting to be used in mine projects. Although stochastic orebody models and mine planning brings a lot of advantages, they also add complexity to these stages. Because of that, several methods for both geostatistical simulations and mine planning have to be developed and efficiency is a crucial feature of these new implementations.

1.1 Goal and Objectives

The main goal in this thesis is to apply an efficient wavelet based multiple-point geostatistical spatial method in an actual mineral deposit. This way, it is possible to learn practical intricacies and contributions of such method for orebody modeling. More specifically:

- Review the literature on geostatiscal simulations, from earlier two-point based methodologies to newer developments, such as approaches based on multiple-point and high-order statistics.
- Describe the multiple-point simulation method based on wavelet analysis tested herein.
- Apply the method to Olympic Dam copper deposit in Southern Australia, showing all its intricacies, especially in the training image generation.
- State conclusions and suggest future work.

1.2 Thesis Outline

Chapter 1 provides an introduction to the thesis, together to its goals, objectives and outline.

Chapter 2 presents the literature review on geostatistical simulations and also contextualizes simulations in the mine project framework.

Chapter 3 describes the multiple-point simulation method based on wavelet decomposition used in this thesis. The methods to generate the training images are also described in the chapter.

Chapter 4 contains the application of methods presented in Chapter 3 to the Olympic Dam deposit in South Australia, along with some information about the deposit.

Chapter 5 contains the conclusions and related future work to be developed.

Chapter 2

Literature Review

The mine production chain is characterized by a sequence of operations, which comprises the extraction of material from a deposit and several processing streams, where the materials are sent to depending on their type and grade. Traditionally, the planning stage of mining and process is done based on single estimated, or average type, models of the deposit, therefore not taking uncertainty into account (Lerchs and Grossman, 1965; Johnson, 1969; Piccard, 1976; Dagdelen and Jonhson, 1986; Whittle, 1988; Hustrulid and Kuchta 1995; Tolwinski and Underwood, 1996; Cacceta and Hill, 2003; Boland et. al. 2009). In those approaches, the goal is to maximize the discounted cash flow of the project, but because only one model of the deposit is used as input, all attributes of the deposit are deemed to be known. As consequence, many operating mines do not meet the planned net present value (NPV) and production targets.

Although there are many sources of uncertainty when dealing with orebody modelling and mine planning, geological uncertainty is seen as the main reason why projects do not meet their expectations (eg. Vallee, 2000; Baker and Giacomo, 2001). The use of conditional simulation to generate orebody models, which can further be used to analyse risk in mining projects, is shown in David et al. (1974), Ravenscroft (1992), Dowd (1994, 1997), and Dimitrakopoulos, et al. (2002). Later, some methods to optimize open pit mine production and schedule stochastically were developed (Godoy, 2003; Ramazan and Dimitrakopoulos, 2007; Menabde et.al., 2007; Leite and Dimitrakopoulos; 2007 and 2009; Boland et al., 2008; and Lamghari and Dimitrakopoulos, 2012; Dimitrakopoulos and Ramazan, 2013). In these approaches several equi-probable scenarios of the orebody are used as input to the optimization, and the goal is not only maximizing project's NPV, but also minimizing deviations from production targets. More recently, there are works on global optimization of mining complex (Whittle, 2010), in which mine, processing and transportation are considered simultaneously in the model (Montiel, 2014; Goodfellow, 2014).

Since geological uncertainty has a major impact on open pit mine production and scheduling, it is important generate a representative model of the orebody. Traditionally, this is done through simulation methods based on second-order statistics. However, these methods are not able neither to capture complex pattern in geological environment nor to reproduce spatial connectivity of extreme values. In order to overcome these limitations, multiple point simulation (MPS) methods are developed. This thesis is devoted to show an application of a MPS method based on discrete wavelet transform, termed *wavesim* . But first, some other relevant simulation techniques are reviewed in this chapter, such as second-order and then MPS methods.

2.1 Variogram-based Simulation Methods

Spatial uncertainty is typically modeled by generating multiple realizations of the joint distribution (random field) of a given attribute, a process known as geostatistical simulation (Matheron, 1973; Journel, 1974; David, 1977, 1988; Goovaerts, 1997). Differently to modeling in petroleum reservoir, where the flow is what needs to be modeled, in orebody modeling the location and the connectivity of the grades, especially the high ones, have to be well understood, so that the planning and scheduling stages can be done successfully. This is possible through geostatistical simulation.

By using stochastic geostatistical simulation, severe limitations related to working with estimated models are overcome. The main ones are: having only one representation of the deposit; the conditional bias, which is the underestimation of high values and overestimation of low values; and treating each block separately, even if they should be treated jointly. As many realizations of the deposit are made available through simulation techniques, a quantitative measure of uncertainty is provided. Very important characteristics of stochastic simulations are: honoring data values in their locations; reproducing declustered data histogram; and reproducing covariance model. As it will be discussed latter in this chapter, multiple-point simulation methods reproduce also high-order statistics which the deposit is believed to have. It is important to state that the random field used to model the deposit must be ergodic and second-order stationary so that two-point simulation methods can be applied.

Two-point geostatistical simulation relies on the sequential simulation framework (Rosenblatt, 1952; Kolmogorov, 1956). In this approach, in each node to be simulated, a conditional cumulative probability function (ccdf) has to be generated considering both initial hard data set and previously simulated nodes. For doing that, first consider the distribution of the random field (RF) $\{Z(\boldsymbol{u}_i), i = 1, \dots, N\}$, where $\mathbf{u}_i$ define different locations in the study area, conditioned to the data set $\{z(\boldsymbol{u}_\alpha), \alpha = 1, \dots, n\}$, as follows (Rosenblatt, 1952):

$$F\big(\boldsymbol{u}_1, \boldsymbol{u}_2, \dots, \boldsymbol{u}_N; z_1, z_2, \dots, z_N \big| (n)\big) = Prob\{Z(\boldsymbol{u}_i) < z_i, i = 1, \dots, N | (n)\} \tag{2.1}$$

Equation (2.1) can be decomposed using equation (2.2) so that the spatial law of the RF can be written as a product of univariate conditional distributions.

$$\begin{aligned} F\big(\boldsymbol{u}_1, \boldsymbol{u}_2, \dots, \boldsymbol{u}_N; z_1, z_2, \dots, z_N \big| (n)\big) = F\{Z(\boldsymbol{u}_i; z_i) | (n)\} \\ \cdot F\{Z(\boldsymbol{u}_2; z_2) | (n+1)\} \cdots \\ \cdot F\{Z(\boldsymbol{u}_N; z_N) | (n+N-1)\} \end{aligned} \tag{2.2}$$

Since a conditional cumulative distribution function (ccdf) can be generated for all of the nodes in the grid, one at a time, it is possible to randomly draw a value for each of them.

In summary, sequential simulation is implemented as follows:

1. Define random path.
2. Generate a ccdf for a given node.
3. Draw a value from ccdf in Step 2, which become conditioning datum for further drawings.
4. Repeat Steps 2 and 3 until all nodes are simulated.
5. Repeat Steps 1 to 4 to generate more realizations.

2.1.1 Sequential Gaussian Simulation

Sequential Gaussian Simulation (SGS) (Goovaerts, 1997; Isaaks, 1990) is a commonly used method which requires that the ccdf's follow the Gaussian distribution. Two additional steps are needed in addition of ones shown above: in the start, the data set is transformed into standard normal scores; and after the simulation process is complete, the

simulated values are back-transformed from Gaussian to data space. In this approach, mean and variance are estimated through kriging, for each node. Then, these two parameters are used to build a normal distribution which defines the ccdf of any given node.

Equivalent method to SGS is the simulation through LU decomposition (Davis, 1987). In this method, the covariance matrix is decomposed in lower and upper triangular matrices using the so called Cholesky decomposition and the realizations are generated by multiplying the lower one with a vector of independent and normally distributed random numbers. Each row of the resulting vector corresponds to one node in the simulation grid; so, if implemented row by row it is equivalent to the sequential Gaussian simulation. This method is extremely efficient, but it is only able to handle very small grids, and also assume a Gaussian RF.

Dimitrakopoulos and Luo (2004) proposed the generalized sequential Gaussian simulation (GSGS), which is a hybrid between the two previously mentioned Gaussian-based methods: a random path is defined to visit a group of nodes at a time, which is simulated at once using LU. Therefore, it is able to deal with large grids in a more efficient way than SGS. In order to perform the simulation in groups, in GSGS the random field is decomposed into groups of nodes, rather than in single nodes, as in SGS. Despite of these advantages, GSGS still generates realizations on point support. Thus, in order obtain realizations in block support, the nodes discretizing each block have to be averaged out in a post-processing step.

A simulation method which generates realization directly in block support is shown in Godoy (2003), termed direct block simulation (DBSIM). The idea is similar to GSGS: first, a group of nodes are simulated using LU; then these nodes are averaged to obtain the value for the block; and finally the values of the nodes are discarded, only the block value is kept for further conditioning. In order to do this, the assumption made is that the random fields related to point and block supports are jointly Gaussian. This framework was extended to simulate multiple correlated variables (Boucher and Dimitrakopoulos, 2009a). In this method, the variables are first orthogonalized through

minimum/maximum autocorrelation factor (MAF) and after the simulation is done, the independent factors are back-transformed to the data space.

2.1.2 Sequential Indicator Simulation

Sequential indicator simulation (SIS) differs to SGS only on the generation of the ccdf in each node. In this approach, the ccdf is generated as follows (Journel, 1989a): first, K threshold values are defined; then indicator transformation is performed for conditioning information regarding each of the K threshold; and finally, the ccdf can be generated by using the transformed values and indicator kriging. The advantages of SIS over SGS are: data does not need to follow Gaussian distribution and it reproduces better the continuity of extreme values. However, it is computationally more expensive and still does not take high-order statistics into account.

2.1.3 Limitations

Two-point statistics is not enough to model some complex cases shown by mineral deposits, such as non-linear features and spatial connectivity of extreme values. The latter is a severe limitation, since high grades drive the profitability of mining projects and high permeability values indicate the direction of flow in petroleum reservoirs. Limitations of variograms-based simulation methods are discussed in Journel (2005) and Journel (2007). An example of this is presented in Figure 1, which was taken from Journel (2007). It shows that structures with very different spatial patterns can share very similar histograms and variograms. Because of that inability of simulation methods based on two-point statistics to model common features in mineral deposits, development of techniques which make use of high-order statistics was necessary.

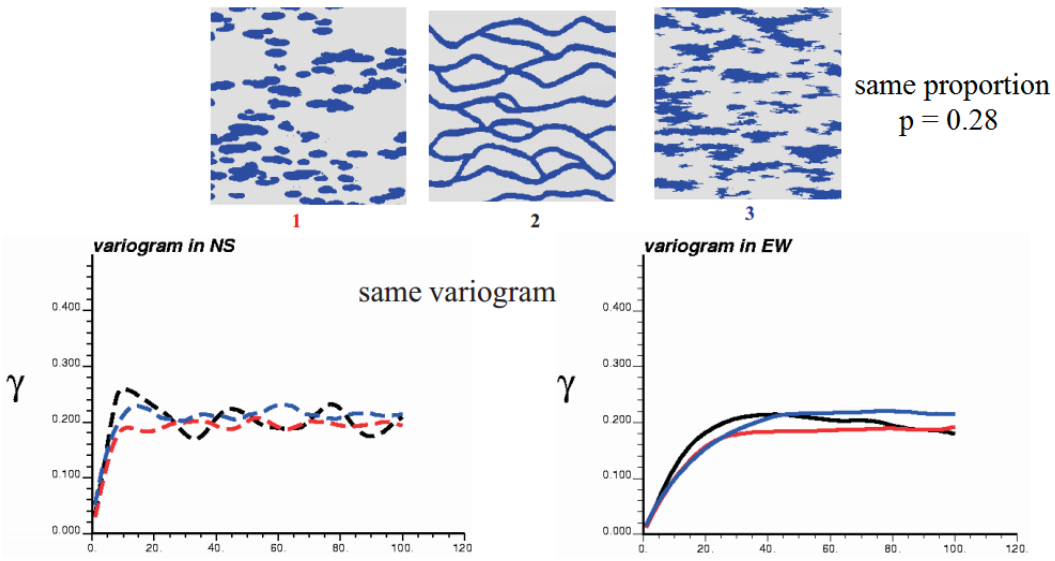


Figure 1- Different patterns, same statistics up to order two. Figure taken from Journel (2007).

2.2 Multiple-point Simulation Methods

2.2.1 Probabilistic Framework

Since all multiple point simulation (MPS) (Journel and Alabert, 1989) methods make use of training image (TI), it is important to define it. Training image is a conceptual rendering of the major variations that may exist in the area being studied. It may be based on actual data, or on other exhaustive data set considered to be representative of the area being modeled. In any of the two situations, it does not need to have the same dimensions of the area to be represented or to be constrained to available data; however it should present similar pattern of spatial continuity to the actual phenomenon (Strebelle, 2000; Zhang, 2006).

Multiple point statistics (MPS) was first incorporated in a geostatistical simulation framework by Guardiano and Srivastava (1993) through an algorithm called extended normal equation simulation (*enesim*). Also based on the sequential simulation paradigm, at each node of the realization grid, *enesim* retrieves a conditioning neighbourhood comprising a multiple-point *data event* (i.e., samples and previously simulated nodes) defined over an *n*-configuration template of (0, h_1…, h_n). So, let A_k defines the

occurrence of category s_k in u and D defines the occurrence of data event d_n constituted by n conditioning data, as follows:

$$A_k = \begin{cases} 1 & if\ S(u) = s_k \\ 0 & otherwise \end{cases} \quad \text{and} \quad D = \begin{cases} 1 & if\ S(u_\alpha) = s_{k_\alpha}, \forall\ \alpha = 1, \dots, n \\ 0 & otherwise \end{cases} \tag{2.3}$$

the exact probability of A_k conditioned to D is given by the simple kriging expression:

$$Prob\{A_k = 1 | D = 1\} = E\{A_k\} + \lambda[1 - E\{D\}] \tag{2.4}$$

where $D = 1$ is the observed data event, $E\{D\} = Prob\{D = 1\}$ is the probability of the data event to occur, and $E\{A_k\} = Prob\{S(u) = s_k\}$ is the prior probability of category k occurs in location u. Then, the single normal equation providing the single kriging weight λ is:

$$\lambda Var\{D\} = Cov\{A_k, D\} \tag{2.5}$$

from where the weight λ can be calculated:

$$\lambda = \frac{E\{A_k D\} - E\{A_k\}E\{D\}}{E\{D\}(1 - E\{D\})} \tag{2.6}$$

and replacing equation 2.6 in equation 2.4:

$$Prob\{A_k = 1 | D = 1\} = E\{A_k\} + \frac{E\{A_k D\} - E\{A_k\}E\{D\}}{E\{D\}} \tag{2.7}$$

$$= \frac{E\{A_k\}}{E\{D\}} = \frac{Prob\{A_k = 1, D = 1\}}{Prob\{D = 1\}}$$

The denominator can be determined by counting the amount of replicates of conditioning data event D in the training image and the numerator is the amount of replicates, among the previous ones, associated with the category s_k. The equations from 2.3 to 2.7 are taken from Strebelle (2002). The central node is, then, simulated by drawing a value from this conditional distribution. As one may note, in *enesim* the whole TI needs to be scanned again every time a grid node has to be simulated. Therefore, due to its high computational complexity and CPU cost, the proposed approach of Guardiano and Srivastava (1993) remained impractical for several years. A significant improvement on this original

algorithm, which allowed the practical simulation of large models, was accomplished through the development of *snesim* in Strebelle (2002), which is the first structured multiple-point statistics algorithm for the simulation of categorical variables. The same basic ideas of *enesim* are kept in *snesim*, however, the computational complexity is drastically reduced by computing the conditional probabilities prior to the simulation. Instead of repeatedly scanning the TI for each conditioning data-event, it stores all probabilities in a search-tree structure by a single-time scanning of the TI with a global template. The introduced data structure allows fast retrieval of all required conditional probabilities by the time of the simulation. Such drastic improvement in running time came along with a high RAM memory requirement to build the search trees. In an extreme case, where all possible data events defined by the global template are present in the TI, the number of patterns to be stored is M^N, where M is the number of categories and N the size of the global template. In practice, however, the TIs do not have all that amount of patterns, which allows the practical implementation of such data structure. In addition, *snesim* uses a pixel-based sequential simulation approach, which makes the conditioning to hard data easy. Besides the potential prohibitive memory costs for the applications in large mineral deposits, *snesim* presents the following shortcomings: (a) it cannot handle simulation of continuous variables; (b) *snesim* typically captures stationary features of the training image; (c) when the probability of a data event is not found in the search-tree, the furthest node is dropped, which incurs in some loss of conditioning information; (d) which is the most important one: it is training image driven, as all MPS methods.

In order to reduce memory requirements of *snesim*, instead of using a search-tree, Straubhaar et al. (2011) proposed a list structure for storing multiple-point statistics inferred from the TI. Each element of the list is designed to store a pair of vectors (d, c), where *d* stores a spatial pattern from the TI and *c* stores the associated counters for the different categories at the central node. After being built, the list is sorted based on a reference category, in order to allow a fast search for desired data events during the simulation. This new list structure requires much less memory than the original search-tree. In addition, it allows parallelizing the retrieval of conditional probabilities. Despite these improvements, the algorithm behaves similar to the previously introduced *snesim*.

Boucher et. al., (2014) applied *snesim* for the simulation of contacts only in geological environments.

2.2.2 Pattern-based Approaches

All the pattern based approaches present one common feature: unlike the previous probabilistic methods, not only a central node value is drawn from a local conditional distribution, but a local pattern is pasted accordingly to its 'similarity' to the conditioning information. Because of that, the measure of similarity, which is done through distance measure function, plays a very important role in pattern based approaches. An example of such distance measure function is the L_2-norm, defined as:

$$d(x,y) = \sum_{i=1}^{3} \left\{ \omega_i * \left[\frac{1}{n_{type}} \sum_{j=1}^{n_{type}} \big(x(j) - y(j)\big)^2 \right] \right\} \qquad (2.8)$$

Where *x* and *y* represent the conditioning data event and pattern respectively, ω is the weight for different conditioning data types (hard data and previously simulated nodes, for example) and n_{type} is the amount of data of each type.

Zhang et. al. (2006) and Zhang (2006) proposed a new algorithm, called *filtersim*. In this algorithm, the training image (TI) is treated as a collection of *patterns*, which are meaningful geological entities defined over a multiple point configuration (a *template*). Scanning the TI with a given template results in a *pattern database*, which can be seen as multiple pieces of a jigsaw puzzle that needs to be put together in a logical fashion during the stochastic simulation. Each of the patterns of the database is characterized by a series of *filter scores*, which are real values resulting from the application of some positional linear filters. Then, the patterns are grouped into classes according to some similarity criterion regarding their filter scores (e.g., by using *k-means* clustering) and each class is labeled by a prototype, which is simply the pixel-by-pixel average of all patterns in the given class. After these pre-processing steps, a sequential simulation takes place. At each node location, the multiple-point conditioning *data event* is retrieved, but it is only compared to each of the classes' *prototypes*. Then, a random pattern is drawn from the class with the closest *prototype* found and it is patched on the realization grid. An inner

part of the pattern is frozen for the next sequence of simulation (i.e., besides the central node, a surrounding is simultaneously simulated). The outer part is pasted on the grid as 'soft data', assisting on similarity calculations, but the corresponding nodes are re-visited along the random path, hence re-simulated. In *Filtersim*, because the *data event* may contain information with difference importance (i.e., samples, inner and outer patches), weights are given for the similarity calculation between the *data event* and the *prototypes*. An interesting point in *filtersim* is the feature of randomly drawing patterns from a class, which introduces a major stochasticity to the process. Besides, this method solves the great burden associated to RAM usage from *snesim* proposed by Strebelle (2002), because in fact, the *pattern database* extracted from the TI is never explicitly built, it remains in the TI and only the locations of each pattern are indexed. Despite the numerous improvements, the dependence on linear filters for the *patterns classification* is one of the main pitfalls of Filtersim. Different geological domains may require the usage of different filters, since their capability to correctly exploit the differences between the patterns is intrinsically related to the structures and geometries of the patterns themselves. So, it becomes cumbersome in practice to define the appropriate filters to use.

Arpat (2005) and Arpat and Caers (2007) proposed a simulation algorithm called simulation with patterns (*simpat*). In *simpat*, the pattern database is generated as in *fultersim*; however, the simulation process is different. The sequential simulation paradigm is also used to generate a realization but, at each node location, a *data event* is compared to all patterns from the *pattern database*. Then, the most similar *pattern* is entirely pasted on the realization grid. The simulation proceeds until all nodes of the grid have been visited. In this method, the function used to measure similarity between data event and patterns have a great impact on the quality of the simulation. Arpat (2005) suggests the use of Minkowski metrics, such as Euclidean and Manhattan distances, since they are commonly used in Earth's sciences applications. For categorical simulations, the author also suggests the use of these metrics coupled with a "proximity transform", which transform binary images to continuous images, by mapping the proximity of each node of the grid to the target object. The main issues concerning such approach are: (a) the large computational complexity associated with the similarity searches, because it actually compare the retrieved *data event* to all patterns in a TI, leading to a poor CPU

performance; and (b) the fact that the randomization of the process is mostly linked on the definition of the random path, hence the realizations are likely to be merely shuffles of the TI.

Honarkhah (2011) and Honarkhah and Caers (2010) propose a framework where a distance-based method is used to represent patterns of the TI as points in an arbitrary space, such that the dissimilarities between the patterns is related to the distances between the points in that space. First, a dissimilarity matrix between the several patterns is calculated using a distance function such as the Euclidean distance. Then, multidimensional scaling (Cox and Cox, 2010) is used to transform the dissimilarity matrix into a set of coordinates such that the Euclidean distances obtained from these coordinates approximate as well as possible the original distances. Once the high-dimensional data (patterns) are represented in a lower-dimensional space (preserving the intrinsic dissimilarities), a clustering algorithm is used to group the patterns into classes. For this purpose, Honarkhah (2011) and Honarkhah and Caers (2012) suggest the application of kernel k-means which works on a high-dimensional feature space. Applying k-means in the feature space increases the capability of the algorithm for dealing with non-linear and complex structures. The main drawback concerning such distance-based framework is its computational complexity. Both the application of multidimensional scaling and kernel k-means becomes very computationally costly for large datasets. Honarkhah and Caers (2010) give several solutions to tackle such issues by either adapting these algorithms or by using some pre-processing step to reduce the dimensionality of the problem.

Mustapha et. al. (2014) introduced an algorithm in which the spatial patterns are first mapped from their high-dimensional space to a set of real values that are further classified in a partitioning step (*CDFSim*). In this later process, the real values are plotted in a two dimension Cartesian coordinate system with their respective values in the y-axis and their locations in the x-axis. Then, each value is scaled by the total sum and the obtained values are reordered in increasing order. From these new values, a cumulative distribution is built, with the ordinate-values bounded between zero and one. Afterwards, ‘quantiles’ can be used to split the values into different classes (e.g., if two classes are

desired, patterns mapped with values below 0.5 threshold belong to a given class and the values above to another). Another option is to find the center of each of those intervals and retrieve the closest patterns to them. Then, these patterns are used as classes' centroids and the other patterns are grouped in the class of the closest centroid, measured using Euclidean distances between the patterns in their original space. Such method is potentially faster than k-means based algorithms because the centroids of classes are only defined a single time rather than iteratively until convergence is reached. The downside of such approach is that summarizing patterns in one dimensional space may hide important dissimilarities arising from complex structures in the spatial patterns.

Mariethoz et. al. (2010) proposed a direct sampling (DS) method which, as opposed to storing and counting the configurations found in the TI (*snesim*), directly samples the TI in a random manner, conditioned to the data event. Contrary to the other methods inspired on *simpat*, in *DS* the shape and size of the templates are not predefined. DS proceeds as following: (i) a random path is defined to sequentially visit each of the nodes; (ii) at a given node location, *n* neighbouring information within a search radius is retrieved; (iii) this data event (neighbourhood) defines a set of lag vectors (multi-pixel configuration) that is used to randomly scan the TI, seeking for similar patterns. Whenever a pattern whose dissimilarity falls below a given threshold is found, the algorithm stops the search and the central node is simulated. If such a pattern is not found after a given number of iterations, the closest pattern found so far is retrieved for the simulation of the central node.

An important accomplishment of *DS* is that, reproduction of large scale structures is inherent of its process, without the need of a multiple grid implementation, since in the beginning of the simulation the search radius tend to be much larger than in the end, in order to retrieve the same amount of *n* neighbouring information. A faster version of *DS* is accomplished in Rezaee et. al., (2013) where a bunch of nodes are simultaneously patched with the central node being simulated. The Direct Sampling algorithm is similar to *simpat* but potentially increases it CPU performance. However, it accounts with a large number of parameters that need to be set and tuned for providing reasonable realizations.

Aiming to provide realizations with a better continuity, a patchwork simulation method (PSM) is proposed by Faucher et. al. (2013) where simulation is carried out on a moving squared template that, because of the unilateral path, overlaps an L-shape area of previously simulated sites, which are the data events to be compared with the patterns of the TI database. For the candidate patterns which are most similar to the data event, one is randomly drawn accordingly to a *transition probability*, and only its bottom-right region is patched on the realization grid. An important achievement of this approach is the adjustment of the *transition probabilities* in order to enforce the reproduction of histograms during the simulation process, instead of assigning equal probabilities for each of the closest patterns to be randomly drawn.

There are also methods based on other types of frameworks, which are not described here, such as object-based simulation (Haldorsen and Lake, 1984) and iterative approaches (Bratvold et. al., 1994; Tyler et. al., 1992).

2.2.3 Dealing with Stationarity and Long-range Structures in MPS Methods

Over the years, a series of general improvements were proposed and they are extendable to multiple of the MPS algorithms previously shown. For example, when *snesim* was developed by Strebelle (2000), the author proposed an extension of the multiple-grid simulation, first introduced by Tran (1994) for variograms-based approaches. A number G of nested and increasingly finer grids are sequentially simulated, with templates proportionally re-scaled to the spacing of the nodes. Such strategy allows capturing for large-scale training structures at coarse grid levels, which improves the quality of the realizations. Strebelle and Cavelius (2014) add the idea of intermediate subgrids to allow the simulation of a higher number of nodes during coarse levels and using data templates that preferentially include previously simulated nodes. Arpat (2005) coupled the multiple-grids with the idea of dual templates, for populating the finer grids during the simulation of coarse grids. Such idea is interesting because, in pattern based MPS methods, it is often desired to have a fully known data event, which allows similarity calculations in a low dimensional space (e.g., Filtersim (Zhang et. al., 2006); and Wavesim (Chaterjee et. al., 2011, 2012). Strebelle and Zhang (2005) brought a series of ways to deal with non-stationary features (e.g., rotation and affinity), especially for the

context of *snesim*. For doing so, multiple search-trees must be built in order to allow the characterization of multiple domains of stationary. Similarly for pattern-based MPS, Honarkhah and Caers (2012) present two different ways of considering the spatial coordinate of the patterns during the dissimilarity calculation, thereby only drawing patterns locally from the training image rather than globally (i.e., hypothesis of local stationarity). Therefore, in such extension, the TI and the realization grid must be of the same sizes. A location dependent strategy to deal with non-stationarity is also implemented in Wavesim for the case study detailed in Chapter 3.

2.2.4 High-order Simulation Based on Spatial Cumulants

Also to reduce the high dependency on the TI, seen in most of the MPS methods, Mustapha and Dimitrakopoulos (2010) introduced a MPS framework based on spatial cumulants. Cumulants, which can be written as a function of moments, bring an attractive way for characterizing non-Gaussian random process (Rosenblatt, 1985). Dimitrakopoulos et al. (2010) have shown that high-order spatial cumulants, calculated through the definition of a spatial template, can be used to capture complex geological features and geometrical shapes of the natural phenomena. Through the visualisation of cumulant maps it is possible to identify and characterize redundancy, orientation and a series of other features of the spatial process. The deeper understanding about the relationship between cumulants as mathematical entities and their ability to characterize geological patterns led the same authors to develop a high-order simulation method based on spatial cumulants, termed *hosim* (Mustapha and Dimitrakopoulos, 2010), which is also based on a sequential simulation paradigm. The core modification is during the estimation of the local conditional probabilities. Consider the following relation:

$$f\left(z_0 \middle| (\Delta_0)\right) = \frac{f_{z_0, z_1, \ldots, z_n}\,(z_0, z_1, \ \ldots, \ z_n)}{f_{\Delta_0}\,(\Delta_0)} \tag{2.9}$$

where $\Delta_0 = \{z_1, \ldots, z_n\}$ are the conditioning data, z_0 is the node to be simulated and:

$$f_{\Delta_0}(\Delta_0) = \int_D f_{z_0, z_1, \ldots, z_n}(z_0, z_1, \ldots, z_n)\, dz_0 \tag{2.10}$$

The numerator in equation 2.9 can be obtained through a series of orthonormal polynomials weighted by Legendre cumulants:

$$f_Z(z_0, z_1, \ldots, z_n) = \sum_{i_0=0}^{\infty} \ldots \sum_{i_{n-1}=0}^{\infty} \sum_{n=0}^{\infty} L_{\bar{\imath}_0, \ldots, \bar{\imath}_{n-1}, \bar{\imath}_n} \bar{P}_{\bar{\imath}_0}(z_0) \ldots \bar{P}_{\bar{\imath}_{n-1}}(z_{n-1}) \bar{P}_{\bar{\imath}_n}(z_n) \tag{2.11}$$

$$\bar{\imath}_k = i_k - i_{k+1}\,, k < n$$

where $\bar{P}_m(z)$ is the normalized m[th]-order Legendre polynomial and L denotes the Legendre cumulants. If a maximum order of approximation ω is defined, $f_Z(z_0, z_1, \ldots, z_n)$ can be approximated through the following relation:

$$f_Z(z_0, z_1, \ldots, z_n) = \sum_{i_0=0}^{\omega} \ldots \sum_{i_{n-1}=0}^{i_{n-2}} \sum_{n=0}^{i_{n-1}} L_{\bar{\imath}_0, \ldots, \bar{\imath}_{n-1}, \bar{\imath}_n} \bar{P}_{\bar{\imath}_0}(z_0) \ldots \bar{P}_{\bar{\imath}_{n-1}}(z_{n-1}) \bar{P}_{\bar{\imath}_n}(z_n) \tag{2.12}$$

The Legendre cumulants are calculated by using the following equation:

$$L_{i_0 \ldots i_N} = \int_{D^N} \overline{P_{\imath_0}}(z_0) \ldots \overline{P_{\imath_N}}(z_N) \ldots f(z_0 \ldots z_N) dz_0 \ldots dz_N \tag{2.13}$$

All the equations from 2.9 to 2.13 were taken from (Mustapha and Dimitrakopoulos, 2010). Hosim is data driven, as opposed to the other high-order simulation methods, which are training image driven. This means that *hosim* tries to first infer the multiple-point statistics from the data, only relying on the TI when a small amount of replicates are found.

2.2.5 Validating Simulated Realizations

The simulated realizations of Olympic Dam copper deposit, presented in Chapter 4, are validated in terms of histograms, variograms and high-order statistics. The latter is analyzed through high-order spatial cumulants (*hosc*) (Dimitrakopoulos et. al., 2010), whose use for validating simulation results is shown in De Iaco and Maggio (2011).

Cumulants are an extension of the covariance function, and are able to describe non-Gaussian stationary and ergodic random fields, since they can capture complex spatial patterns and their connectivity in geological environments (De Iaco and Maggio, 2011). All the statistics from the simulated realizations are compared to both the training image and data set.

Chapter 3

Multiple-Point Simulation Based on Wavelet Analysis

The wavelet transformation based simulation method of Chatterjee et. al. (2011, 2012) is outlined in this section, pointing at the differences between categorical and continuous simulations, where appropriate. It consists in four main steps: 1) generation of the pattern database, 2) Dimensional reduction of the pattern database by using discrete wavelet transform, 3) Classification of the pattern database, and 4) Simulation based on the patterns. Each of these steps are described in more detail in this chapter.

3.1 Pattern Database Generation

The training image is usually discretized in a regular grid G_{ti} and, for each location $u \in G_{ti}$ in this grid, $ti(u)$ represents the value of this node in the training image. Besides, $ti_T(u)$ indicates a multiple-point vector, which is defined over a spatial template T centered in node u, i.e:

$$\mathrm{ti_T(u)} = \{\mathrm{ti(u + h_1)}, \mathrm{ti(u + h_2)}, \dots, \mathrm{ti(u + h_\alpha)}, \dots, \mathrm{ti(u + h_{nT})}\} \quad (3.1)$$

where h_α are the vectors which define the geometry of the template containing nT nodes, and $\alpha = \{1, 2, \dots, \mathrm{nT}\}$. The vector $\mathrm{h_1} = 0$ represents the central node of the template. Figure 2 shows an example of a template. In order to generate the pattern database, a template size and geometry has to be first determined, according to the definition presented above. Then, the training image is scanned using this template, and each pattern is generated by centering it in each of the nodes in the training image. It is important to observe that, after a pattern is obtained, its location does not matter anymore.

For categorical simulation, in some cases, the process has to be performed for many categories. The way to deal with this is to generate one training image for each category, using indicator variables transformation. For example, say there are M categories to be simulated. The indicator variable $I_m(u)$, $m = 1, \dots, \mathrm{M}$, $u \in G_{ti}$ is defined as:

$$I_m(u) = \begin{cases} 1, & \text{if } u \text{ belongs to category } m \\ 0, & \text{otherwise} \end{cases} \quad (3.2)$$

Therefore, each location is represented by a vector of binary values, where the m^{th} element is 1 if that node belongs to category *m*, and 0 otherwise. Thus, for each node, there will be exactly one element equal to 1. In the other hand, for the continuous case, there is no need to transform the values. The patterns are stored exactly as they appear in the training image. Figure 3 shows the pattern database related to the template defined in Figure 2, for a two-category training image.

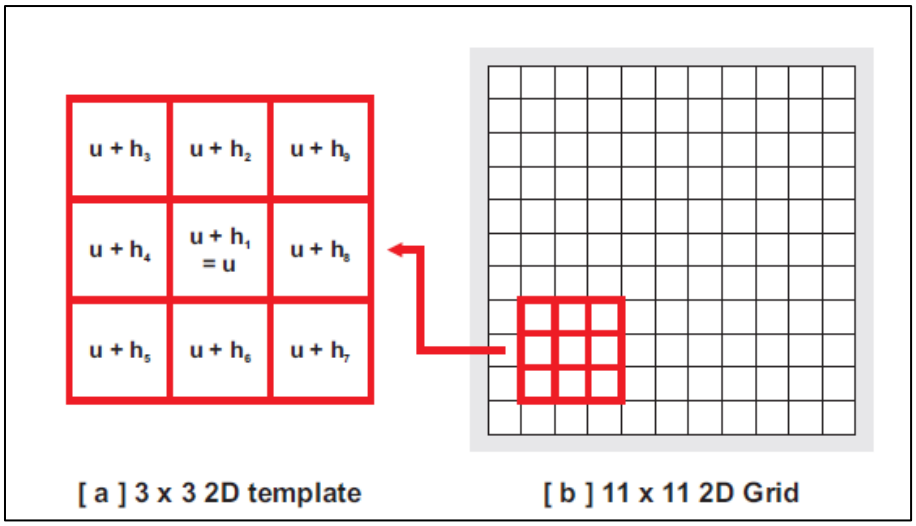


Figure 2 - Example of a spatial template. Figure taken from Honarkhah and Caers (2010).

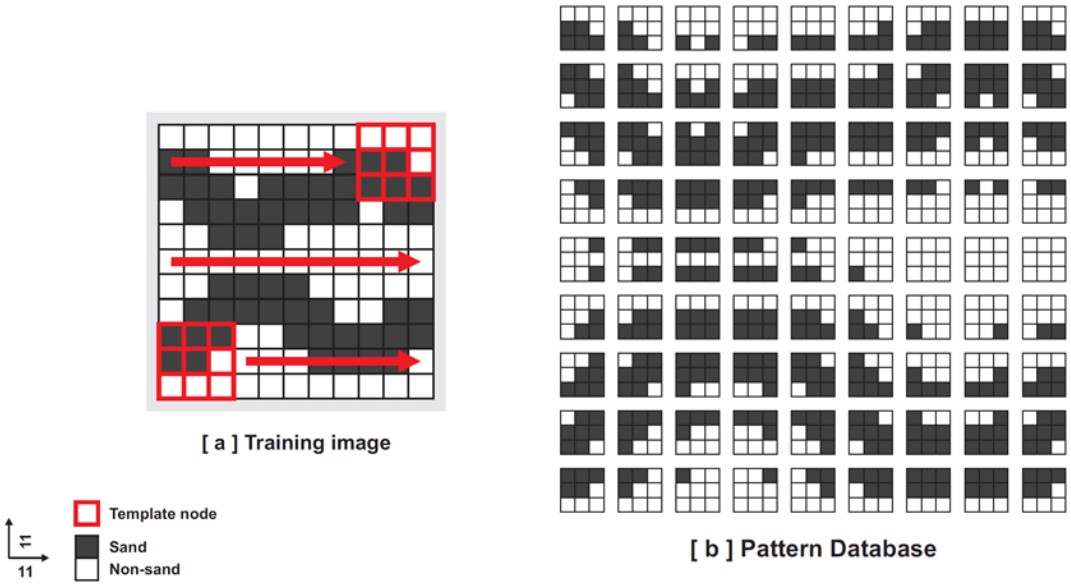


Figure 3 - Generation of the pattern database. Figure taken from Honarkhah and Caers (2010).

3.2 Pattern Database Decomposition

After the pattern database is generated, it has to be divided into classes, so that the simulation can be performed. However, if it is done to the original patterns, this classification can be extremely computational expensive, and the time it would take for large cases could be prohibitive. Hence, before being divided into classes, the patterns have their dimension reduced, so that the classification step can be done efficiently. This same idea is used in *filtersim* (Zhang, 2006; Zhang, Switzer and Journel 2006; Wu, Zhang and Journel, 2008), where the dimension reduction was done by applying some filters to each of the patterns, and the classification was carried out based on the filter scores. In the two-dimensional case, 6 x *M* filters are used and in three-dimensional case, 9 x *M* filters are used to classify the patterns, where *M* is the number of categories in the training image. In the case of *wavesim*, the dimension reduction is performed through wavelet analysis.

Wavelet transformation is widely used, and its main applications are in signal processing and in data compression. Bénéteau and Fleet (2011) present an intuitive introduction to this subject, and Mallat (1998) shows it in more detail. The idea in wavelet analysis applied to image compression is to decompose an image in two kinds of information: 1)

average type information of the nearby pixels, called approximate sub-band of the image; and 2) how these pixels depart from local average. Both of them together keep all the information about the image, but the first one keeps most of its variability. Hence, the approximate sub-band is usually an enough representation of the complete image for several applications and can be used instead of the original image. The advantage of using the sub-band image instead of the original one is related to less memory requirements and efficiency, in computational terms. Figure 4 shows an image and its first and second scale wavelet decomposition. In 2D cases, the approximate sub-band images are calculated by averaging each 2x2 squares, and the second kind of information mentioned above is how the values of each of the directions (horizontal, vertical and diagonal) depart from that average. That is why there are 4 resulting images after each scale of decomposition. Besides, as one can see Figure 4 the second scale decomposition is obtained by applying wavelet analysis to the approximate sub-band after first scale decomposition.

In a more formal way, define $\widetilde{W}_N$ as being the following matrix:

$$\widetilde{\mathrm{W}}_{\mathrm{M}} = \left[\begin{array}{cccccccc} \frac{1}{2} & \frac{1}{2} & 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} & & 0 & 0 \\ & & \vdots & & \ddots & \vdots & \\ 0 & 0 & 0 & 0 & \cdots & \frac{1}{2} & \frac{1}{2} \\ \hline -\frac{1}{2} & \frac{1}{2} & 0 & 0 & & 0 & 0 \\ 0 & 0 & -\frac{1}{2} & \frac{1}{2} & \cdots & 0 & 0 \\ & & \vdots & & \ddots & \vdots & \\ 0 & 0 & 0 & 0 & \cdots & -\frac{1}{2} & \frac{1}{2} \end{array}\right] \tag{3.3}$$

$$\widetilde{\mathrm{W}}_{\mathrm{M}} = \left[\frac{\widetilde{\mathrm{H}}_{\mathrm{M}/2}}{\widetilde{\mathrm{G}}_{\mathrm{M}/2}}\right] \tag{3.4}$$

The wavelet decomposition of a 2D image $\mathrm{A}_{\mathrm{MxN}}$ is given by:

$$\widetilde{\mathrm{W}}_{\mathrm{M}} * \mathrm{A} * \widetilde{\mathrm{W}}_{\mathrm{N}}^{\mathrm{T}} = \left[\frac{\widetilde{\mathrm{H}}_{\mathrm{M}/2}}{\widetilde{\mathrm{G}}_{\mathrm{M}/2}}\right] * \mathrm{A} * \left[\widetilde{\mathrm{H}}_{\mathrm{N}/2} \middle| \widetilde{\mathrm{G}}_{\mathrm{N}/2}\right] \tag{3.5}$$

$$\widetilde{W}_M * A * \widetilde{W}_N^T = \left[\frac{\widetilde{H}_{M/2}}{\widetilde{G}_{M/2}}\right] * A * \left[\widetilde{H}_{N/2} \middle| \widetilde{G}_{N/2}\right] \tag{3.6}$$

$$\widetilde{W}_M * A * \widetilde{W}_N^T = \left[\frac{\widetilde{H}_{M/2} * A * \widetilde{H}_{N/2}}{\widetilde{G}_{M/2} * A * \widetilde{H}_{N/2}} \middle| \frac{\widetilde{H}_{M/2} * A * \widetilde{G}_{N/2}}{\widetilde{G}_{M/2} * A * \widetilde{G}_{N/2}}\right] \tag{3.7}$$

$$\widetilde{W}_M * A * \widetilde{W}_N^T = \left[\frac{B}{H} \middle| \frac{V}{D}\right] \tag{3.8}$$

where *B*, *V*, *H* and *D* are the 4 components resulting from the wavelet decomposition, as shown in Figure 4. Equations 3.3 to 3.8 were taken from Bénéteau and Fleet (2011), and they can easily be extended to 3D cases.

The idea presented in matricial form in equation 3.3 to 3.8 can be translated to a summation, as in equation 3.9, which states that a pattern ti_T with dimensions NxN can be decomposed as:

$$ti_T = \sum_{i,l=0}^{N_j-1} a_{J,i,l}\phi_{J,i,l}^{LL} + \sum_{B\in D}\sum_{j=1}^{J}\sum_{i,l=0}^{N_j-1} \omega_{j,i,l}^{B}\psi_{j,i,l}^{B} \tag{3.9}$$

where D = {LH, HL, HH}, L and H are low-pass and high-pass filters, $N_j = N/2^j$, J is the number of scales, $N = p$ when p is even and $N = (p+1)$ when p is odd, ϕ_j is the scaling function and ψ_j^B are the wavelet functions. The coefficients a_{j-1} and ω_{j-1} are calculated by taking the inner products between a pattern (ti_T) and scaling (ϕ_j) and wavelet functions (ψ_j^B), respectively, according to the following expressions:

$$\begin{aligned} a_{j-1} &= \langle ti_T, \phi_j \rangle \\ \omega_{j-1}^B &= \langle ti_T, \psi_j^B \rangle \end{aligned} \tag{3.10}$$

Wavesim uses the Haar wavelet basis function, which is defined as follows:

$$\psi^H(x) = \begin{cases} 1 & 0 \le x < 1/2 \\ -1 & 1/2 \le x < 1 \\ 0 & otherwise \end{cases} \text{ and } \phi^L(x) = \begin{cases} 1 & 0 \le x < 1 \\ 0 & otherwise \end{cases} \tag{3.11}$$

Recall that, for categorical simulation, *M* training images are generated from the original one through indicator transformation, where *M* is the number of categories. The dimension of the approximate sub-band for a *M* categories image will be:

$$\mathrm{LN} = \left(\frac{\mathrm{N}}{2^{\mathrm{j}}}\right)^{\mathrm{d}} \mathrm{x\ M} \tag{3.12}$$

where d is the dimension of the image, and j is the number of scales. As the size of the original image is $(N^d\ x\ M)$ the reduction factor is 2^{jd}.

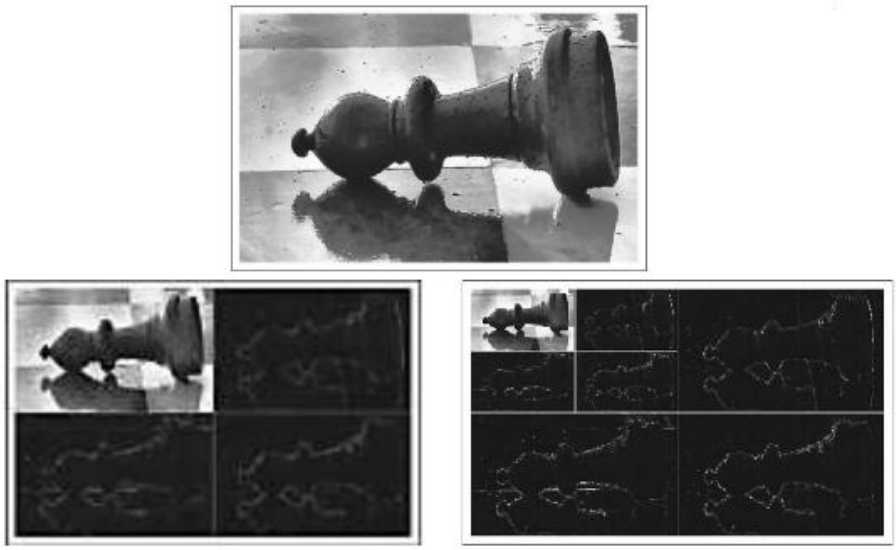


Figure 4 - Original image (top), image after one scale wavelet decomposition (left) and after two scale decomposition (right). Image taken from Bénéteau and Fleet (2011).

3.3 Classification of the Pattern Database

The classification of the pattern database is performed based on the approximate sub-band of the patterns, which had their dimensionality reduced according to *j* in the last section. This way, the classification can be done much more efficiently. The algorithm used here is the k-means clustering. The K-means clustering technique aims to divide *M* points in *N* dimensions into *k* clusters, in such a way that the within cluster variance is minimized (Hartigan and Wong, 1979). Some advantages of this algorithm is being easy to program and computationally economical. Besides Hartigan and Wong (1979), MacQueen (1967) and Ding and He (2004) provide good description and information about k-means clustering.

First, the number (k) of clusters needs to be provided. Then, *k* patterns are chosen randomly from the pattern database to be the initial centroids of the classes. Subsequently the database is entirely visited, and each pattern is compared to the initial centroids; the class corresponding to the most similar centroid is the chosen one for that pattern. Then, the *k* new centroids are calculated by averaging all elements within each of the classes. This is done iteratively, until the position of the centroids do not change anymore, which is when the final configuration of the clusters is achieved. At each iteration, the objective is to minimize the following function:

$$Z = \sum_{j=1}^{k} \sum_{i=1}^{n} \left\| t_i^{(j)} - c_j \right\|^2 \tag{3.13}$$

where $t_i^{(j)}$ represents patterns i classified in cluster j, c_j is the centroid of class j and $\left\| t_i^{(j)} - c_j \right\|^2$ is the squared Euclidean distance between $t_i^{(j)}$and c_j. So Z is the sum over all distances between the patterns and their respective class centroid. When this process is completed, each class is labeled by its prototype, which is the average over all patterns in that class (same as centroid).

3.4 Simulation

After the pattern database is divided into classes, the simulation itself can be performed. This algorithm makes use of the sequential simulation paradigm (Goovaerts, 1997) and it comprises two main steps, for each node in the random path:

1. Finding the best match class, by measuring the similarity between conditioning data event and classes prototypes;
2. Drawing a pattern from the chosen cluster to be pasted back onto the simulation grid.

First of all, a random path has to be defined, in such a way that all nodes in the grid are visited. Then, the same template used to scan the training image to generate the pattern database is placed on a node, so that the conditioning data event related to that location can be obtained. This data event has to be compared to each of the class prototypes, in

order to choose the most similar one. In this work, this is done through a distance function called L_2-norm, defined as:

$$d(x,y) = \sum_{i=1}^{3} \left\{ \omega_i * \left[\frac{1}{n_{type}} \sum_{j=1}^{n_{type}} \left(x(j) - y(j) \right)^2 \right] \right\} \tag{3.14}$$

$$\sum_{i=1}^{3} \omega_i = 1 \tag{3.15}$$

where x is the conditioning data event, y is the class prototype, n_{type} is the amount of data of a given data type and ω_i is the weight associate of data type i. Three data types are considered here: hard data, previously simulated node inside inner patch and previously simulated node outside inner patch. The weight associated with hard data is the highest, and the one associate with previously simulated node outside inner patch is the lowest. When a pattern is pasted onto the simulation grid, the user can choose to freeze an innermost portion of it, which will not be visited afterwards, whereas the nodes outside this innermost portion will be revisited and consequently, re-simulated. This innermost part is called inner patch.

As the process goes and more nodes are simulated, a situation where all nodes inside the conditioning data event are informed may happen. For these cases, distances between data event and prototypes are calculated based on their sub-band coefficients, obtained through wavelet decomposition. For these cases, the following distance function is used:

$$d(x,y) = \frac{1}{n_{approx}} \sum_{i=1}^{n_{approx}} \left(x^{approx}(j) - y^{approx}(j) \right)^2 \tag{3.16}$$

where n_{approx} is the amount of sub-band coefficients, and x^{approx} and y^{approx} are the approximate sub-band coefficients of the data event and of the prototype class, respectively. If there is a hard data within the conditioning data event, equation (3.14) will be used even if all nodes are informed. After the distance between a conditioning data event and each of the classes prototypes are calculated, the most similar one i.e., the one holding the lowest value for equation (3.14) or equation (3.16) depending on the

case, is chosen. So now, step number 2 mentioned above (drawing a pattern from chosen cluster) can be performed.

For categorical simulation, a cumulative distribution function (cdf) relative to the central node is built for that class. Then, a Monte-Carlo sampling is done in that ccdf in order to choose the category of the central node only. After that, a pattern is chosen randomly among the ones which have the central nodes belonging to the same category drawn in the Monte-Carlo sampling. For the continuous simulation, a pattern is chosen randomly from the best matched class; no cdf is generated. The chosen pattern is finally pasted back onto the simulation grid, and the inner patch defined by the user is frozen. This procedure is repeated until all the nodes defined in the random path are simulated.

In summary, the simulation method using wavelet analysis described herein has the following steps:

1. Generate the pattern database by scanning the training image with a given template.
2. Decompose the patterns using wavelet analysis.
3. Group these patterns into classes using k-means algorithm.
4. Calculate the prototypes of each class.
5. Define random path to visit all nodes to be simulated.
6. Compare data event to prototypes.
7. Choose a pattern from best matched.
8. Past it back onto simulation grid.
9. Repeat steps 6 to 8 until all nodes are simulated.
10. Repeat steps 5 to 9 to generate multiple realizations.

3.5 Generation of Training Images

Multiple Point Simulation methods are training image driven. Therefore, a training image cannot present very different features (or statistics) compared to the conditioning data, otherwise the simulated realizations will have poor reproduction of both training image's and samples' statistics. In this section, the methods used to generate both categorical and continuous training images in the case studies will be shown.

3.5.1 Categorical Training Image

The generation of categorical training images may be based on different kinds of information, depending on what will be simulated and the information the practitioner has in hands. When modeling petroleum reservoirs, for example, the following information can be used: outcrops, photographs of present day deposits or depositional systems, drawings from experts, geological interpretation, etc. (Strebelle, 2000 and 2002). Boucher (2009) discusses the importance of training images for capturing various features of the area to be modeled, as well as some intricacies in its generation.

In mining, categorical training images are usually generated through a geological interpretation using drill hole data and geological background information (Jones et. al., 2013; Boucher et al., 2013; and Goodfellow et al. 2012), as was the case of this work. Exhaustive information may also be used as in Osterholt and Dimitrakopoulos (2007), in which case information from a previously mined area from the same mine was available.

3.5.2 Continuous Training Image

The continuous training image used in the case study was generated using a Low Rank Tensor Completion (LRTC) method (Yahya, 2011, and Liu et.al. 2013). The goal of tensor completion methods is to determine values for missing elements, considering all the information available, and not only the neighboring ones, through its rank. However, rank constraint optimization problems are non-convex. Because of that, usually the trace norm is used to approximate the rank of a tensor, consisting in a convex relaxation version of the minimization problem.

The tensor completion optimization can be formulated as follows:

$$min \sum_{i=1}^{n} \alpha_i \left\| X_{(i)} \right\|_* \qquad s.t.\ X_\Omega = T_\Omega \tag{3.17}$$

where $\alpha_i \left\| X_{(i)} \right\|_*$ is the trace norm of tensor X, α are constants satisfying $\sum_{i=1}^{n} \alpha_i = 1$ and $\alpha_i \geq 0$ and $X_{(i)}$ represents the unfolded tensor along each mode. In this problem, the

matrices $X_{(i)}$ share the same entries and as a result, they cannot be optimized directly. Therefore, additional matrices $M_1, \dots, M_n$ are introduced and the problem can be relaxed:

$$\begin{gathered} min \sum_{i=1}^{n} \alpha_i \left\| M_{(i)} \right\|_* + \frac{\beta_i}{2} \left\| X_{(i)} - M_i \right\|_F^2 \\ s.t.\ X_\Omega = T_\Omega \end{gathered} \tag{3.18}$$

where $\beta_i \geq 0$ and $\|.\|_F$ is the Frobenius norm operator of a matrix. As this problem is convex, but non-differentiable, the block coordinate descent (BCD) can be used for its optimization. The basic idea of BCD is to optimize a group (block) of variables while the rest are fixed. The variables are divided in $n + 1$ blocks: $X, M_1, M_2, \dots, M_n$.

The optimal solution of X with all the other variables fixes is given by:

$$\begin{gathered} min \sum_{i=1}^{n} \frac{\beta_i}{2} \left\| M_i - X_{(i)} \right\|_F^2 \\ s.t.\ X_\Omega = T_\Omega \end{gathered} \tag{3.19}$$

and the solution to this problem is:

$$X_{i1,\dots,in} = \begin{cases} \left(\frac{\sum_i \beta_i fold_i(M_i)}{\sum_i \beta_i} \right)_{i1,\dots,in} & (i_1, \dots, i_n) \notin \Omega \\ T_{i1,\dots,in} & (i_1, \dots, i_n) \in \Omega \end{cases} \tag{3.20}$$

Finally, M_i is given by solving the following problem:

$$\begin{gathered} min \sum_{i=1}^{n} \alpha_i \left\| M_{(i)} \right\|_* + \frac{\beta_i}{2} \left\| X_{(i)} - M_i \right\|_F^2 \equiv \\ \sum_{i=1}^{n} \frac{\alpha_i}{\beta_i} \left\| M_{(i)} \right\|_* + \frac{1}{2} \left\| X_{(i)} - M_i \right\|_F^2 \end{gathered} \tag{3.21}$$

Hence, the optimal M_i is given by $D_\tau\left(X_{(i)}\right)$, where $D_\tau(.)$ is the shrinkage operator and $\tau = \alpha_i/\beta_i$.

Chapter 4

Simulation of Olympic Dam Copper Deposit, South Australia

4.1 The Deposit

Olympic Dam deposit is part of Gawler Craton, and it is located in Australia, in the center of the province South Australia, approximately 520 km NNW of Adelaide. It is a very large (6 km x 3 km x 800 m) polymetallic orebody, containing Cu, U, Au and Ag. Nowadays, it is the fourth largest producer of copper and the largest producer or uranium ("Olympic Dam Mine", InfoMine Inc.). In this case study, only copper will be considered in the simulation. Olympic Dam is a huge breccia complex, hosted by deformed and highly brecciated granite, which is slightly older than the mineralization. It is covered by 300 meter layer of flat-lying sedimentary rocks. This deposit is a copper-gold type of mineralization: it presents a complex copper mineral zoning pattern, centered on a structurally controlled barren quartz-hematite breccia. There are two types mineralization: 1) the strata-bound bornite-chalcopyrite-pyrite one, confined to the Olympic Dam formation, and 2) chalcocite-bornite in lenses and cross-cutting veins in both Olympic Dam and neighboring formations. Moving outward/downward, the following copper minerals are more common: chalcocite-bornite, bornite, chalcopyrite-bornite, chalcopyrite and chalcopyrite-pyrite, where the highest grades are usually associated with bornite ± chalcopyrite. Sulfide mineral assemblages in the Olympic Dam deposit are demonstrably in equilibrium with ubiquitous hematite (Fe_2O_3) alteration of the granite host rock that is thought to be older than the sulfide deposition. The disappearance of chalcocite in favor of chalcopyrite and subsequently bornite for pyrite mark the locations where these reactions proceed to completeness (Roberts and Hudson, 1983, Skirrow et al., 2007, Belperio and Freeman, 2004, and Hahn, 2008).

4.2 Simulation of Material Types

4.2.1 Data Set and Training Image

The drill hole data set consists of 5-m composites and the total of sulfides is calculated as the sum of the following minerals: chalcocite (Cu_2S), bornite (Cu_5FeS_4), chalcopyrite ($CuFeS_2$) and pyrite (FeS_2). As mention before, the sulfides are zoned from inner bornite ± chalcocite through bornite ± chalcopyrite to outer chalcopyrite ± pyrite. The mineral content is calculated based on copper and sulfide sulfur assays, and they are believed to be at the same level of accuracy as the underlying assays. Sulfide-bearing intervals are defined as greater than or equal to 0.05 % of total sulfides, which is a somewhat arbitrary threshold value. Table 1 shows how the material types to be simulated are defined. In this table, BN, CC, PY and CPY mean, respectively, bornite, chalcocite pyrite and chalcopyrite. Domains 1 and 2 are the most important. Domain 3 also contains important amount of copper and, finally, domain 0 is mostly waste.

Table 1 - Definition of material types used to classify data into categories. BN, CC, PY and CPY mean, respectively, bornite, chalcocite pyrite and chalcopyrite.

Geological Domain	Definition
0	Total Sulfides < 0.10%
1	(BN+CC) ≥ 0.05 and (PY+CPY) < 0.05
2	(BN+CC) ≥ 0.05 and (PY+CPY) ≥ 0.05
3	(PY+CPY) ≥ 0.05 and (BN+CC) < 0.05

Figure 5 shows the samples within the study area. The colors correspond to the definitions in Table 1. The training image is generated through a geological interpretation, as mentioned in Section 3.1, using the data in Figure 5. Figure 6 shows three sections of the training image generated, whereas Figure 7 displays the spatial configuration of material types 1 and 2, so that it is possible to see the spatial complexity of these units. The training image is discretized in a grid of 15m x 15m x5m, resulting in 120 x 150 x 170 nodes in *X*, *Y* and *Z* directions, respectively and a total of 2,813,670 nodes. The deposit to be simulated is discretized the same way as the training image.

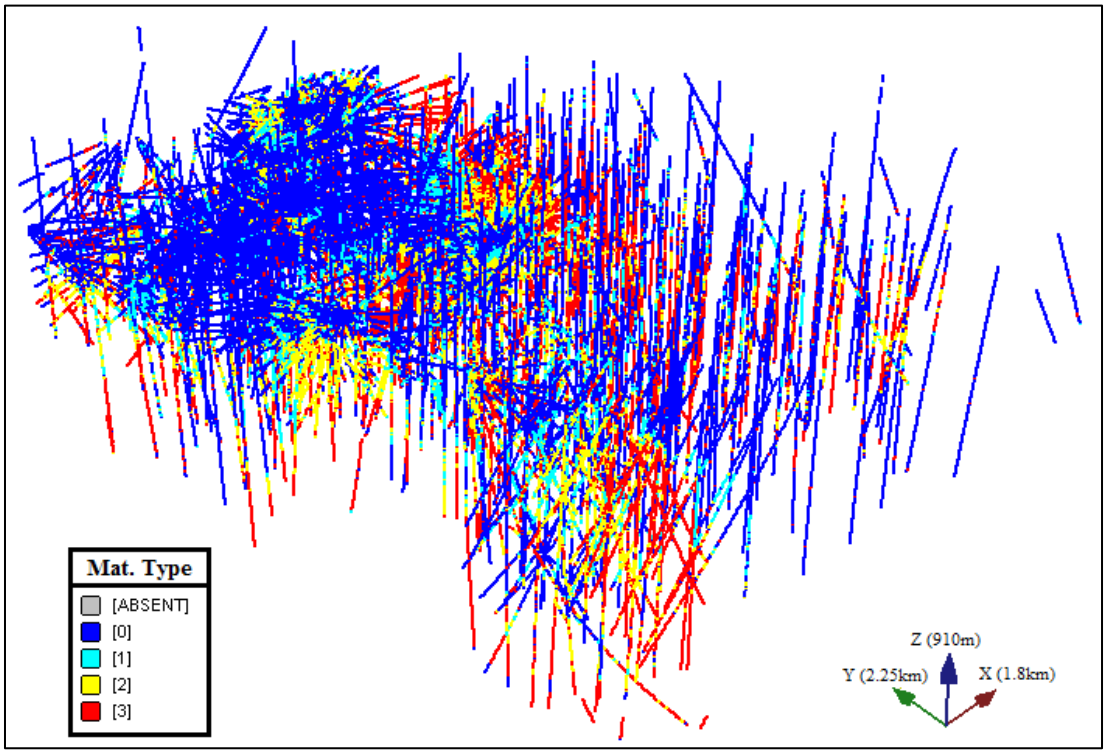


Figure 5 - Drillhole samples of Olympic Dam copper deposit, colored according to material types defined in Table 1.

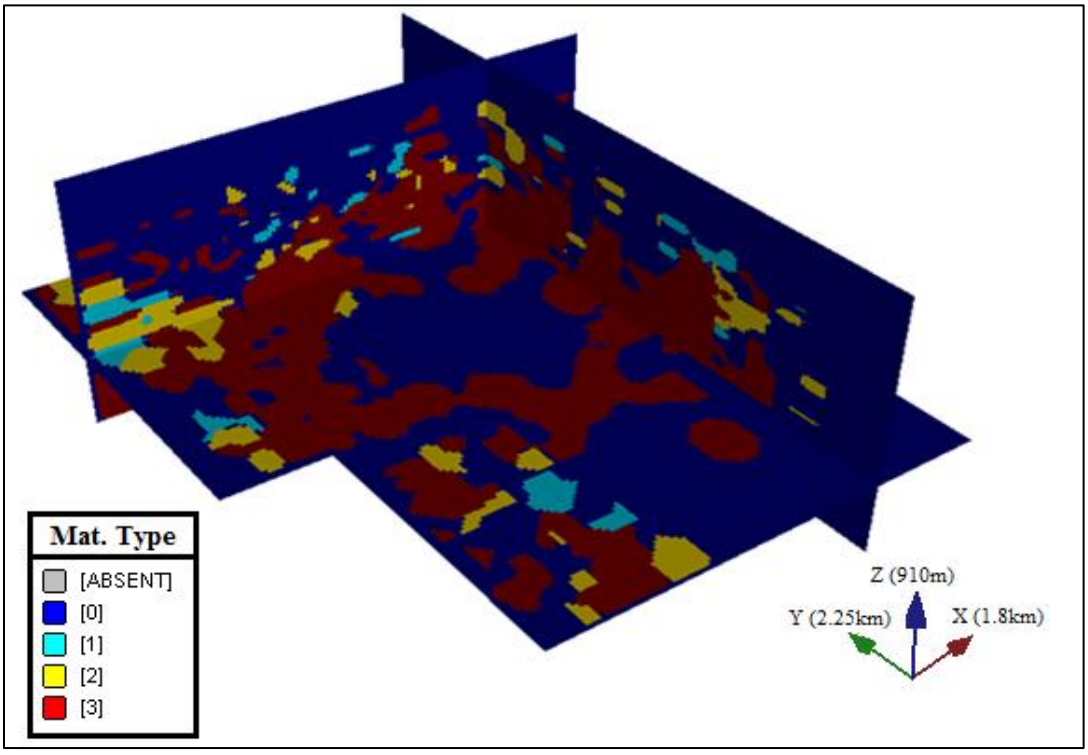
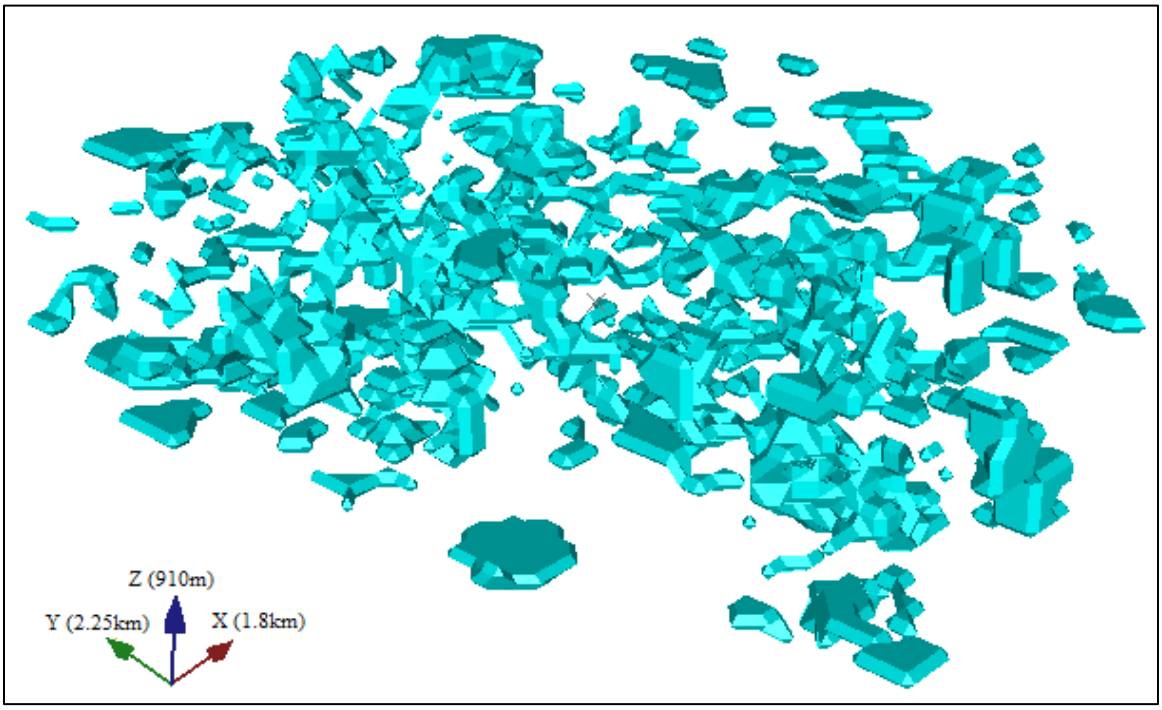
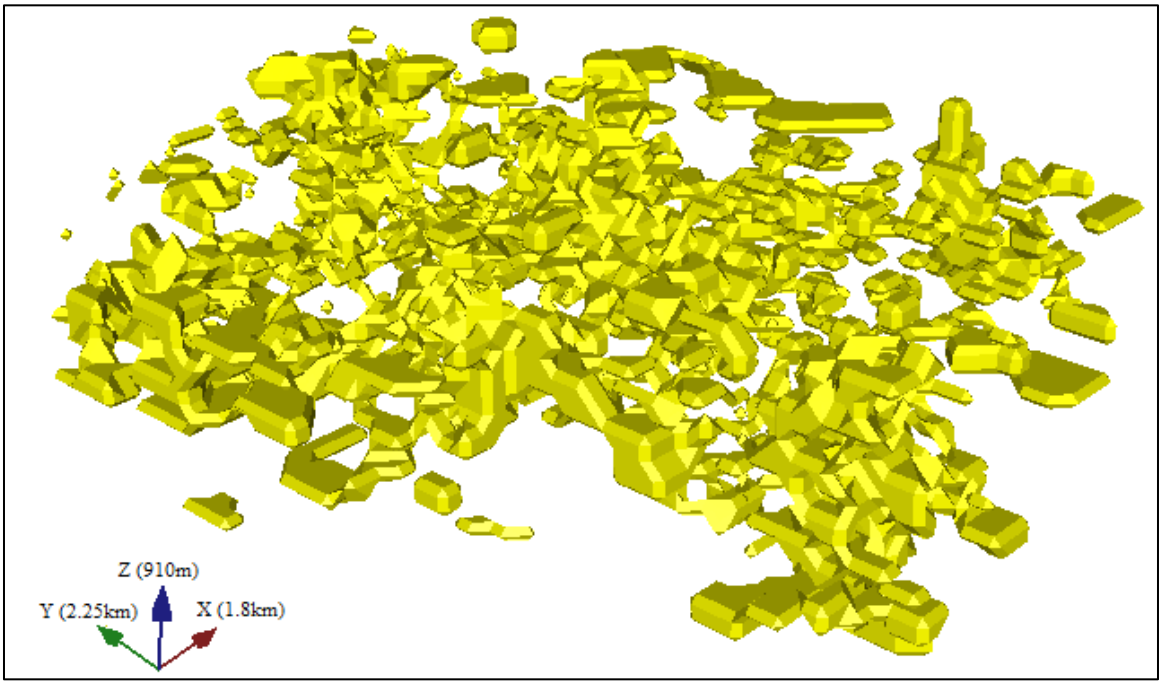


Figure 6 - Cross-sections of the training image (X = 96, Y = 118 and Z = 51).



a)



b)

Figure 7: Spatial visualization of wireframes representing domains 1 and 2, in the training image: a) domain 1; and b) domain 2.

4.2.2 Simulation Results

The number of clusters and template sizes are defined through trial and error. It is important to note that, the larger the template and the number of clusters, the better the results tend to be, however the run time of the algorithm becomes longer. Hence, the values to be used for these parameters are the ones showing the best trade-off between quality of results and computational time. Table 2 shows the parameters that were chosen through testing to simulate Olympic Dam deposit's material types.

Table 2 - Parameters used in simulation.

Parameter	Value
Template	11 x 11 x 5
Inner Patch	5 x 5 x 3
Number of Clusters	500
Number of Realizations	20

Template, inner patch and number of clusters are defined in sections 2.1, 2.4 and 2.3, respectively. Figure 8 displays 3 sections of two simulated realization. These sections are the same ones shown for the training image in Figure 6. Comparison between training image and simulations section shows that both present the same regional configuration: same categories tend to appear in the same regions. However, as expected, the simulated realizations are less smooth, presenting more variable patterns. Figure 9 shows the histogram of the 3 mineralized categories in the 20 simulations, compared to training image and declustered samples. According to Figure 9, simulations reproduce well the proportions of the 3 material types. It also shows the effect of the training image on the simulations' histograms. Since in this case study, there are a relatively large amount of hard data, the simulations reproduced data's histogram as opposed to training image's one. Figure 10, Figure 11 and Figure 12 show the direct variogram for the 3 material types, in 3 directions: East-West (EW), North-South (NS) and vertical. The cross-correlograms between these 3 material types for EW, NS and vertical directions are displayed in Figure 13, Figure 14 and Figure 15. All results suggest a reasonable reproduction of training image's and samples' variograms and cross-variograms by the

simulated realizations of the 3 material types at Olympic Dam. As the simulation was performed with a training image driven method, it was expected that simulations' variograms were closer to training image's ones. However, as there are a large amount of samples in this case study, they are more similar to data's variograms. Validations are also performed in terms of high-order statistics, which are analyzed through cumulant maps, as shown in De Iaco and Maggio (2011). In order for the cumulant maps to be calculated, spatial templates have to be defined. In this case, an L-shape template was used to calculate 3^{th} order cumulant {(1,0,0); (0,1,0)} and for the 4^{th} one, the template {(1,0,0); (0, 1, 0); (0,0,1)} was used. Figure 16 and Figure 17 show cumulant maps calculated considering material types 1, 2 and 3. As it may be noted, the simulations' cumulant maps can be seen as being "in between" samples' and training image's ones: they show a similar general pattern to latter, but with lesser continuity, due to influence of the former.

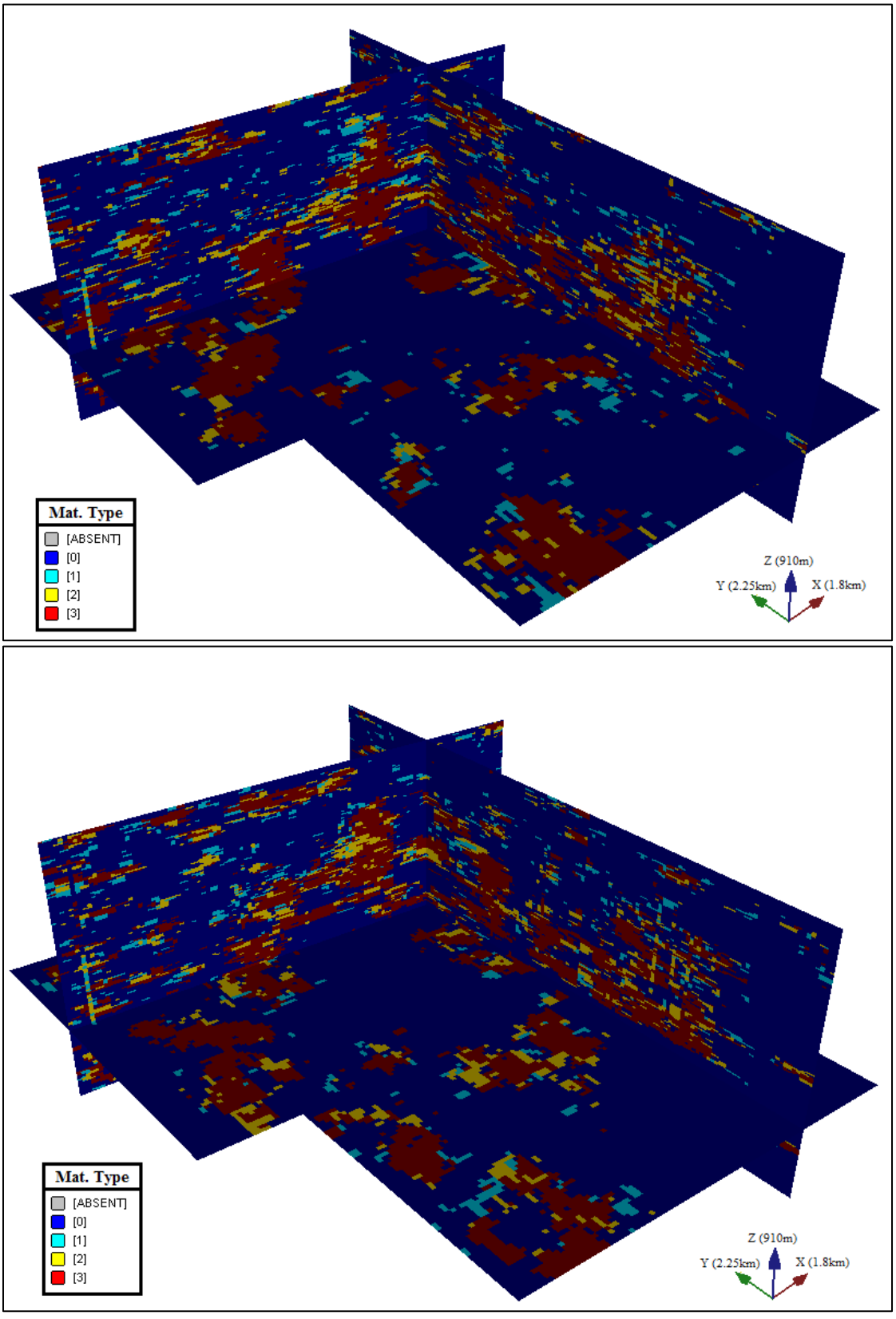


Figure 8 - Two simulations of material types. The sections are the same than in Figure 6.

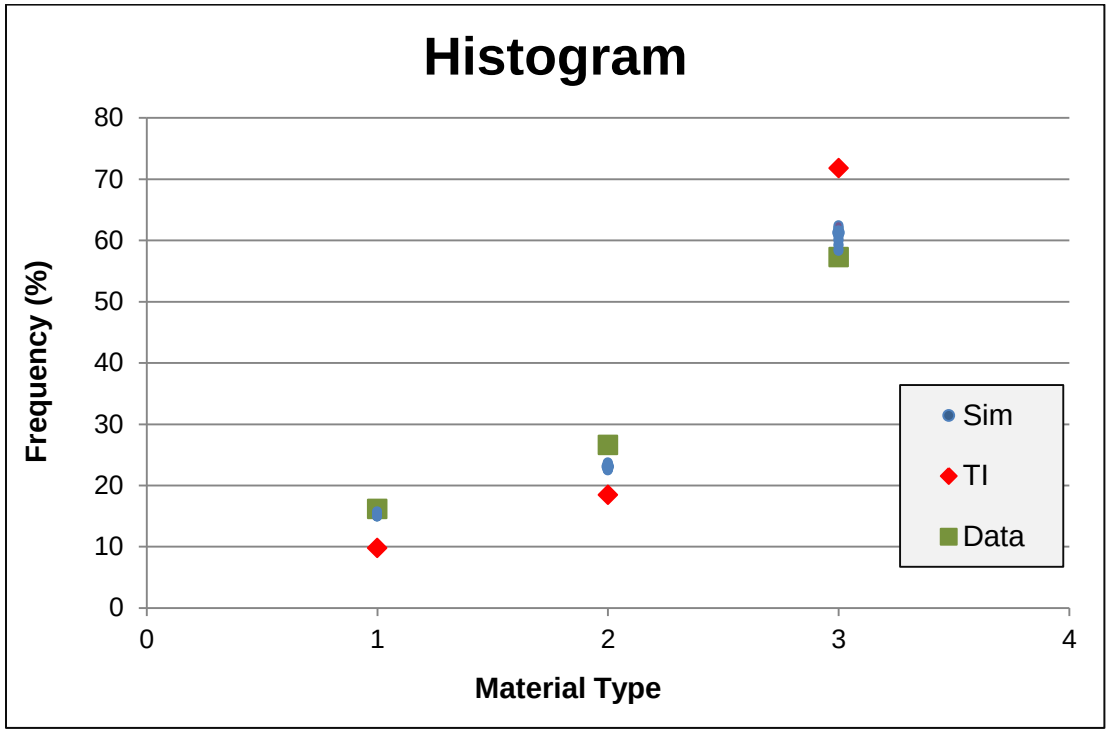


Figure 9 - Histogram of 20 material types' simulations compared to data and training image.

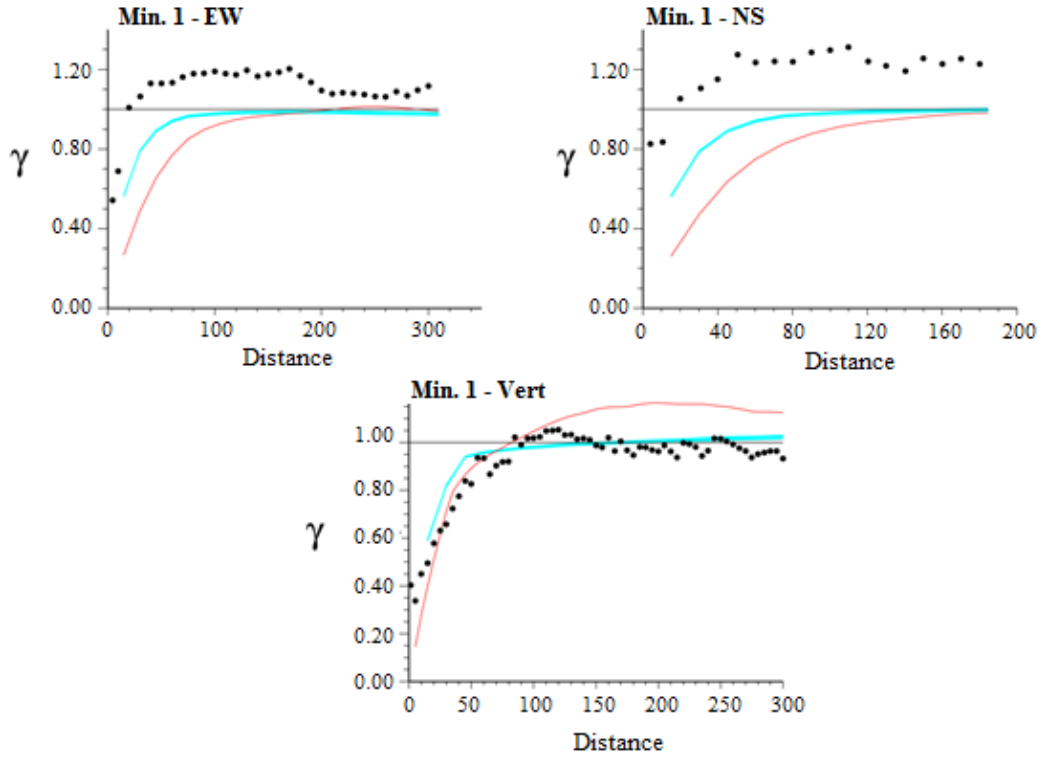


Figure 10 -Variograms of 20 simulations of material types (light blue line), samples (black dot) and training image (red line), for material type 1.

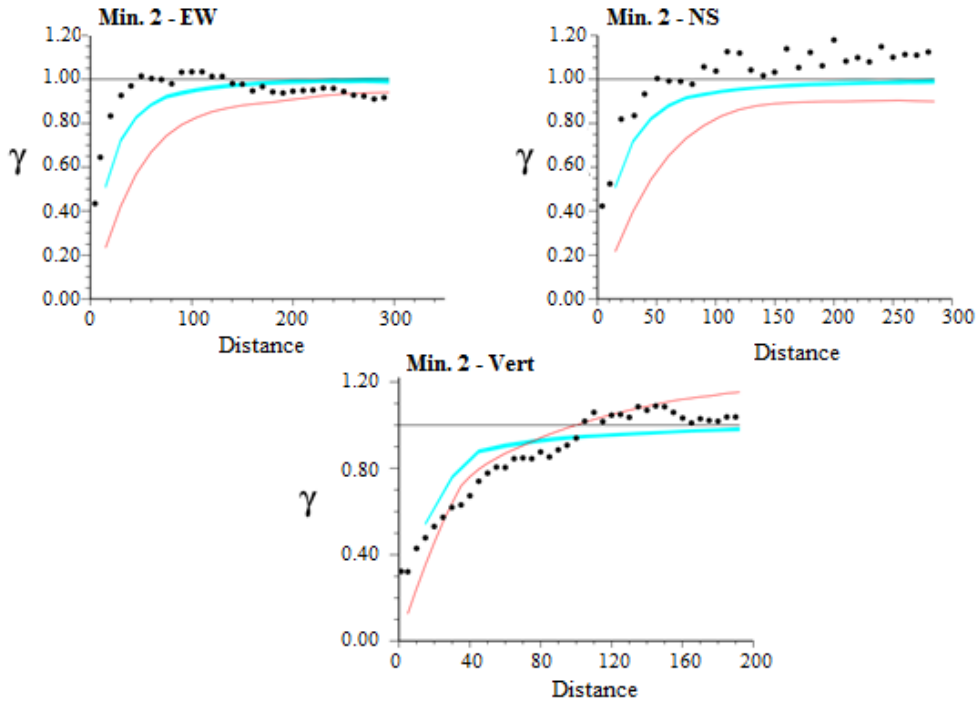


Figure 11 - Variograms of 20 simulations of material types (light blue line), samples (black dot) and training image (red line), for material type 2.

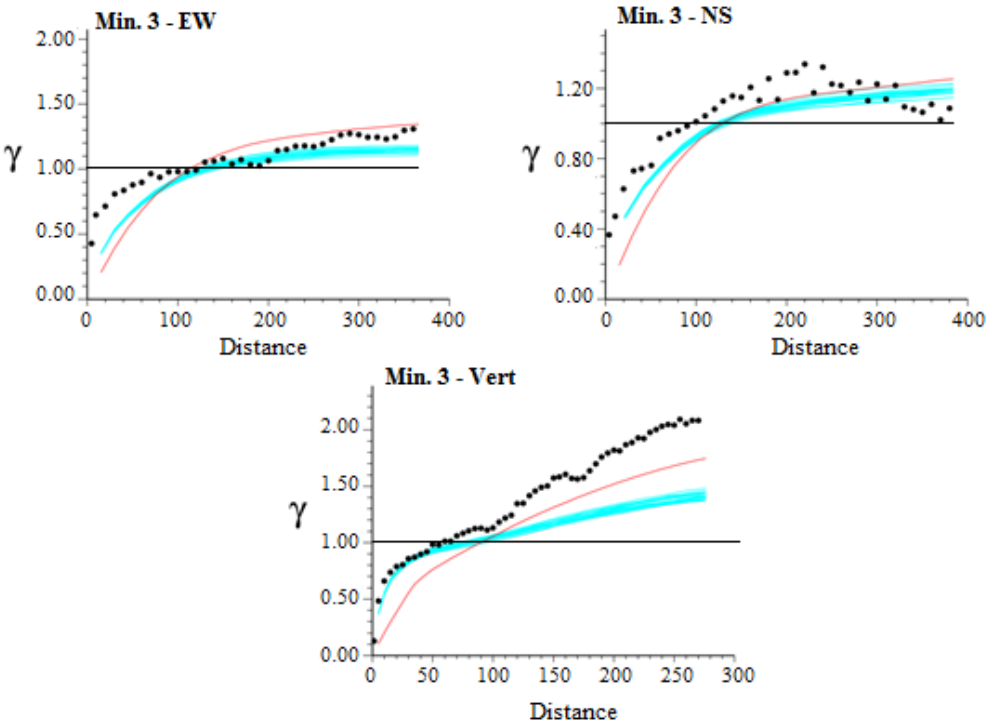


Figure 12 - Variograms of 20 simulations of material types (light blue line), samples (black dot) and training image (red line), for material type 3.

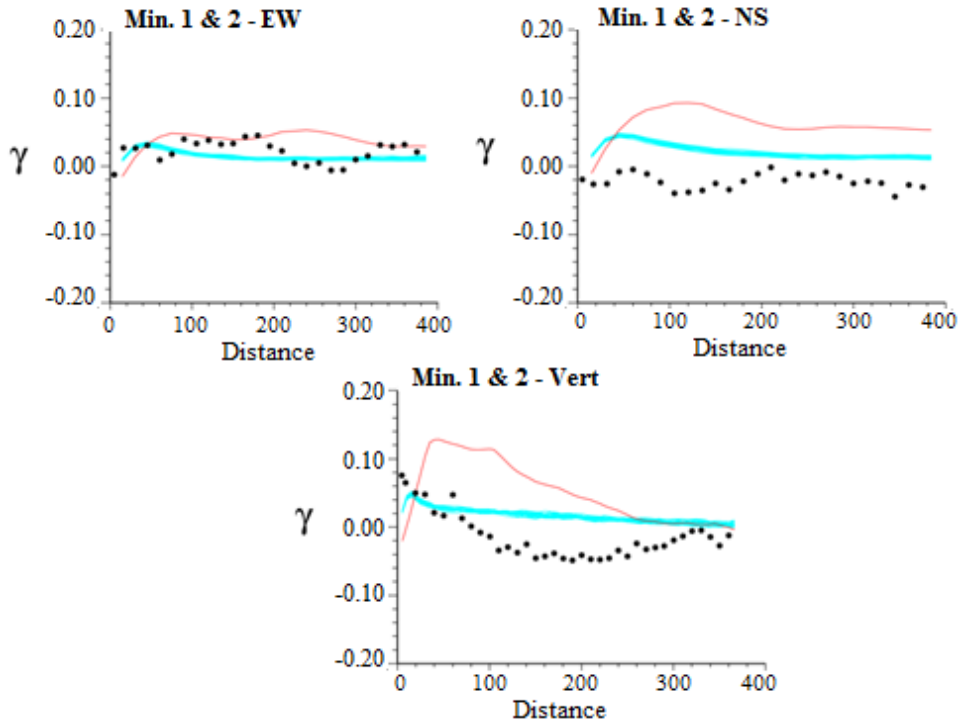


Figure 13 - Cross-correlograms of 20 simulations of material types (light blue line), samples (black dot) and training image (red line), for material types 1 and 2.

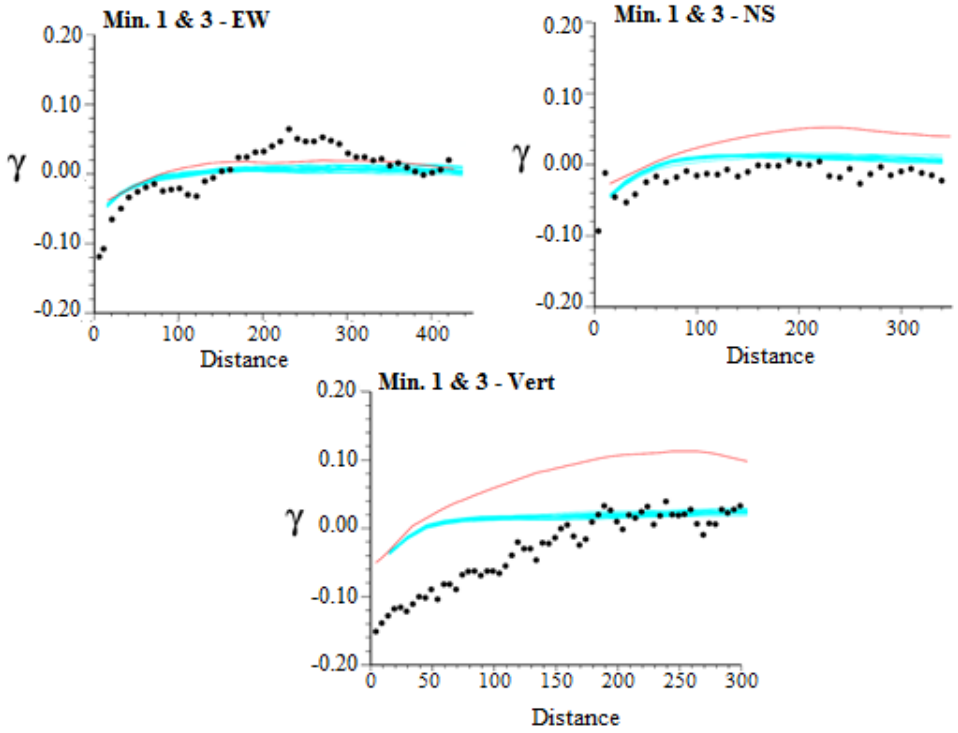


Figure 14 - Cross-correlograms of 20 simulations of material types (light blue line), samples (black dot) and training image (red line), for material types 1 and 3.

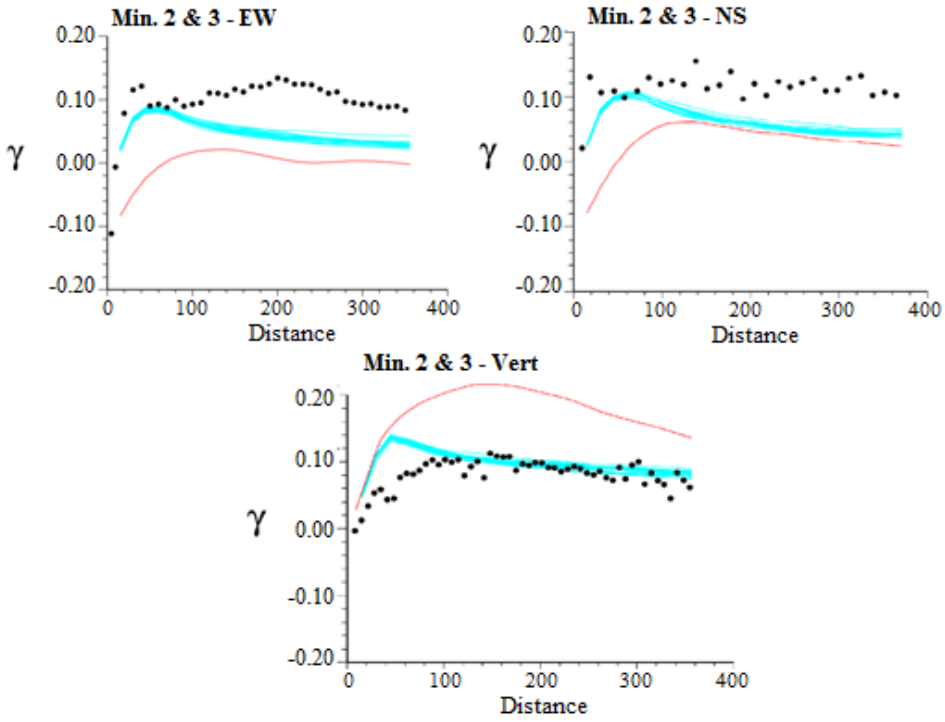


Figure 15 - Cross-correlograms of 20 simulations of material types (light blue line), samples (black dot) and training image (red line), for material types 2 and 3.

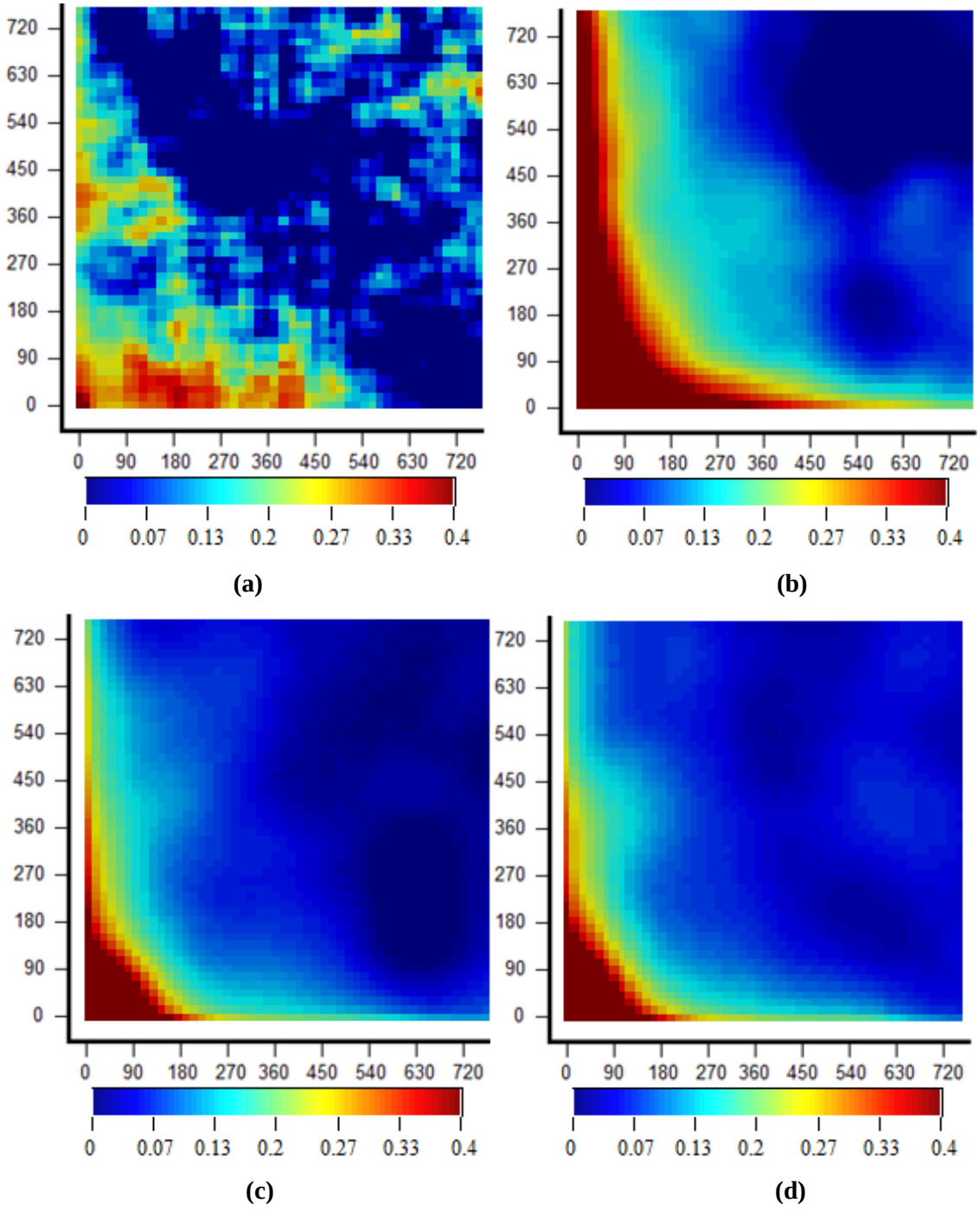


Figure 16 - Third-order cumulant maps for a) samples; b) training image; c) and d) two realizations. Direction of cumulant: {(1,0,0); (0,1,0)}.

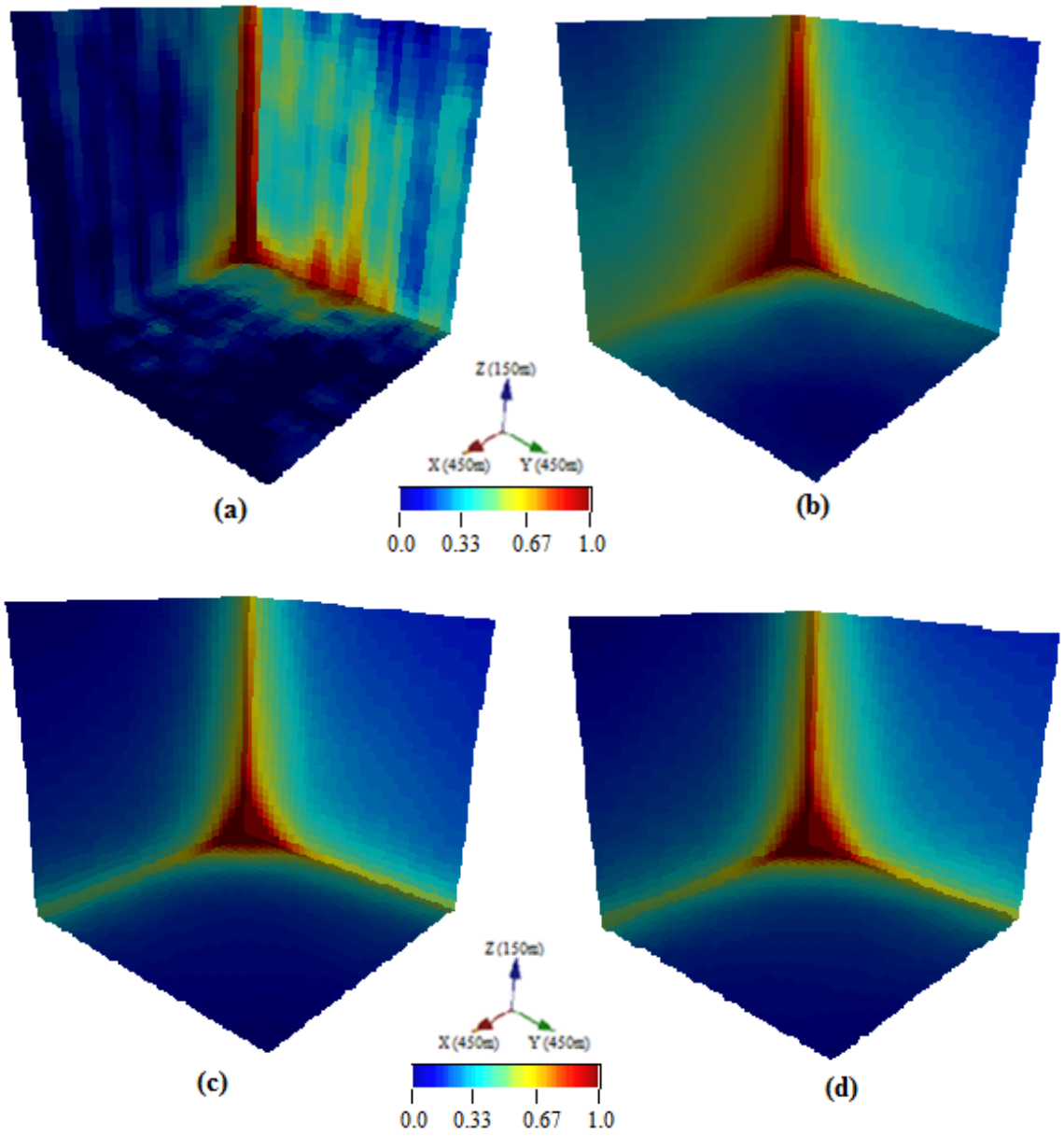


Figure 17 - Fourth-order cumulant maps for a) samples; b) training image; c) and d) two realizations. Direction of cumulant: {(1,0,0); (0, 1, 0); (0,0,1)}.

4.3 Simulation of Copper Grades

4.3.1 Data Set and Training Image

Having defined material types of Olympic Dam deposit through categorical simulation, copper grades are simulated within these boundaries, using the same wavelet based method. Figure 18 and Table 3 show, respectively, the histograms and statistics of copper

grade for each material type, separately. Material types 1 and 2 are the richest ones; type 3 also contains high copper grade samples; and finally type 0 is mostly waste and, as result, it is not simulated.

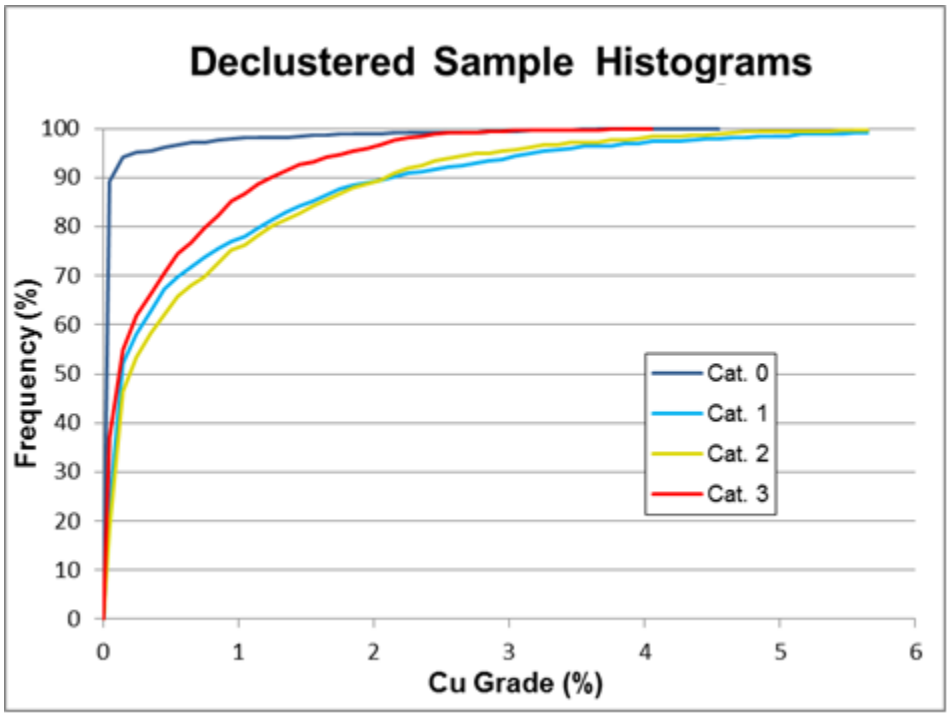


Figure 18 - Declustered copper grade histograms for each geological domain.

Table 3 - Statistics of copper grade per domain.

Statistics	Cat0	Cat1	Cat2	Cat3
Mean	0.104	0.733	0.741	0.453
Stand. Dev.	0.372	1.137	1.012	0.614
Variance	0.139	1.292	1.023	0.377
Kurtosis	67.301	7.318	5.854	5.250
Skewness	7.655	2.567	2.249	2.139
Minimum	0.005	0.005	0.005	0.005
Maximum	4.58	7.66	6.866	4.08
10th Perc.	0.007	0.020	0.024	0.010
25th Perc.	0.014	0.109	0.123	0.071
50th Perc.	0.030	0.200	0.256	0.180
75th Perc.	0.059	0.858	0.993	0.621
90th Perc.	0.114	2.184	2.125	1.259

The training image used in this case study is generated through low rank tensor completion (Liu et.al., 2013), as described in Section 3.2. The algorithm works better for higher density of samples, hence the training image was generated based on the densest sampled part of the deposit, as shown in Figure 19. Figure 20 displays 3 cross-sections of the training image. Its dimensions in *X*, *Y* and *Z* direction are 80, 80 and 111, respectively. The continuous simulation is performed as follows. To simulate material type 1, for example, only patterns which lie inside the wireframe related to this material, as defined in Figure 7 (a), are taken from the training image. Then, these patterns are pasted on the simulation grid, but only on the nodes which lie inside the simulated wireframe related to this material type, according to the simulation of material types in the previous section. The same procedure is then repeated to simulate copper grades for categories 2 and 3.

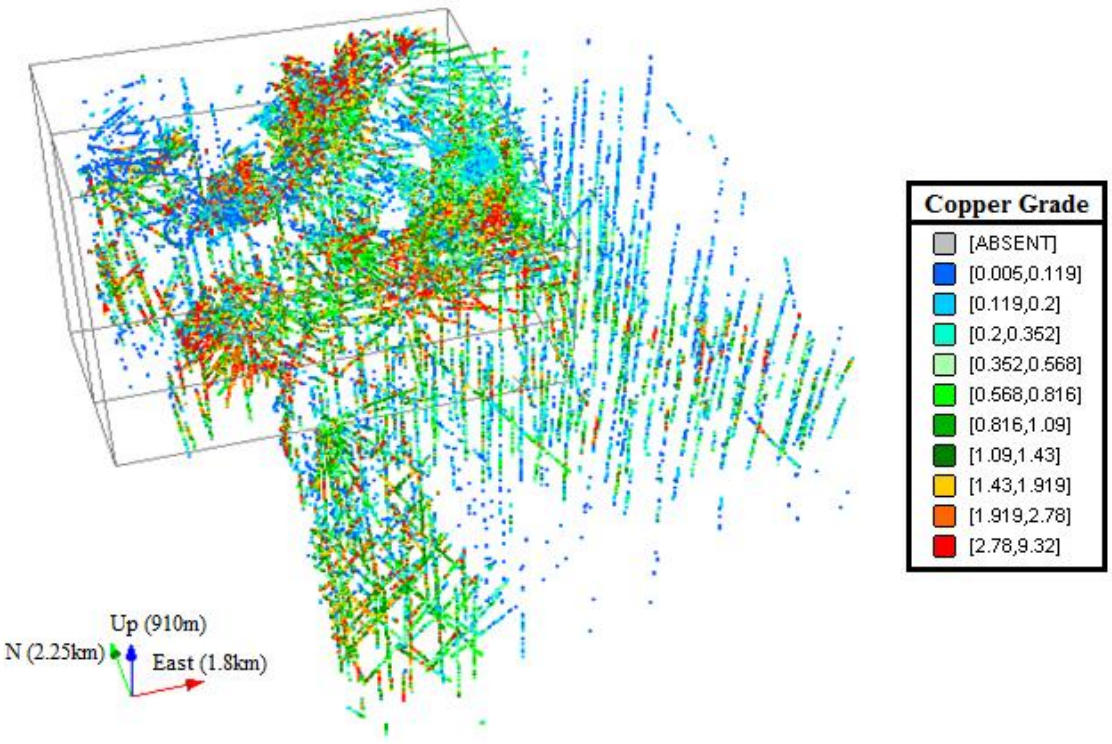


Figure 19 - Data set colored regarding copper grade. Samples used to build training image is highlighted.

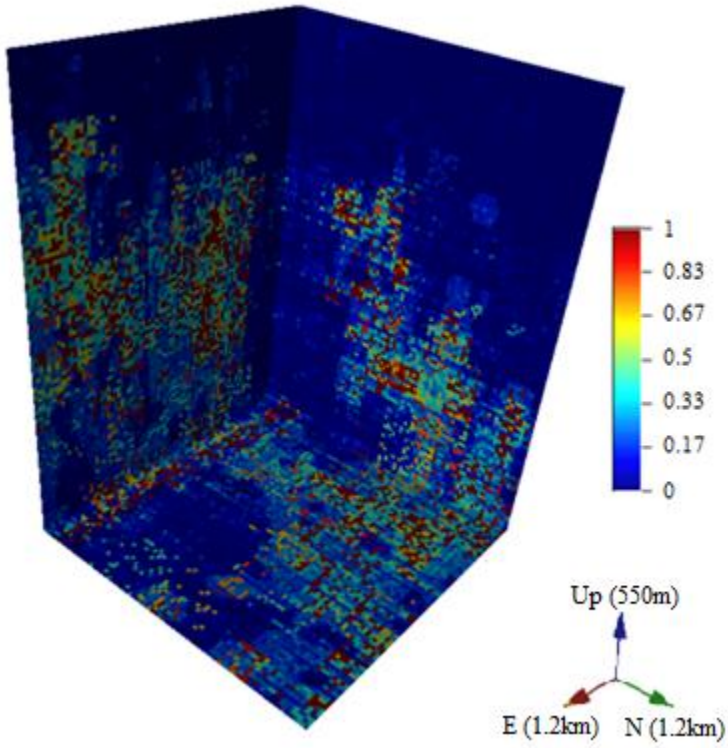


Figure 20 - Cross-sections of the continuous training image (X = 0, Y = 0 and Z = 0).

4.3.2 Simulation Results

The parameters used to simulate the copper grades within each material type are displayed in Table 4. As in the simulation of material types, the simulation grid is discretized by 15m x 15m x 5m, resulting in 120 x 150 x 170 nodes in *X*, *Y* and *Z* directions respectively, and a total of 2,813,670 nodes.

Table 4 - Parameters used for the copper simulations.

Parameter	Value
Template	11 x 11 x 9
Inner Patch	5 x 5 x 5
Number of Clusters	500
Number of Realizations	3

The template size and the number of clusters to be used are defined based on trial and error. Several values are tested, their results are analyzed and the values of these parameters which present the best trade-off between good quality results and

computational time are chosen. The tendency is that, the bigger the template and number of clusters, the better the results are and the longer it takes to run. Figure 21, Figure 22 and Figure 23 show cross-sections in *X*, *Y* and *Z* directions respectively for 2 simulations, at point support. The 3 material types are being displayed together. The gray color corresponds to the category 0, which was not simulated. It is possible to see that, despite of presenting different shapes, due to be related to different categorical simulations, the copper simulations above tend to present the same spatial pattern; i.e. coincident rich and poor areas. Figure 24, Figure 25 and Figure 26 display the histograms of the 3 simulated realizations compared to training image and hard data, for material types 1, 2 and 3, respectively. They show that the simulated copper realizations present very similar histograms, compared to both samples and training image. It is noteworthy that the histograms of samples and training image are very similar: it is important to multiple-point simulation techniques, including the method used herein, that the training image presents similar statistics to the hard data. Otherwise, the simulated realizations may show conflicting features. Figure 27, Figure 28 and Figure 29 show the copper grades variograms in the east/west, north/south and vertical directions for material types 1 and 2 and 3, respectively. Simulated realizations show good reproduction of samples' and training image's variograms. Here, it is possible to note, again, the tendency of realizations to present their variograms in between the ones of samples and training image. Besides, the variograms of training image and samples are very similar to each other. Figure 30 and Figure 31 show, cumulant maps of copper grade regarding each material type for samples, training image and two simulated realizations. They show how similar realizations' maps are compared to training image's ones, but with less continuity. This is due to the influence of more discontinuous samples' maps over their cumulant maps. When comparing samples' cumulant maps to simulation's and training image's ones, it is important to recall that the samples contains much less replicates for each lag. This may cause it to have some local artifacts, such as the discontinuities seen in the Figure 30 and Figure 31, which probably would not happen if more data were available.

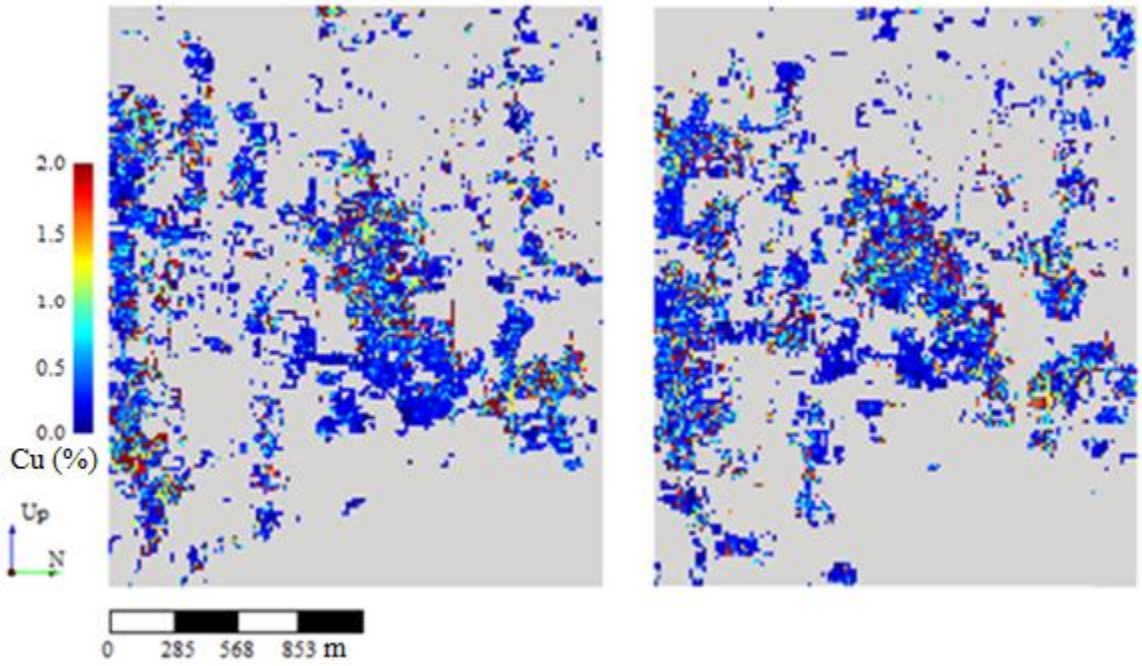


Figure 21 - Cross-section (X = 40) showing realizations of copper grades (%), for 2 simulations.

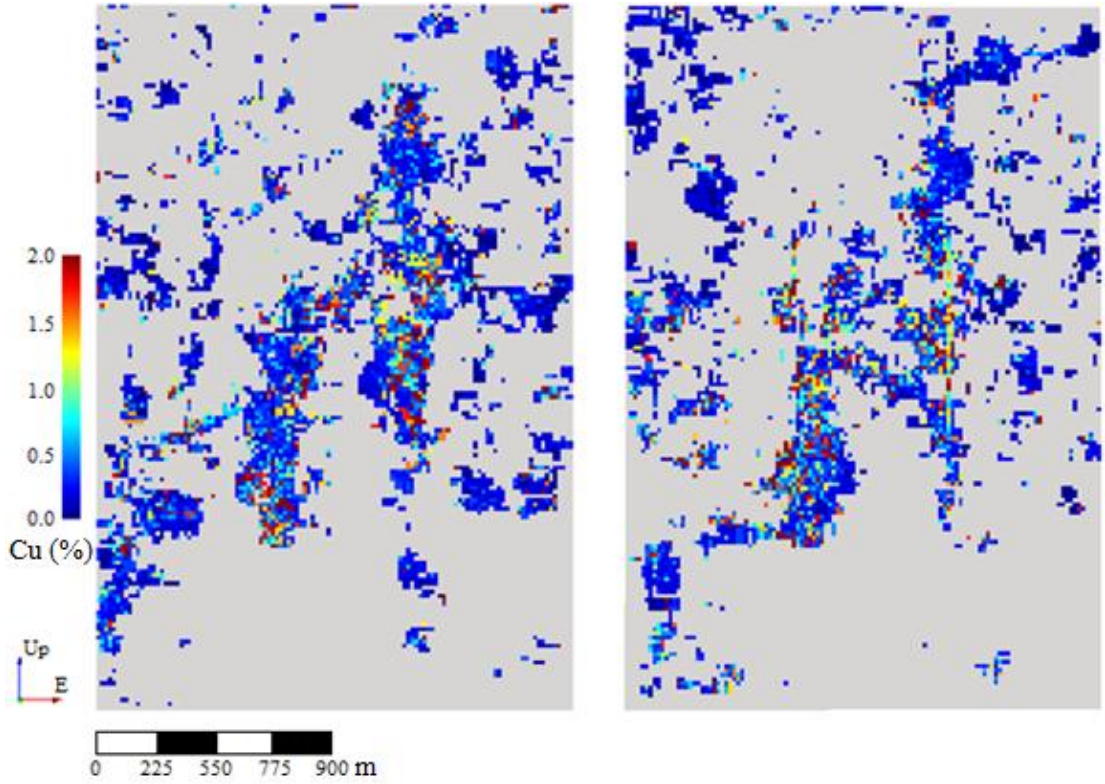


Figure 22 - Cross-section (Y = 120) showing realizations of copper grades (%), for 2 simulations.

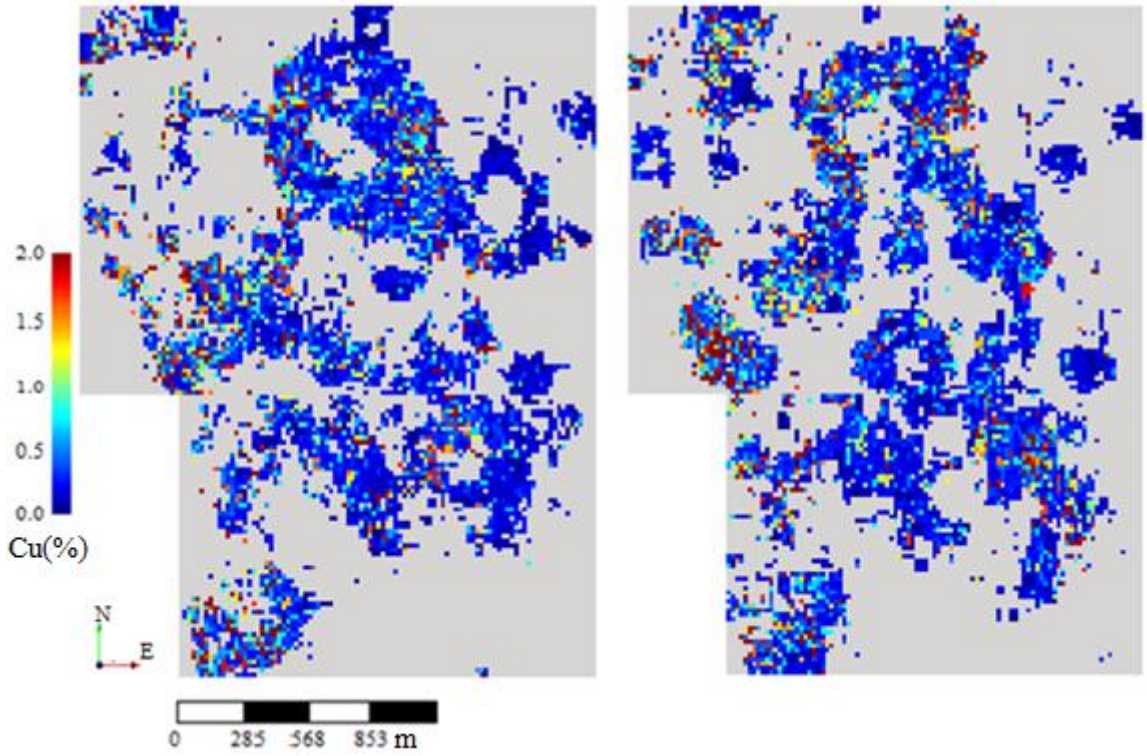


Figure 23 - Horizontal view (Z = 80) showing realizations of copper grades (%), for 2 simulations.

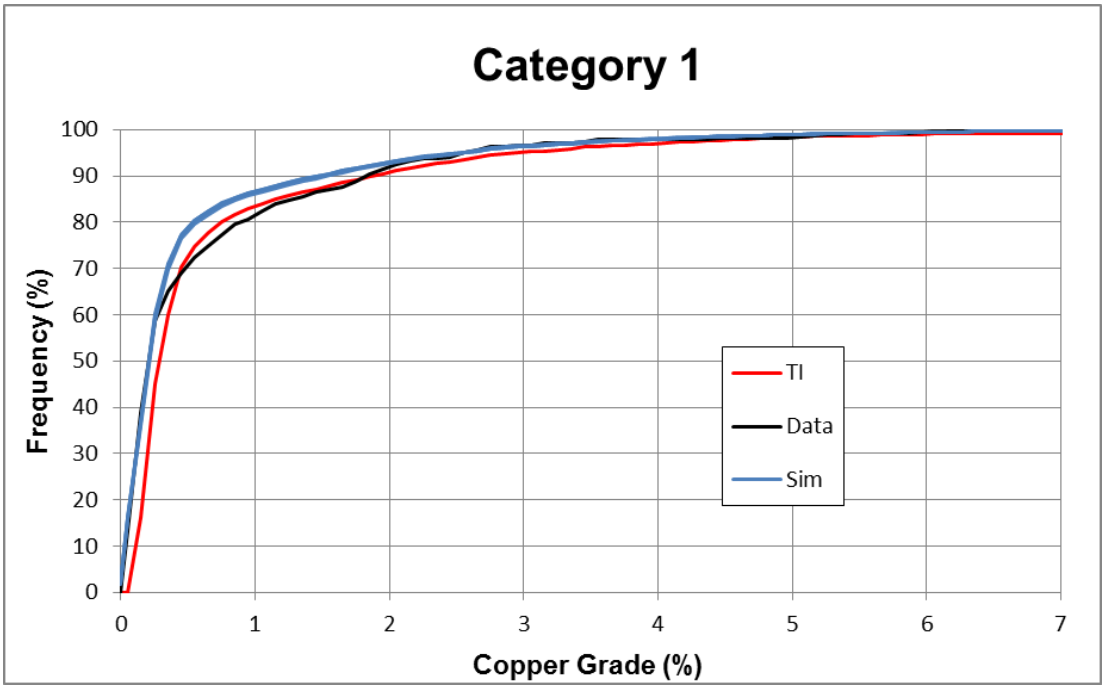


Figure 24 - Copper histogram of material type 1. Comparison between training image, hard data and simulations.

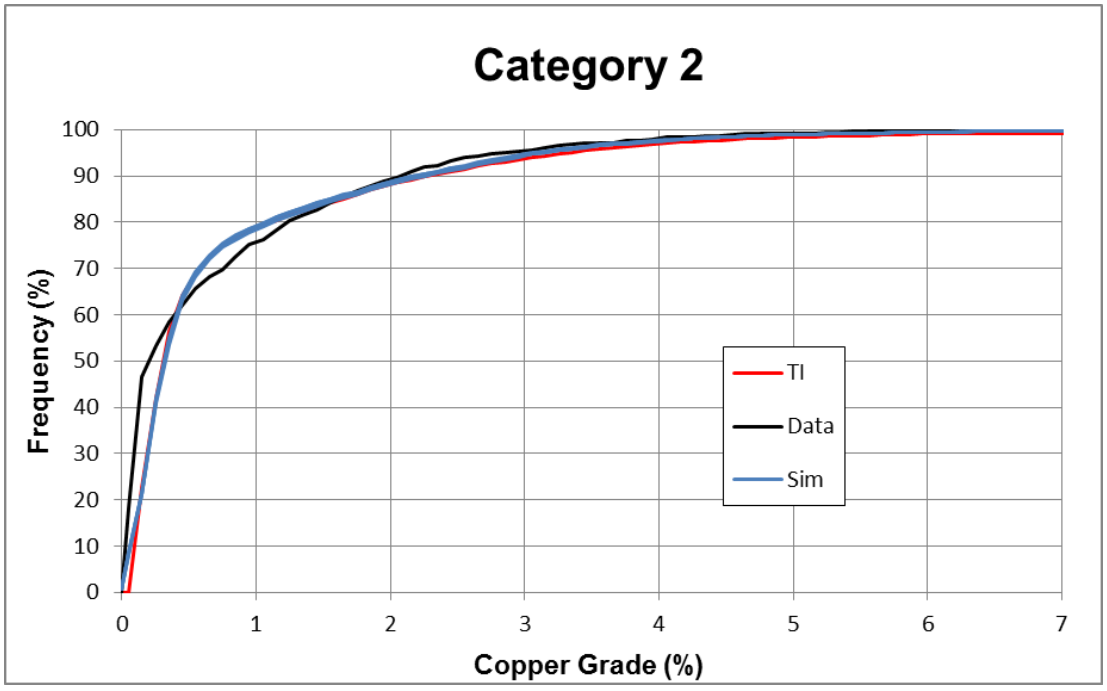


Figure 25 - Copper histogram of material type 2. Comparison between training image, hard data and simulations.

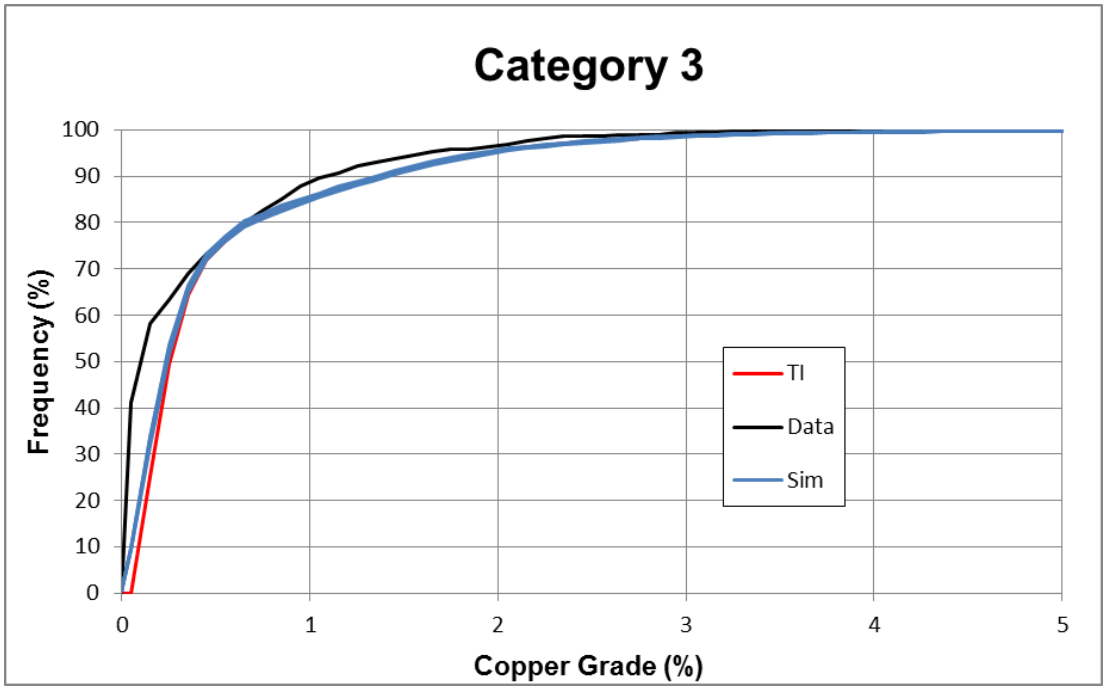


Figure 26: Copper histogram of material type 3. Comparison between training image, hard data and simulations.

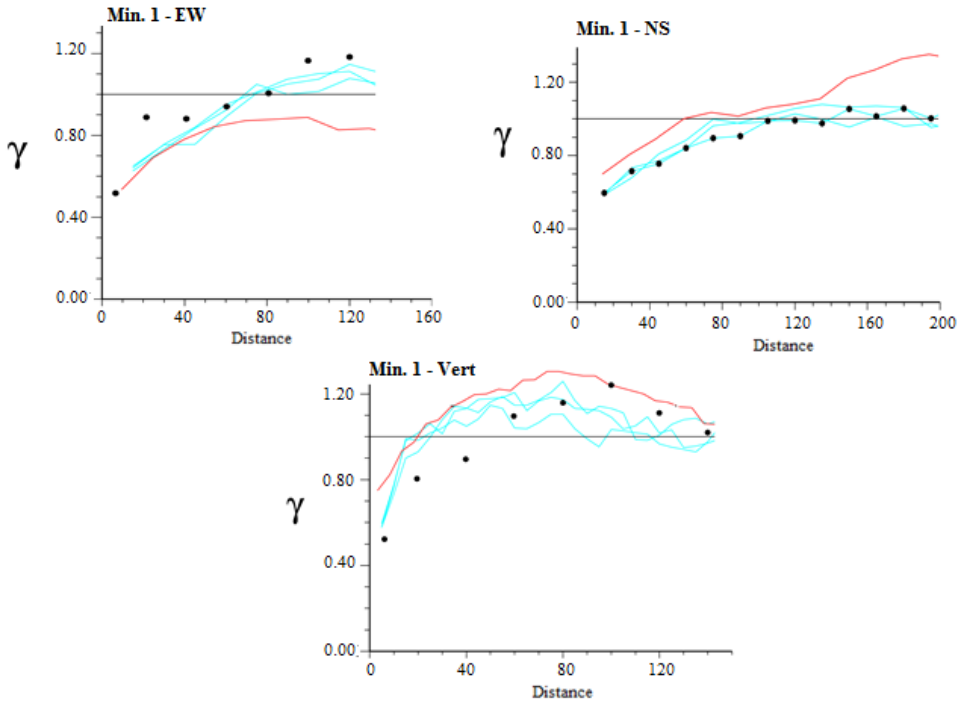


Figure 27 - Copper grade variograms of 3 simulations (light blue line), samples (black dot) and training image (red line) for material type 1.

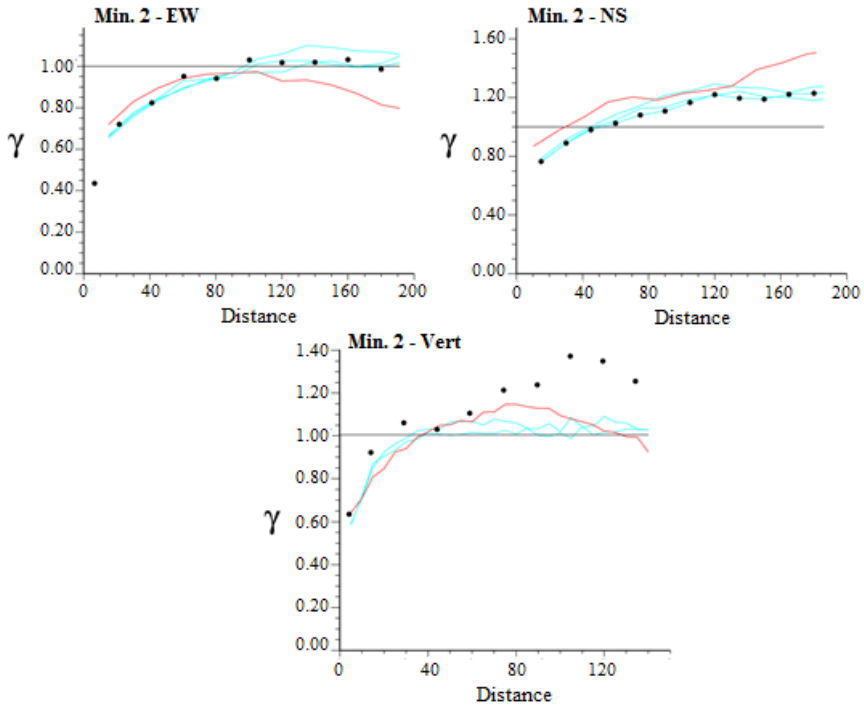


Figure 28 - Copper grade variograms of 3 simulations (light blue line), samples (black dot) and training image (red line) for material type 2.

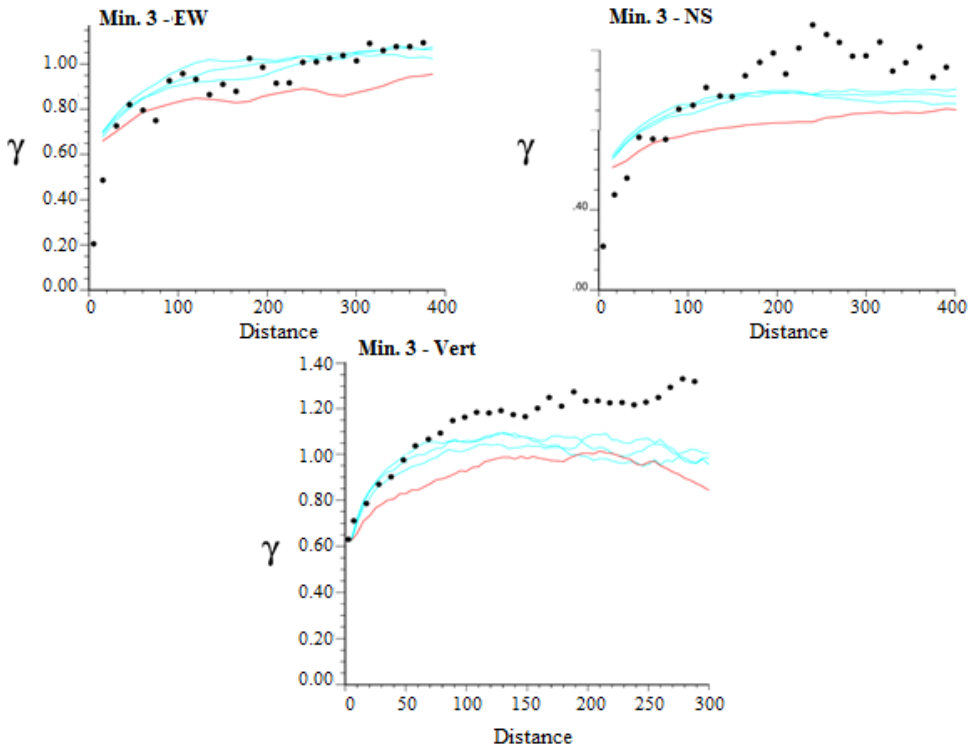


Figure 29 - Copper grade variograms of 3 simulations (light blue line), samples (black dot) and training image (red line) for material type 3.

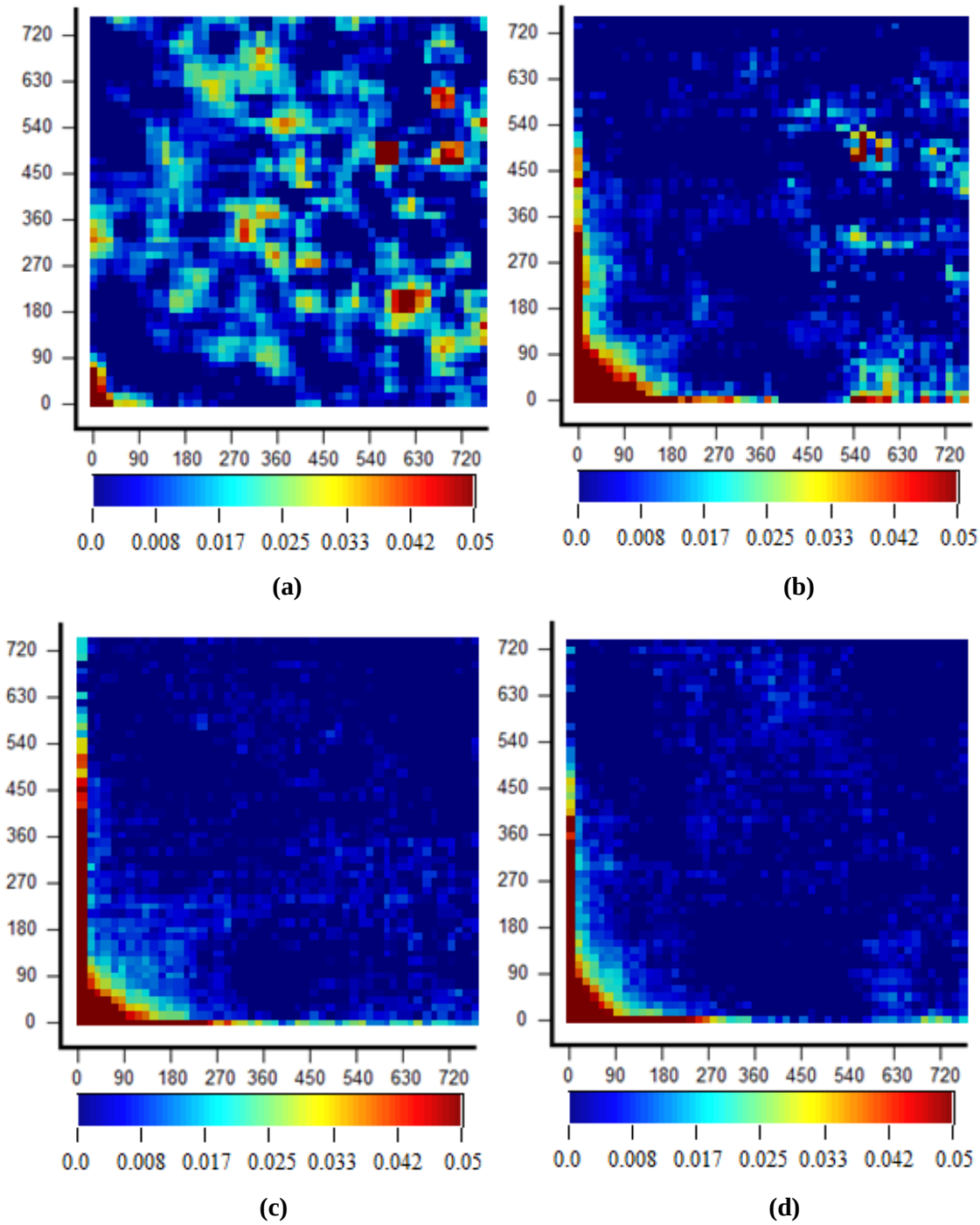


Figure 30 - Copper grade third-order cumulant maps of a) training image; b) and c) two realizations. Direction of cumulant: {(1,0,0); (0,1,0)}.Distance in meters.

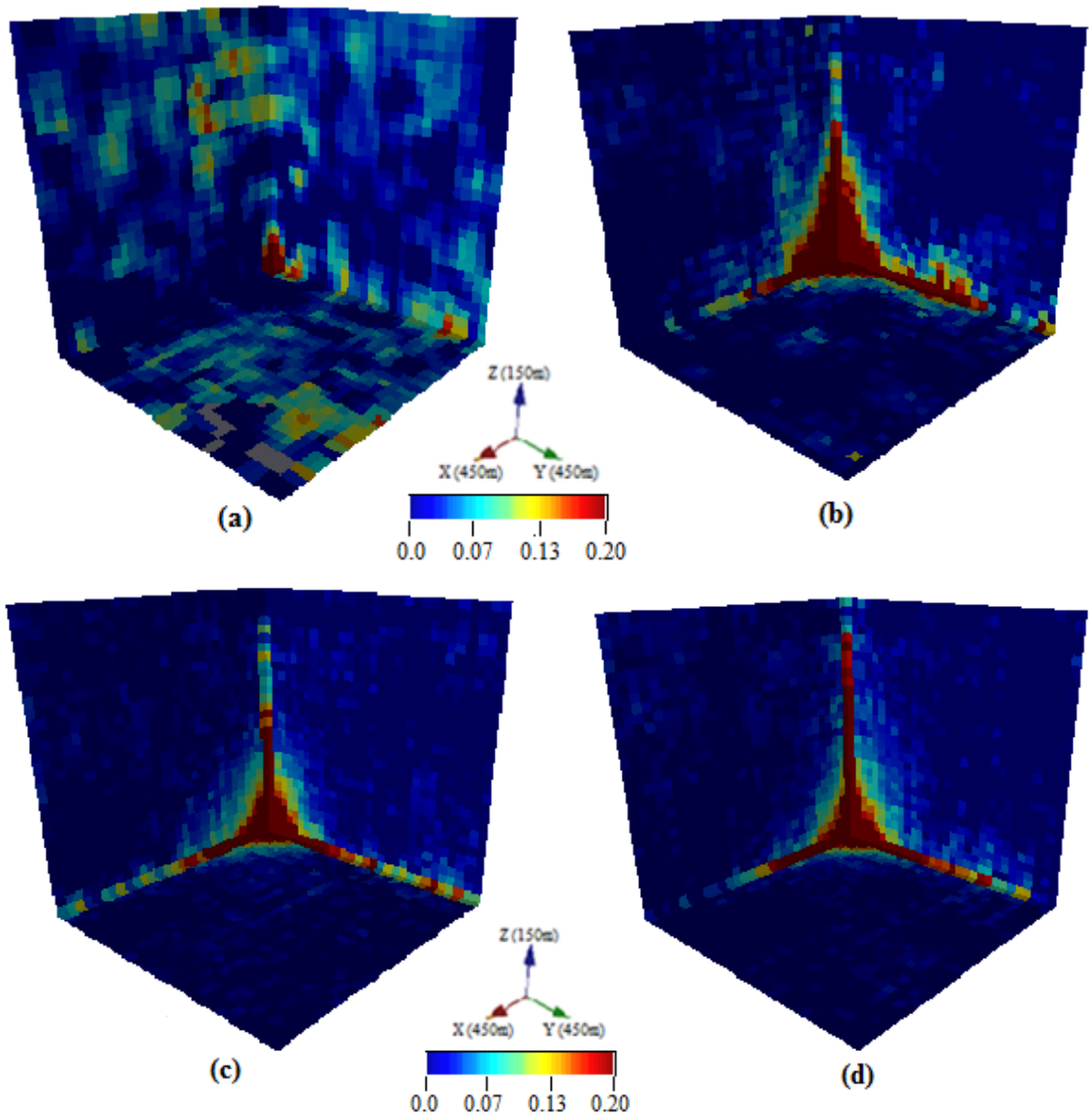


Figure 31 - Copper grade fourth-order cumulant maps of a) training image; b) and c) two realizations. Direction of cumulant: {(1,0,0); (0,1,0); (0,0,1)}.

Chapter 5

Conclusions and Future Work

In this thesis, a multiple-point statistics (MPS) simulation method for mineral deposits, which is based on wavelet analysis, is presented and tested in a full field case study, at Olympic Dam Copper Deposit, in South Australia.

Firstly, the literature review about main two-point statistics simulation methods used in the past and how they evolved to the new MPS methods is described. Past work mentioned in this chapter compared these techniques and show a better performance of the latter to model orebodies.

Then, the MPS simulation method based on wavelet analysis used herein and the techniques used to generate both categorical and continuous training images are described. Finally, the application of the simulation method to Olympic Dam copper deposit located in Southern Australia is presented, in order to evaluate both volumetric and grades uncertainty. Olympic Dam presents 4 different material types and the training image used to simulate the orebody model is obtained through geological interpretation. Then, 3 of those categorical realizations are retained for the simulation of copper grades, which is performed considering one material type at a time. The continuous training image was generated through low rank tensor completion. The result is a set of equiprobable orebody models which accounts for volumetric and grade uncertainty. The validation of the simulations' results is performed for low order statistics (histograms and variograms) and also for high-order statistics, through third and fourth-order spatial cumulant maps. This validation suggested that method tested herein can be successfully applied to real sized deposits, since Olympic Dam was discretized in 2,813,670 nodes. In all the cases, the statistics of the realizations were compared to both training image's and samples' statistics. This comparison showed an interesting point, which is common to all multiple point simulation methods, since they are training image driven: the statistics values of the realization tend to be in between training image's and data's ones. Because of that, conflicting information between training image and data samples has a negative

impact on the simulation's results. Therefore, the training image has to be representative of the deposit to be simulated.

In order to improve the efficiency of the simulation method presented herein, parallel and graphics processing unit (GPU) programming are to be tested. Besides, for simulating continuous variables, the results can be improved by calculating the histogram of values and storing the bins the values of each location in the pattern belongs to, instead of storing the values themselves.

List of References

Albor FR, Dimitrakopoulos R (2009) Stochastic mine design optimisation based on simulated annealing: pit limits, production schedules, multiple orebody scenarios and sensitivity analysis. Min. Technol. (Trans. Inst. Min. Metall. A), 118, (2), 80–91.

Arpat G, and Caers J (2007) Conditional simulation with patterns. Mathematical Geology 39 (2):177-203.

Arpat GB (2005) Sequential simulation with patterns. PhD thesis, Stanford University, Stanford, CA, USA.

Baker CK, Giacomo S M (2001) Resources and reserves: their uses and abuses by the equity markets. In Edwards, A. C. (Ed.), Mineral Resource and Ore Reserve Estimation – The AusIMM Guide to Good Practice (pp 666–676). Melbourne, Australia: AusIMM.

Bastrakov, EN, Skirrow, RG, Davidson, GJ (2007) Fluid evolution and origins of iron oxide Cu-Au prospects in the Olympic Dam district, Gawler Craton, South Australia. Economic Geology, 102(8), 1415-1440.

Belperio A, Freeman H (2004) Common geological characteristics of Prominent Hill and Olympic Dam-implications for iron oxide coppergold exploration models. Australian Institute of Mining Bulletin, Nov-Dec. Issue, 67-75.

Bénéteau C, Van Fleet PJ (2011) Discrete wavelet transformations and undergraduate education. Notices of the AMS 58.05.

Boland N, Dumitrescu I, Froyland G (2008) A multistage stochastic programming approach to open pitmine production scheduling with uncertain geology. Optimization Online.Retrieved from www.optimizationonline.org/DBHTML/2008/10/2123.html

Boland N, Dumitrescu I, Froyland G, Gleixner AM (2009). LP-based disaggregation approaches to solving the open pit mining production scheduling problem with block processing selectivity. *Computers & Operations Research,* 36, (4), 1064–1089.

Boucher A, Dimitrakopoulos R (2009a) Block simulation of multiple correlated variables. Mathematical Geosciences, 41(2), 215-237.

Boucher A (2009b) Considering complex training images with search tree partitioning. Computers & Geosciences, 35(6): 1151-1158.

Boucher A. Costa JF, Rasera LG, Motta E (2014) Simulation of geological contacts from interpreted geological model using multiple-point statistics. Mathematical Geosciences, 1-12.

Bratvold RB, Holden L, Svanes T, Tyler K, (1994) STORM: integrated 3D stochastic reservoir modeling tool for geologists and reservoir engineers. SPE Paper Number 27563, pp. 242–261.

Caccetta L, Hill SP (2003) An application of Branch and Cut to open pit mine scheduling. Journal of Global Optimization, 27, (2-3), 349-365.

Caers J (2011) Modeling Uncertainty in the Earth Science, Willey-Blackwell, Chichester, UK.

Caers J, Zhang T (2004) Multiple-point geostatistics: a quantitative vehicle for integrating geologic analogs into multiple reservoir models. Memoirs – American Association of Petroleum Geologists, 383-394.

Chatterjee S, Mustapha H, Dimitrakopoulos, R (2011) Geologic heterogeneity representation using high-order spatial cumulants for subsurface flow and transport simulations. Water Resources Research, 47(8).

Chatterjee S, Dimitrakopoulos R, Mustapha H (2012) Dimensional reduction of pattern-based simulation using wavelet analysis, Mathematical Geosciences 44: 343-374.

Cox TF, Cox MA (2010) Multidimensional scaling. Boca Raton (Florida), CRC Press.

Dagdelen K, Johnson TB (1986) Optimum open pit mine production scheduling by lagrangian parameterization. In Proc. 19th Internat. Appl. Comput. Oper. Res. Mineral Indust. Sympos. (APCOM) (pp. 127-142). Littleton, CO: SME.

David M. (1977) Geostatistical ore reserve estimation. Amsterdam: Elsevier.

David M (1988) Handbook of applied advanced geostatistical ore reserve estimation. Amsterdam, Elsevier

David M, Dowd P, Korobov S (1974) Forecasting departure from planning in open pit design and grade control. In Proc. 12th Internat. Appl. Comput. Oper. Res. Mineral Indust. Sympos. (APCOM) (pp. F131-F142). Golden, CO.

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

De Iaco S, Maggio S (2011) Validation techniques for geological patterns simulations based on variogram and multiple-point statistics. Mathematical Geosciences, 43(4), 483-500.

Dimitrakopoulos R, Farrelly CT, Godoy M (2002) Moving forward from traditional optimization: grade uncertainty and risk effects in open pit design, Min. Technol. (Trans. Inst. Min. Metall. A), 111(1), 82–88

Dimitrakopoulos R, Luo X (2004) Generalized sequential Gaussian simulation on group size n and screen-effect approximations for large field simulations. Mathematical Geology, 36, (5), 567-591.

Dimitrakopoulos R, Mustapha H, and Gloaguen E (2010) High-order statistics of spatial random fields: Exploring spatial cumulants for modeling complex non-Gaussian and non-linear phenomena. Mathematical Geosciences 42(1):65-99.

Dimitrakopoulos R (2011) Stochastic optimization for strategic mine planning: a decade of developments. Journal of Mining Science, 47(2): 138-150.

Ding C and He X (2004) K-means clustering via principal component analysis. Proc. of Int'l Conf. Machine Learning (ICML 2004): 225-232.

Dowd P (1994) Risk assessment in reserve estimation and open pit planning, Min. Technol. (Trans. Inst. Min. Metall. A), 103, 148–154.

Dowd P (1997) Risk in minerals projects: analysis, perception and management, Min. Technol. (Trans. Inst. Min. Metall. A), 106, 9–18.

Faucher C, Saucier A, Marcotte D (2013) A new patchwork simulation method with control of the local-mean histogram. Stochastic Environmental Research and Risk Assessment, 27(1), 253-273.

Godoy M (2003) The effective management of geological risk in long-term production scheduling of open pit mines, Ph.D. thesis (unpublished), University of Queensland, Brisbane, Australia.

Goodfellow R, Albor Consuegra F, Dimitrakopoulos R, Lloyd T (2012) Quantifying multi-element and volumetric uncertainty, Coleman McCreedy deposit, Ontario, Canada. Computers & Geosciences, 42, 71-78.

Goodfellow R, Dimitrakopoulos R (2013) Mining supply chain optimization under geological uncertainty. Retrieved from Les Cahiers du GERAD website: http://www.gerad.ca/fichiers/cahiers/G-2013-54.pdf

Goodfellow R (2014) Unified modelling and simultaneous optimization of open pit mining complexes with supply uncertainty. Thesis (Ph.D.) - Montreal: McGill University Libraries

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

Guardiano F, Srivastava R (1993) Multivariate geostatistics: beyond bivariate moments. In A. Soares (Ed.), Geostatistics Troia. Vol. 1 (pp. 133-144). Kluwer Academic Publications.

Hahn T (2008) The Olympic Dam Cu-U-Au-REE Deposit, Australia. Retrieved from: http://www.geo.tu-freiberg.de/oberseminar/os07_08/Thomas%20Hahn.pdf.

Haldorsen HH, Lake LW (1984) A new approach to shale management in field-scale models. SPE Journal, April, pp.447–457.

Hartigan, JA, Wong, MA (1979) Algorithm AS 136: A K-means clustering algorithm. Journal of the Royal Statistical Society, Series C (Applied Statistics) 28 (1): 100–108.

Honarkhah M, Caers J (2010) Stochastic simulation of patterns using distance-based pattern modeling. Mathematical Geosciences, 42(5), 487-517.

Honarkhah M (2011) Stochastic simulation of patterns using distance-based pattern modeling. Ph.D. thesis, Stanford University, Stanford, CA, USA.

Honarkhah M, Caers J (2012) Direct pattern-based simulation of non-stationary geostatistical models. Mathematical Geosciences, 44(6), 651-672.

Huang T, Lu DT, Li X, Wang L (2013) GPU-based SNESIM implementation for multiple-point statistical simulation. Computers & Geosciences, 54, 75-87.

Hustrulid W, Kuchta M (1995) Open pit mining planning and design. AA Balkema, Rotterdam.

Isaaks EH (1990) The Application of Monte Carlo methods to the analysis of spatially correlated data, Ph.D. Thesis, Stanford U., 213 pp.

Johnson T (1969) Optimum production scheduling. In Proceedings of the 8th International Symposium on Computers and Operations Research (pp. 539–562). Salt Lake City.

Jones P, Douglas I, Jewbali A (2013) Modeling combined geological and grade uncertainty: application of multiple-point simulation at the Apensu gold deposit, Ghana. Mathematical Geosciences, 45(8), 949-965.

Journel AG (1974) Geostatistics for conditional simulation of ore bodies. Economic Geology, 69(5), 673-687.

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

Journel AG, Alabert FG (1988) Focusing on spatial connectivity of extreme-valued attributes: Stochastic indicator models of reservoir heterogeneities. SPE paper, 18324.

Journel AG (1989a) Fundamentals of geostatistics in five lessons (Vol. 8). Washington, DC: American Geophysical Union.

Journel AG, Alabert F (1989b) Non-Gaussian data expansion in the Earth Sciences. Terra Nova, 1(2), 123-134.

Journel AG (2005) Beyond covariance: the advent of multiple-point geostatistics. In Geostatistics Banff 2004 (pp. 225-233). Springer Netherlands.

Journel AG (2007) Roadblocks to the evaluation of ore reserves - the simulation overpass and putting more geology into numerical models of deposits. In R. Dimitrakopoulos (Ed.), Orebody Modelling and Strategic Mine Planning - Uncertainty and Risk Management International Symposium (pp. 29–32). Burwood: AusIMM.

Kolmogorov AN (1956) Foundations of the theory of probability. New York: Chelsea Publishing Co.

Lamghari A, Dimitrakopoulos R (2012) A diversified tabu search approach for the open-pit mine production scheduling problem with metal uncertainty. European Journal of Operational Research, 222, 642-652.

Leite A, Dimitrakopoulos R (2007) Stochastic optimization model for open pit mine planning: application and risk analysis at a copper deposit. Min. Technol. (Trans. Inst. Min. Metall. A), 116, (3), 109-118.

Leite A, Dimitrakopoulos R (2009) Production scheduling under metal uncertainty – Application of stochastic mathematical programming at an open pit copper mine and comparison to conventional scheduling. In R. Dimitrakopoulos (Ed.), Orebody Modeling and Strategic Mine Planning (pp. 35-39). AusIMM.

Lerchs H, Grossmann IF (1965) Optimum design of open-pit mines. Canad. Inst. Mining Bull., 58, 47-54.

Liu J, Musialski P, Wonka P, Ye J (2013) Tensor completion for estimating missing values in visual data. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 35(1), 208-220.

Lloyd SP (1982) Least square quantization in PCM, IEE Transactions on Information Theory, 2, (2), 129-137.

MacQueen JB (1967) Some methods for classification and analysis of multivariate observations. Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, University of California Press, 1:281-297.

Mallat S. (1998) A wavelet tour of signal processing, Academic Press, San Diego, CA.

Mariethoz G, Renard P, Straubhaar J (2010) The direct sampling method to perform multiple-point geostatistical simulations. Water Resources Research, 46(11).

Matheron G (1973) The intrinsic random functions and their applications. Advances in Applied Probability, 5(3), 439-468.

Menabde M, Froyland G, Stone P, Yeates G (2007) Mining schedule optimisation for conditionally simulated orebodies. In R. Dimitrakopoulos (Ed.), *Orebody Modelling and Strategic Mine Planning - Uncertainty and Risk Management International Symposium* (pp. 379–384). Burwood: AusIMM.

Montiel L, Dimitrakopoulos R (2013) An extended stochastic optimization method for multi-process mining complexes. Retrieved from Les Cahiers du GERAD website: www.gerad.ca/fichiers/cahiers/G-2013-56.pdf.

Montiel L (2014) On globally optimizing a mining complex under supply uncertainty: integrating components from deposits to transportation systems. Thesis (Ph.D.) - Montreal: McGill University Libraries.

Mustapha H, Chatterjee S, Dimitrakopoulos R (2014) CDFSIM: efficient stochastic simulation through decomposition of cumulative distribution functions of transformed spatial patterns. Mathematical Geosciences, 46(1), 95-123.

Mustapha H, Dimitrakopoulos R (2010) High-order stochastic simulations for complex non-Gaussian and non-linear geological patterns, Mathematical Geosciences, 42 (5).

Osterholt V, Dimitrakopoulos R (2007) Simulation of wireframes and geometric features with multiple-point techniques: application at Yandi iron ore deposit. Orebody modelling and strategic mine planning, AusIMM, Spectrum Series, 14, 95-124.

Picard JC (1976) Maximal closure of a graph and applications to combinatorial problems. Management Science, 22, (11), 1268-1272.

Ramazan S, Dimitrakopoulos R (2007) Stochastic optimization of long term production scheduling for open pit mines with a new integer programming formulation. In R. Dimitrakopoulos (Ed.), Orebody Modelling and Strategic Mine Planning - Uncertainty and Risk Management International Symposium (pp. 359–365). Burwood: AusIMM.

Ramazan S, Dimitrakopoulos R (2013) Production scheduling with uncertain supply: A new solution to the open pit mining problem. Optimization and Engineering, 14, (2), 361–380.

Ravenscroft P (1992) Risk analysis for mine scheduling by conditional simulation. Min. Technol. (Trans. Inst. Min. Metall. A), 101, 104–108.

Rezaee H, Mariethoz G, Koneshloo M, Asghari O (2013) Multiple-point geostatistical simulation using the bunch-pasting direct sampling method.Computers & Geosciences, 54, 293-308.

Roberts DE, Hudson GRT (1983) The Olympic Dam copper-uranium-gold deposit, Roxby Downs, South Australia. Economic Geology, 78(5), 799-822.

Rosenblatt M (1952) Remarks on a multivariate transformation. The Annals of Mathematical Statistics, 470-472.

Rosenblatt M (1985) Stationary sequences and random fields. Birkhaüser, Boston.

Skirrow RG, Bastrakov EN, Barovich K, Fraser GL, Creaser RA, Fanning CM, Raymond OL, Davidson GJ (2007) Timing of iron oxide Cu-Au-(U) hydrothermal activity and Nd isotope constraints on metal sources in the Gawler Craton, South Australia. Economic Geology, 102(8), 1441-1470.

Straubhaar J, Renard P, Mariethoz G, Froidevaux R, Besson O (2011) An improved parallel multiple-point algorithm using a list approach. Mathematical Geosciences, 43(3), 305-328.

Strebelle S (2000) Sequential simulation drawing structures from training images. PhD thesis, Stanford University.

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

Strebelle S, Cavelius C (2014) Solving speed and memory issues in multiple-point statistics simulation program SNESIM. Mathematical Geosciences, 46(2), 171-186.

Strebelle S, Zhang T (2005) Non-stationary multiple-point geostatistical models, Geostatistics Banff 2004 (pp. 235-244). Springer Netherlands.

Tolwinski B, Underwood R (1996). A scheduling algorithm for open pit mines. IMA Journal of Mathematics Applied in Business & Industry, 7, 247-270.

Tran TT (1994) Improving variogram reproduction on dense simulation grids. Computers & Geosciences, 20(7), 1161-1168.

Tyler K, Henriquez A, Georgsen F, Holden L, Tjelmeland H (1992a) A program for 3d modeling of heterogeneities in a fluvial reservoir. In 3rd European Conference on the Mathematics of Oil Recovery, Delft, June, pp. 31–40.

Vallee M (2000) Mineral resource + engineering, economic and legal feasibility. CIM bulletin, 93, (1038), 53-61.

Whittle J (1988) Beyond optimization in open-pit design. In K. Fytas (Ed.), Proc. Computer Applications in the Mineral Industries (pp. 331– 337. 115). Quebec City: Balkema.

Whittle J (2010) The global optimizer works-what next. The Australasian Institute of Mining and Metallurgy, Spectrum Series, 17.

Wu J, Zhang T, Journel A (2008) Fast FILTERSIM simulation with score-based distance, Mathematical Geosciences, 40(7): 773-788.

Yahya WJ (2011) Image reconstruction from a limited number of samples: A matrix completion based approach. Montreal: McGill University Libraries.

Zhang T, Switzer P, Journel AG (2006) Filter-based classification of training image patterns for spatial simulation. Mathematical Geology, 38(1): 63–80.

Zhang T (2006) Filter-based training pattern classification for spatial pattern simulation. Ph.D. thesis, Stanford University.